Inverter

8400

E84AVTCxxxxx...

Inverter Drives 8400 TopLine C _ _ _ _ _ _ _ _ _ _

Reference manual EN

13572906

Lenze

Overview of technical documentation for Inverter Drives 8400

Project planning, selection & ordering
☐ 8400 hardware manual
🗎 Catalogue

Mounting & wiring
🗎 MA 8400 BaseLine/StateLine/HighLine/TopLine
🗎 MA for the communication module
🗎 MA for the extension module
🗎 MA for the safety module
🗎 MA for the accessories

Parameter setting	
🗎 BA keypad	
☐ SW 8400 BaseLine	
☐ SW 8400 StateLine	
☐ SW 8400 HighLine	
☐ SW 8400 TopLine	← **This documentation**
☐ KHB for the communication module	

Drive commissioning
☐ SW 8400 BaseLine/StateLine/HighLine/TopLine
→ chapter "Commissioning"
→ chapter "Diagnostics & error management"
☐ Remote maintenance manual

Networking
☐ KHB for the communication medium used

Legend:	
🗎	Printed documentation
☐	Online documentation (PDF/Engineer online help)
Abbreviations used:	
BA	Operating instructions
KHB	Communication manual
MA	Mounting instructions
SW	Software/reference manual

Contents

1 About this documentation

Danger!

The inverter is a source of danger which may lead to death or the severe injury of persons.

To protect yourself and others against these dangers, observe the safety instructions before switching on the inverter.

Please read the safety instructions provided in the **8400 mounting instructions** and in the **8400 hardware manual**. Both documents are supplied with the inverter.

Target group

This documentation is intended for all persons who would like to parameterise, configure and diagnose the 8400 TopLine inverter with the Lenze engineering software »Engineer« and the X400 keypad.

Information regarding the validity

The information in this documentation are valid for the following standard devices:

Product range	Type designation	From software version
8400 TopLine C	E84AVTCxxxxx	01.00

Screenshots/application examples

All screenshots provided in this documentation are application examples. Depending on the software version of the inverter and the version of the »Engineer« software installed, the screenshots in this documentation may differ from the representation in the »Engineer«.

Tip!

Information and tools regarding the Lenze products can be found on the Internet:

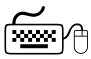
http://www.lenze.com → Download

1.1 Document history

Version			Description
11.1	10/2019	TD06	Corrected term C00142
11.0	05/2019	TD06	Error corrections & supplements for 8400 TopLine C (FW21.00.00)
10.1	02/2018	TD23	Error corrections & supplements
10.0	01/2018	TD23	Extended by new functions for 8400 TopLine C V18.00.00, error corrections
9.0	06/2017	TD23	Extended by new functions for 8400 TopLine C V17.00.00, error corrections & supplements
8.0	02/2016	TD06	Error corrections & supplements
7.1	01/2016	TD06	Oscilloscope function (📖 784) revised
7.0	12/2015	TD06	Extended by new functions for 8400 TopLine C V16.00.00
6.0	06/2014	TD05	Extended by new functions for 8400 TopLine C V15.00.00
5.0	11/2013	TD05	Extended by new functions for 8400 TopLine C V14.00.00
4.1	01/2013	TD05	Error corrections & supplements
4.0	11/2012	TD05	Extended by new functions for 8400 TopLine C V13.00.00 and converted to new layout.
3.1	09/2012	TD05	Error corrections & supplements
3.0	07/2012	TD05	Extended by new functions for 8400 TopLine C V12.00.00
2.0	08/2011	TD05	Extended by new functions for 8400 TopLine C V02.00.00

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Numeric notation		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the "0x" prefix is used. Example: 0x60F4
Binary number	0b	For binary numbers, the "0b" prefix is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that only applies to or from a certain software version of the inverter is marked accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software...
Window	*italics*	The *Message window*... / The dialog box *Options*...
Variable names		By setting *bEnable* to TRUE...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If several commands must be used in sequence to carry out a function, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Use **<F1>** to open the online help.
		If a key combination is required for a command, a "+" is inserted between the key identifiers: Use **<Shift>+<ESC>**...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Symbols		
Page reference	(📖 30)	Optically highlighted reference to another page. It is activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

All information that only applies to or from a certain software version of the inverter is marked accordingly in this documentation.

1.3 Terminology used

Term	Meaning
Engineering tools	Software solutions for easy engineering in all project stages
	»EASY Navigator« – ensures easy operator guidance • All convenient Lenze engineering tools at a glance • Tools can be quickly selected • The clear structure simplifies the engineering process from the start
	»EASY Starter« – easy-to-use tool for service technicians • Specifically designed for commissioning and maintaining Lenze devices • Graphic user interface with very few icons • Easy to run online diagnostics, set parameters and perform commissioning • No risk of accidentally changing an application • Loading off-the-shelf applications onto the device
	»Engineer« – multi-device engineering • For all products in our L-force portfolio • Practical user interface • Graphic interfaces make it easy to navigate • Can be applied in every phase of a project (project planning, commissioning, production) • Parameter setting and configuration
Application block	Block for a technology application (e.g. actuating drive speed) A technology application is a drive solution based on the experience and know-how of Lenze in which function blocks interconnected to a signal flow form the basis for implementing typical drive tasks.
ASM	Abbreviation for asynchronous motor
Service brake	The service brake serves to shutdown rotary or translatory masses in motion in a controlled manner. The energy to be dissipated in this process is produced in the form of friction energy. Unlike emergency braking, this process is a regular and recurring operating mode.
CAN	Abbreviation for Controller Area Network. CAN is an asynchronous, serial fieldbus system.
CANopen	CANopen® is a CAN-based communication protocol. The Lenze system bus (CAN on board) works with a subset of this communication protocol. CANopen® is a registered community trade mark of the CiA® (CAN in Automation e. V.) CAN user organisation. ▶ System bus "CAN on board"
Code	Parameter used for inverter parameterisation or monitoring. Is usually referred to as "index".
Display code	Parameter that displays the current status or value of an input/output of a system block.
EtherCAT	EtherCAT® is a real-time capable Ethernet system with top performance. EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
ETHERNET POWERLINK	Ethernet POWERLINK is a real-time capable fieldbus system based on Ethernet. For user data exchange, Ethernet POWERLINK specifies a communication protocol based on CANopen. Ethernet POWERLINK is a patented technology licensed by the Ethernet POWERLINK Standardization Group (EPSG), Germany.
EtherNet/IP	EtherNet/IP™ (EtherNet Industrial Protocol) is an Ethernet-based fieldbus system that uses Common Industrial Protocol™ (CIP™) to exchange data. EtherNet/IP™ and Common Industrial Protocol™ (CIP™) are brand labels and patented technologies, licensed by the ODVA (Open DeviceNet Vendor Association) user organisation, USA.
FB Editor	Abbreviation for "function block editor". Graphic interconnection tool which is available in the »Engineer« for function block interconnections on the **FB Editor**. ▶ Working with the FB Editor

Term	Meaning
Function block	General designation of a function block for free interconnection in the FB Editor. A function block (short: FB) can be compared with an integrated circuit that contains a specific control logic and delivers one or several values when being executed. Example: "L_Arithmetik_1" (FB for arithmetic operations) Many function blocks are available several times (e.g. L_And_1, L_And_2, and L_And_3). ▶ Function blocks
Holding brake	The holding brake serves to statically hold e.g. a position during the downtimes of a robot/ travelling/synchronous/hoist drive.
HIPERFACE®	HIPERFACE® stands for High Performance Interface and is a universal interface between motor feedback system and inverter. HIPERFACE® is a registered trademark of the SICK STEGMANN GmbH.
INTERBUS	INTERBUS was developed as sensor/actuator/bus system for transmitting process data. Today, maintenance activities for the INTERBUS technologies are carried out by the user organisation PROFIBUS & PROFINET International (PI).
Keypad	The keypad is an alternative to the PC for the local operation, parameterisation, and diagnostics in a simple manner.
LA	Abbreviation for **L**enze **A**pplication block Example: "LA_NCtrl" (block for the "Actuating drive speed" application)
Lenze setting	This setting is the default factory setting of the device.
LP	Abbreviation for **L**enze **P**ort block Example: "LP_CanIn1" (CAN1 port block)
LS	Abbreviation for **L**enze **S**ystem block Example: "LS_DigitalInput" (system block for digital input signals)
MCI	Abbreviation for **M**otionbus **C**ommunication **I**nterface (fieldbus interface) The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system. ▶ Fieldbus interface (MCI)
Emergency brake	The emergency brake serves to shutdown rotary or translatory masses in motion in emergency situations. Emergency situations are exceptional situations that only occur sporadically.
Port block	Block for implementing the process data transfer via a fieldbus
PROFIBUS® PROCESS FIELD BUS	PROFIBUS® (Process Field Bus) is fieldbus system used worldwide for automating machines and production plants. PROFIBUS® is a registered trademark and patented technology licensed by the PROFIBUS & PROFINET International (PI) user organisation.
PROFINET® INDUSTRIAL ETHERNET	PROFINET® (Process Field Network) is a real-time capable fieldbus system based on Ethernet. PROFINET® is a registered trademark and patented technology licensed by the PROFIBUS & PROFINET International (PI) user organisation.
PSM	Abbreviation for permanently excited synchronous motor
QSP	Abbreviation for quick stop
SC	Abbreviation for **S**ervo **C**ontrol
SLPSM	Abbreviation for sensorless control of synchronous motors
SLVC	Abbreviation for **S**ensor**L**ess **V**ector **C**ontrol
SSI	Abbreviation for **S**ynchronous **S**erial **I**nterface Interface for absolute angle encoders developed by SICK STEGMANN GmbH.
Subcode	If a code contains several parameters, they are stored in "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). This term is also referred to as "subindex" in common parlance.
System block	In the function block editor of the »Engineer«, system blocks provide interfaces to basic functions, "free codes", and to the hardware of the inverter (e.g. to the digital inputs). Each system block is available only once. ▶ System blocks

Term	Meaning
USB diagnostic adapter	The USB diagnostic adapter is used for the operation, parameterisation, and diagnostics of the inverter. Data are exchanged between the PC (USB connection) and the inverter (diagnostic interface on the front) via the diagnostic adapter. Order designation: E94AZCUS
VFCplus	Abbreviation for **V**oltage **F**requency **C**ontrol

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Structure of the safety instructions:

Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of damage to material assets** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for easy handling

2 Introduction: Parameterising the inverter

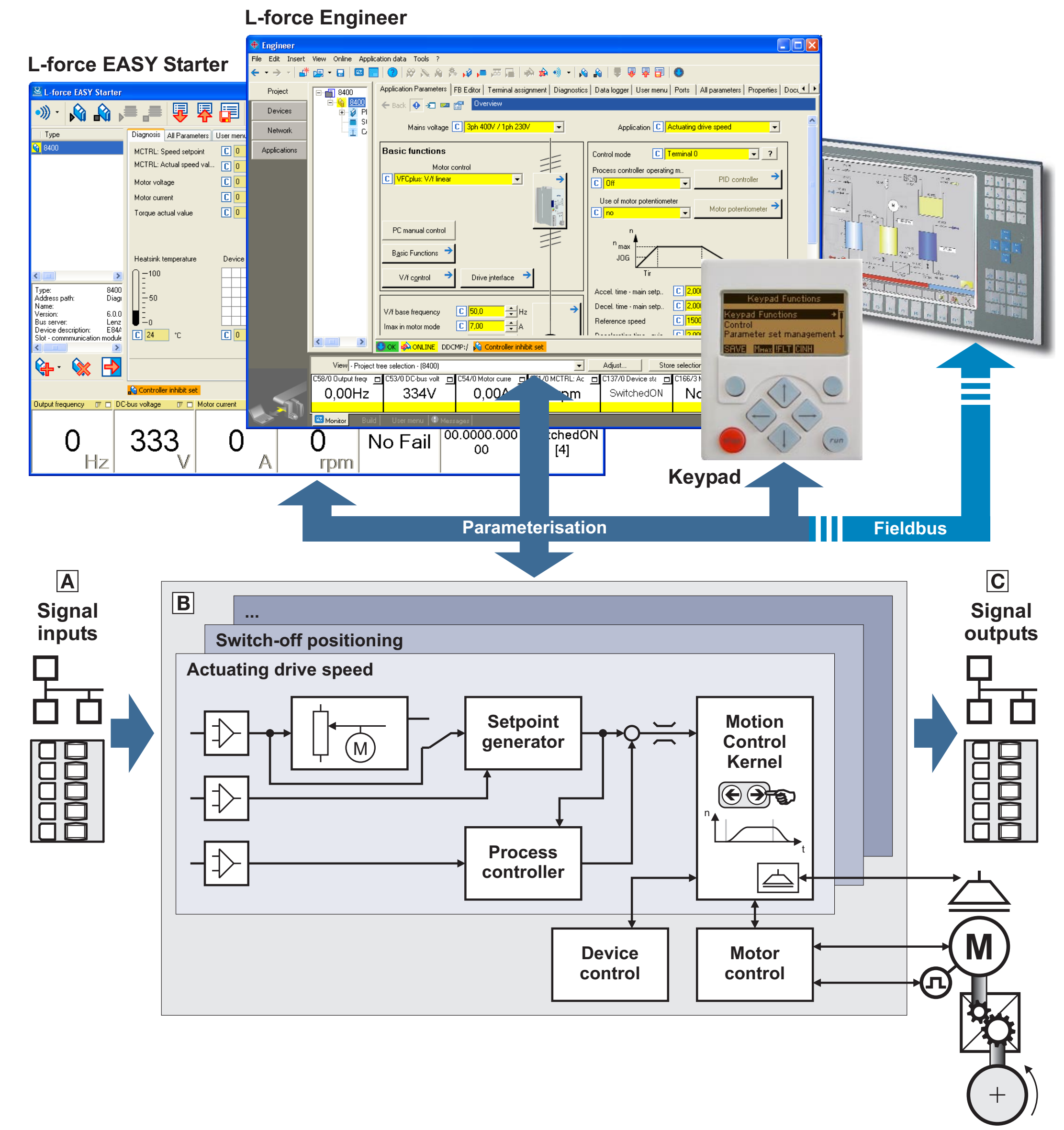
Being a component of a machine which includes a speed-variable drive system, the inverter needs to be adjusted to its drive task. The inverter is adjusted by changing parameters which are saved in the memory module. The parameters can be accessed by keypad, by the »EASY Starter« or by the »Engineer«. Access is also possible by a master control via fieldbus communication. For this purpose, the "CAN on board" CAN interface and the MCI interface for using a communication module (e.g. PROFIBUS) are available.

Danger!

In general, changing a parameter causes an immediate response in the inverter!

An enabled inverter can cause an unwanted behaviour at the motor shaft! For instance, setpoint sources can change over abruptly (e.g. when the signal source is configured for the main setpoint).

Certain device commands or settings which may cause critical states of drive behaviour constitute exceptions. Such parameter changes are only possible if the inverter is inhibited. Otherwise, a corresponding error message will be issued.



A Signal inputs for control and setpoint signals
B Signal flow of the integrated technology application (see the following subchapter)
C Signal outputs for status and actual value signals

[2-1] Adaptation of the drive solution via parameter setting

2.1 Integrated technology applications

The following technology applications integrated in the inverter 8400 TopLine provide the main signal flow for the implementation of a general or a special drive solution:

Technology application "Actuating drive speed"

This preset technology application serves to solve speed-controlled drive tasks, e.g. conveyor drives (interconnected), extruders, test benches, vibrators, travelling drives, presses, machining systems, metering units.

Technology application "actuating drive speed (AC Drive profile)"

This technology application available from version 13.00.00 provides a speed and torque control by means of "AC Drive Profile". The fieldbuses EtherNet/IP™ and system bus (CANopen) are supported.

Technology application "Table positioning"

This technology application serves to solve position-controlled drive tasks which are normally controlled by a higher-level control via a fieldbus, e.g. transport facilities, rotary tables, storage and retrieval units, feed drives, metering units, hoists.

Note: This TA requires an external sequence control!
From version 12.00.00, the FB L_Sequencer_1 can also be used as internal sequence control. This FB processes a positioning program based on a sequence table.

"Switch-off positioning" technology application

This technology application is used to solve speed-controlled drive tasks which require a pre-switch off or stopping at certain positions, e.g. roller conveyors and conveying belts. The pre-switch off is implemented by connecting switch-off sensors.

Note!

Please note that the "StateLine", "HighLine" and "TopLine" device types differ with regard to the number, functional range, and flexibility of the technology applications offered.

Detailed information on each technology application can be found in the main chapter entitled "Technology applications". (📖 452)

2.1.1 Purpose of the technology applications

The stepped Inverter Drives 8400 series provides solutions for simple to complex applications – depending on the user's experience and knowledge about the handling of drives and drive tasks.

On the one hand, a great scope of standard drive tasks for frequency inverters is covered by the technology applications offered by Lenze, and on the other hand, the user is relieved from time-consuming programming activities. In practice, some drive tasks are alike so that minor modifications of the corresponding technology applications lead to quick results.

Other important features of technology applications are:

- Direct implementation of drive tasks without recreating a function block interconnection inside the device
- Operation via keypad and/or operation via convenient operator dialogs in the «Engineer».
- Commissioning via few operating and diagnosing parameters (local keypad operation).
- Achieving a transparency as high as possible via the integrated functionality of the device by representing signal flow diagrams.
- Provision of a basic functionality suitable and often sufficient for many applications.

2.1.2 Application cases for a technology application

You should use a technology application if

- the task can be solved completely or to a great extent by the basic functionality of the technology application.
- the end customer does not want to create the comprehensive core functions of the corresponding technology on his own.
- the creation time for a project is to be reduced by using the ready-made technology application
- the end customer wants to build upon the know-how of Lenze.

Tip!

If the end customer of the machine does not want to use ready-made Lenze functions, it is also possible to implement individual drive solutions with the "StateLine C", "HighLine C" and "TopLine C" versions in the form of "free interconnections". (In case of "StateLine C", free interconnection is only possible from version 12.00.00 and »Engineer« V2.17.)

Here, a technology application can be used as starting basis, which has to be adapted to the requirement by a change or extension via function block editor (see the following chapter).

2.1.3 Technology application = function block interconnection

In case of the 8400 device version, each technology application is connected to a "function block interconnection" ("FB interconnection"). This FB interconnections serve to implement signal interconnections. Various FBs are available for digital signal processing, signal conversion and logic modules.

For special drive tasks it has proved of value to use the integrated technology applications as a basis for modifications or extensions of the available FB interconnections.

I/O level & application level

The interconnection of the interfaces is shown in the I/O level of the function block editor according to the selected control mode. In the "deeper" application level, the main signal flow is realised in the form of an interconnection of various function and system blocks.

For the device versions "StateLine C", "HighLine C" and "TopLine C" the following applies:

- The preconfigured signal links can be reconfigured in the I/O level if required.
- Moreover, the applications integrated in the inverter can be reconfigured and extended by individual functions.
- Moreover, experienced users are offered the opportunity to implement their own drive solutions independent of the predefined technology applications by using so-called "free interconnections". (In case of "StateLine C", free interconnection is only possible from version 12.00.00 and »Engineer« V2.17.)

Motion control kernel

Important basic (drive) functions as well as further basic functionalities are implemented in the firmware of the inverter in the so-called **Motion Control Kernel** (MCK) which can be accessed by the active technology application via defined internal interfaces. By this means the expensive creation of single function block interconnections is omitted so that the expenditure and the complexity for the realisation of standard functions is minimised.

The **Motion Control Kernel** is integrated in the main setpoint path and, depending on the set operating mode, it creates the required control and setpoint signals for the motor control and the drive interface.

More information:

- A detailed description of the basic functions implemented in the **Motion Control Kernel** can be found in the main chapter "Basic drive functions (MCK)". (📖 577)
- Detailed information on the creation or change of interconnections by means of the function block editor can be found in the main chapter "Working with the FB Editor". (📖 1416)
- All available function and system blocks are described in the main chapter "Function library". (📖 1468)

2.2 Selection of the appropriate commissioning tool

There are several possibilities for commissioning the 8400 TopLine inverter:

Commissioning via keypad X400 (or diagnosis terminal X400)

The keypad is an alternative to the PC for the local operation, parameterisation, and diagnostics in a simple manner. The keypad is especially suited for test and demonstration purposes and for the case that only few parameters have to be adapted.

Commissioning with PC and »EASY Starter«

The »EASY Starter« is a Lenze tool for easy online diagnostics, parameter setting and commissioning of the inverter.

Commissioning with PC and »Engineer«

The »Engineer« is a Lenze engineering software for parameter setting across all devices, configuring and diagnosing individual components (as for instance inverters, industrial PCs, motors, I/O systems) and machine control systems.

Tip!

The Engineering tools »EASY Starter« and »Engineer StateLevel« are provided free of charge in the internet:

http://www.lenze.com → Download → Software Downloads

For communication between PC and inverter, the USB diagnostic adapter can be used for instance (see the following subchapter).

2.2.1 Overview: Accessories for commissioning

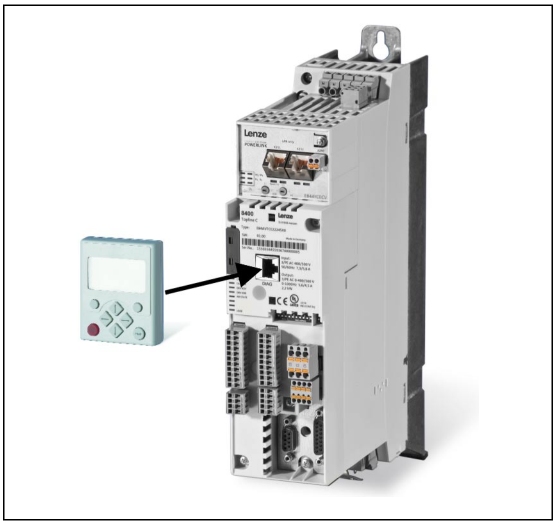
Version	Features	Product key
Keypad X400 	Quick access to parameters and operating data • Supports hot plugging • Graphic display with plain texts • Backlighting • Easy user guidance • 4 navigation keys, 2 context-sensitive keys • Adjustable RUN/STOP function • Can be used for L-force Inverter Drives 8400 and Servo Drives 9400	EZAEBK1001
Diagnosis terminal X400 	Keypad X400 in a robust housing, also suitable for installation into the control cabinet door. • Supports hot plugging • Graphic display with plain texts • Backlighting • Easy user guidance • 4 navigation keys, 2 context-sensitive keys • Adjustable RUN/STOP function • Incl. 2.5 m cable • Enclosure IP20; in case of front installation in control cabinet IP65 • Can be used for L-force Inverter Drives 8400 and Servo Drives 9400	EZAEBK2001
USB diagnostic adapter 	For electrical isolation of your PC and the inverter. • Supports hot plugging • Diagnostic LED for data transfer display • plug and play • Input-side voltage supply via USB connection from PC • Output-side voltage supply via the diagnostic interface of the inverter • Connecting cables can be selected in various lengths:	E94AZCUS
Connecting cable for USB diagnostic adapter	2.5 m length	EWL0070
	5 m length	EWL0071
	10 m length	EWL0072

2.3 General notes on parameters

All parameters for inverter parameterising or monitoring are saved as so-called "codes".

- The codes are numbered and indicated by the prefix "C" before the code, e.g. "C00002".
- Moreover, each code has a name and specific attributes, as for example access type (reading, writing), data type, limit values and default setting ("Lenze setting").
- For the sake of clarity, some codes contain "subcodes" for saving parameters. This Manual uses a slash "/" as a separator between code and subcode, e.g. C00118/3".
- According to their functionality, the parameters are divided into three groups:
  - Setting parameters: For specifying setpoints and for setting device / monitoring functions.
  - Configuration parameters: For configuring signal connections and terminal assignments.
  - Diagnostic/display parameters: For displaying device-internal process factors, current actual values and status messages. These are read-only parameters.

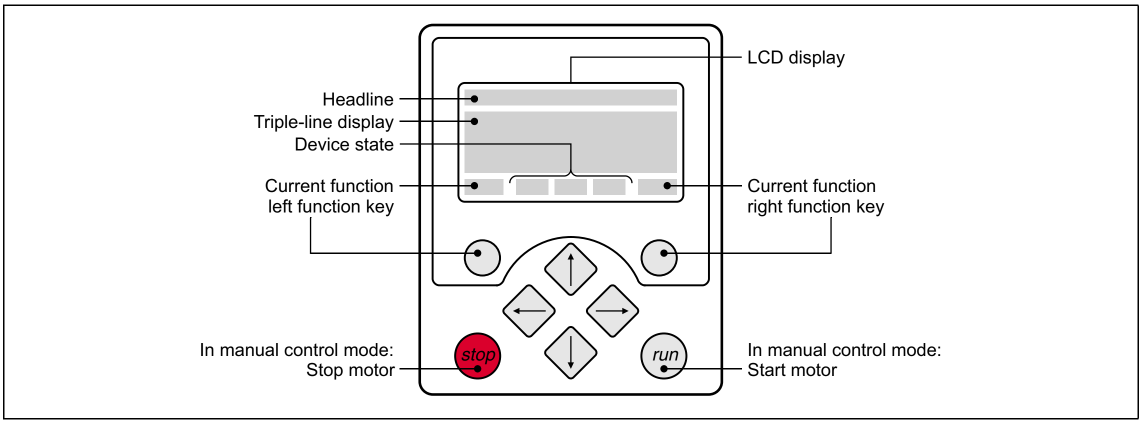
2.3.1 Changing the parameterisation with the keypad



The keypad is simply plugged on the diagnostic interface X6 ("DIAG") at the front of the standard device.

Plugging and unplugging the keypad is possible during operation.

Keypad display and control elements



LCD display			
Headline			
In the menu level: Menu name In the parameter level: Parameter name			
Three-part display			
In the menu level: List of available menus In the parameter level: Code/subcode and setting or actual value			
Device status			
RDY	Inverter is switched on	IMP	Pulse inhibit active
RUN	Inverter is enabled	!FLT	"Fault" device status is active
CINH	Inverter is inhibited	!TRB	"Trouble" device status is active
QSP	Quick stop active	!TQSP	"TroubleQSP" device status is active
Imax	Current limit exceeded	WRN	A warning is indicated
Mmax	Speed controller 1 in the limitation		

LCD display			
Function - left function key		**Function - right function key**	
EDIT	Change parameter setting (change to editing mode)	OK	Accept change in the inverter (no saving with mains failure protection → SAVE)
	Back to main menu	ESC	Abort (discard change)
CINH!!	Parameter can only be changed when the inverter is inhibited		
SAVE	Save all parameter settings in the memory module safe against mains failure		

Control elements	
	Execute the function assigned to the function key (see LCD display)
stop	Execute the stop function set in C00469 (Lenze setting: Inhibit inverter)
run	Deactivate stop function again (Lenze setting: Enable inverter again)
	In the menu level: Select menu/submenu In the parameter level: Select parameter In the editing mode: Change marked digits or select list entry
CINH	In the menu level: Select submenu/change to parameter level In the editing mode: Cursor to the right
RDY	In the menu level: One menu level higher (if available) In the parameter level: Back to the menu level In the editing mode: Cursor to the left

Menu structure

In the keypad, the parameters are classified into various menus and submenus.

- The **USER menu** includes a selection of frequently used parameters.
- The **Code list** contains all parameters.
- The **Go to param** function enables you to reach the corresponding parameter directly.
- The **Logbook** logs all errors and their chronological history.
- The **Diagnostics** menu contains diagnostic/display parameters for displaying device-internal process factors, current actual values and status messages.

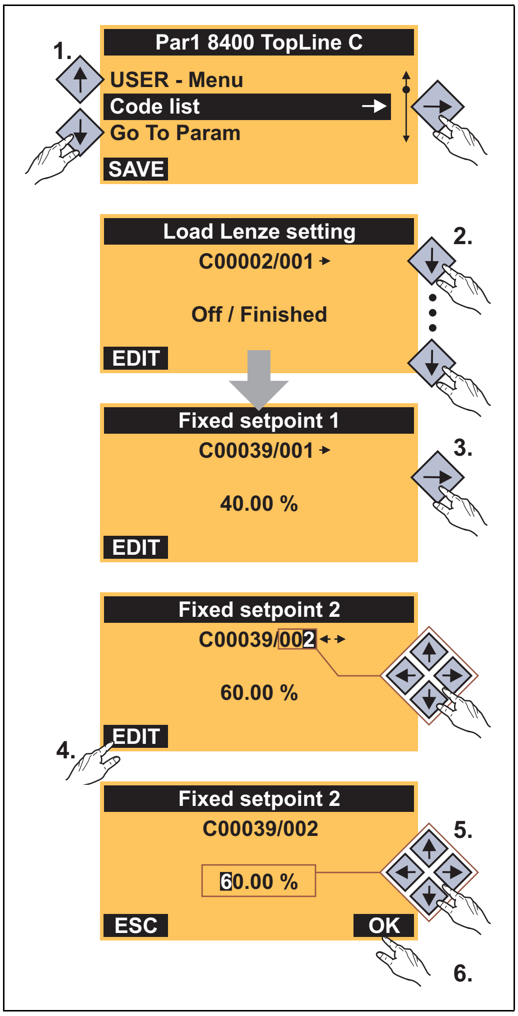
User level

From version 12.00.00 onwards, the extent of menus, submenus an codes shown in the keypad can be adapted by selecting the "Userlevel" in C00001/1:

- **Standard** user level (Lenze setting): Only the most important menus and codes are displayed in the keypad.
- **Expert** user level: All menus and codes are displayed in the keypad.
- **Service** user level: Only for the purpose of service (Lenze Service).

After changing the user level, the menus in the keypad are restructured according to the selected user level. The parameters of plugged-in communication module are always shown completely independent of the set user level.

General operation



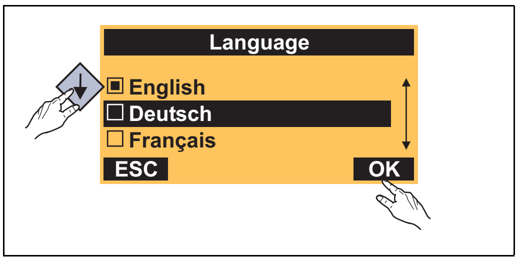
1. Use the ⇧/⇩ navigation keys to select the desired menu.
   - Use the ⇦/⇨ navigation keys to reach a higher/lower menu level.
   - Use the ▤ function key to return to the main menu.
2. Use the ⇧/⇩ navigation keys to select the parameter to be set within a submenu.
3. In order to select another subcode in case of a parameter with subcodes:
   - Press the navigation key ⇨ to change to the editing mode for the subcode.
   - Use the navigation keys to set the desired subcode.
4. Use the EDIT function key to switch over to the editing mode.
5. Use the navigation keys to set the desired value.
6. Use the OK function key to accept the change and to leave the editing mode.
   - Use the ESC function key to leave the editing mode without accepting the change.

[2-2] Example: Changing parameters with the keypad

Multilingualism

All texts displayed in the keypad are in English.

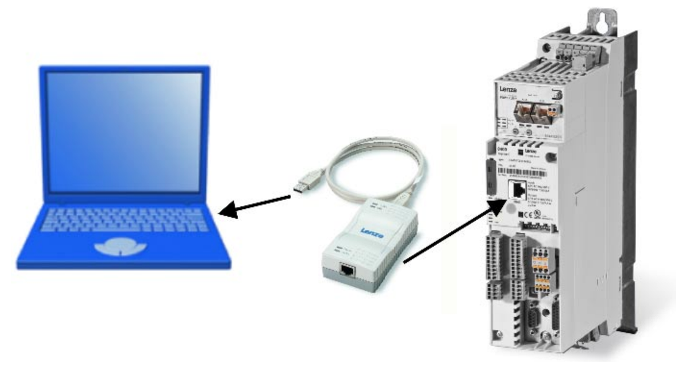
From version 02.00.00 onwards, the most important menus as well as diagnostic and configuration parameters can are also available in German and French. To set a different language, select the **Language selection** menu item in the main menu of the keypad.



- The multilingual texts are stored in the inverter and do not have to be loaded into the device.
- For reasons of disc space, only the most important menus and parameters as well as the error messages are available in several languages.

2.3.2 Change parameter settings with PC and Lenze software

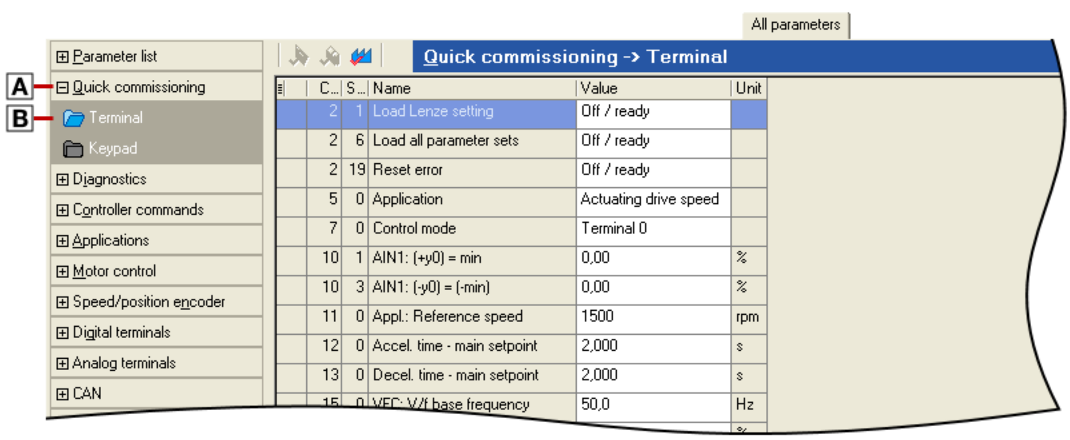
The USB diagnostic adapter, for instance, can be used for the communication between the PC (including the »EASY Starter« or » Engineer« software) and the inverter, see the following illustration. The USB diagnostic adapter is the connection between the PC (free USB port) and the inverter (X6 diagnostic interface).



[2-3] Exemplary constellation for parameterising the inverter

The **All parameters** tab in the »EASY Starter« and the »Engineer« provides a quick access to all parameters of the inverter.

The given categories and subcategories correspond 1:1 to the menus and submenus of the keypad:



A Category
B Subcategories

[2-4] **All parameters** tab in the »Engineer«

Moreover, the »Engineer« provides a commissioning interface on the **Application parameters** tab where you can commission the application in a few steps.

Detailed information on how to handle the »Engineer« can be found in the integrated online help that you can call with the **[F1]** function key.

2.3.3 Save parameter settings in the memory module safe against mains failure

Inverter parameter changes via the EASY Starter /»Engineer«, the keypad, or a master control via fieldbus communication will be lost after mains switching of the controller unless the settings have been explicitly saved to the integrated memory module.

General information

- In the delivery state, the Lenze setting of the parameters has been saved to the integrated memory module. These parameters are
  - the parameters of the inverter
  - the parameters of the communication module plugged into the MCI interface
  - the parameters of the possibly existing safety module (device variant)
- When the device or the external 24 V voltage supply is switched on, all parameters are automatically loaded from the memory module into the main memory of the inverter.
- Full functionality of the memory module is even provided if the power supply has been switched off and only the electronic components of the inverter are externally supplied by a 24 V DC voltage, e.g. via the X4/24E terminal.
- The memory module can be preconfigured with customised data.
- The memory module is available as a spare part - without any data.

During operation

- Parameter sets can be saved and loaded manually.
- Using the keypad, you can press function key SAVE to save the parameter settings.
- The »EASY Starter«/»Engineer« serves to execute the saving via the icon in the *toolbar* or via the device command "Save all parameter sets" (C00002/11 = "1: On / start").
  - The storage process may take a couple of seconds. After the device command has been called in C00002/11, dynamic status information ("Work in progress 20%" ➜ "Work in progress 40%" ➜ "Work in progress 60%", etc.) is returned.

Note!

In order to prevent data inconsistencies during the saving process:
- Do not switch off the supply voltage!
- Do not remove the memory module from the device!

Automatic saving of changed parameter settings is explicitly not supported because this significantly reduces the service life of the memory module.

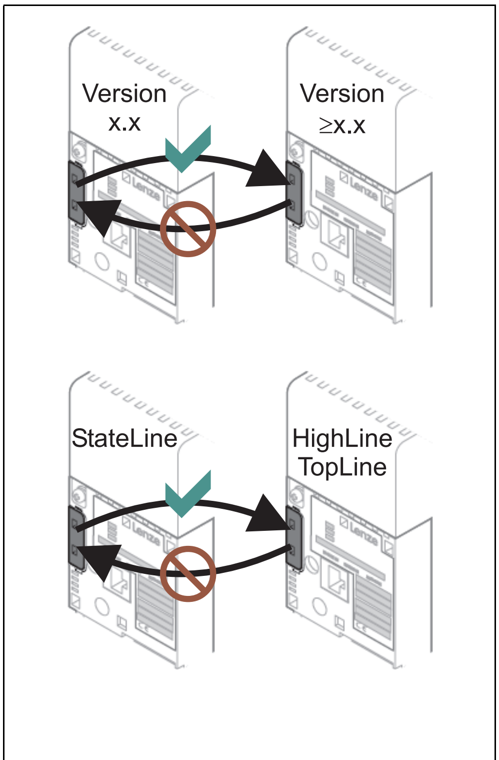
Unplugging the memory module

The memory module is hot-pluggable. A removal during operation causes a warning "PS01: No memory module" and should thus be avoided. The behaviour of the device, however, remains unchanged as all parameters are available in the RAM memory after the device has been started. The device can also be parameterised when the memory module has been unplugged. In this case, the parameter sets cannot be saved in the memory module.

Replacement of the inverter

In the event of a device replacement, the entire parameter data of an axis can be copied to the replacement device by "taking along" the memory module, so that additional PC or keypad operations are not required.

When replacing the inverter, the versions of the old and new device are of importance. Before data are actually transferred, the versions are checked internally. As a general principle, the following applies:



- Parameter sets of old devices can only be processed on new devices with the same or higher device version (downward compatibility).
- Parameter sets of devices with versions that have less functions (e.g. 8400 StateLine) can be loaded into and executed on devices with versions that have more functions (e.g. 8400 HighLine). The reverse is not possible!
- If the parameter set stored in the memory module is incompatible with the standard device, the "PS03: Par.set device invalid" error message is output and the inverter changes to the "Fault" device state.
- If the parameter set stored in the memory module is compatible with the standard device but has a different (lower) version, the "PS03: Par.set device invalid" message is only output as "Information". The message can be eliminated by storing the parameter set again.
  Note: If you save the parameter set in a higher device version, you can no longer load this parameter set to a lower device version.

Example of parameter set compatibilities:

Parameter version on memory module	Device version			
	StateLine V12	HighLine V12	HighLine V13	TopLine V12
StateLine V12	OK	PS03 (Information)	PS03 (Information)	PS03 (Information)
HighLine V12	PS03 (Fault)	OK	PS03 (Information)	PS03 (Information)
HighLine V13	PS03 (Fault)	PS03 (Fault)	OK	PS03 (Fault)
TopLine V12	PS03 (Fault)	PS03 (Fault)	PS03 (Fault)	OK

2.3.4 User menu for quick access to frequently used parameters

When a system is installed, parameters must be changed time and again until the system runs satisfactorily. The user menu of the inverter contains a selection of frequently used parameters to be able to access and change these parameters quickly:

Parameters	Name	Lenze setting
C00051	MCTRL: Actual speed value	-
C00053	DC-bus voltage	-
C00054	Motor current	-
C00061	Heatsink temperature	-
C00137	Device status	-
C00166/3	Mess. - status det. error	-
C00011	Appl.: Reference speed	1500 rpm
C00039/1	Preset setpoint 1	40.00 %
C00039/2	Preset setpoint 2	60.00 %
C00012	Acceleration time - main setpoint	2.000 s
C00013	Deceleration time - main setpoint	2.000 s
C00015	VFC: V/f base frequency	50 Hz
C00016	VFC: Vmin boost	1.60 %
C00022	Imax in motor mode	depending on the device power
C00120	Setting of motor overload (I^2xt)	100.00 %
C00087	Rated motor speed	1460 rpm
C00099	Firmware version	-
C00200	Firmware product type	-
C00105	Decel. time - quick stop	2.000 s
C00173	Mains voltage	0: "3ph 400V / 1ph 230V"
Greyed out = display parameter		

Tip!

The user menu can be freely configured in C00517.

In the »Engineer«, you can configure the user menu comfortably via the **User menu** tab (see »Engineer« online help).

The password protection serves to restrict the access to parameters of the user menu. Then, all other parameters cannot be accessed without knowing the password and are thus protected against unwanted changes.

2.4 Device access protection

Various tasks can be executed via the functions of the device access protection:

- Password protection
  - Only authorised persons (with password knowledge) may read/change all parameters of the inverter.
  - Non-authorised persons (without password knowledge) can only access the max. 32 parameters of the user menu.
- Device personalisation
  - Only inverters and memory modules personalised with a specific binding ID can be used in the system.

Note!

If password protection/device personalisation is used:

- Inform the end customer that Lenze can only provide restricted service for the devices with access protection.
- It is not possible for Lenze to modify a replacement device via special accesses in such a way that it cooperates with a personalised memory module.
- From firmware version 4.2 onwards, the X400 keypad supports the alphanumeric entry of a password. Keypads with lower firmware versions cannot be used for the entry.

2.4.1 Password protection

When the password protection is active, only write/read access to the parameters of the user menu is possible.

- From version 15.00.00, the password protection can be configured individually for every single communication channel. ▶ Individual password protection for single communication channels (📖 52)
- The following describes how to set/check/delete a password by means of the parameters relevant for these functions.
- From »Engineer« V2.14 onwards. these functions can also be executed via dialog (menu command **Online → Set/check/delete password**).

> **Note!**
>
> The password protection serves to only limit the access to parameters of the standard device. The access to parameters of a plugged-in communication module is not concerned.

Short overview of the relevant parameters for password protection:

Parameters	Info		Lenze setting
C00505/3	Password • The password must have a length of 4 ... 16 characters. • The password may consist of any character. However, this is not sensible. Recommended characters are: lower case letters (a - z), upper case letters (A - Z) and digits (0 - 9). **Note:** After the execution of one of the device commands listed below, this parameter provides the current password status:		
	off	No password is set, password protection is not active (Lenze delivery status).	
	on	Password is set, password protection is active. • This status is also displayed if checking/deleting the password has not been successful due to an invalid entry.	
	ok	Password is set, password protection is not active. • The password protection is temporarily deactivated.	
Device commands Before the following device commands are executed, enter the corresponding password in C00505/3.			
C00002/31	Set password ▶ Activate the password protection		0: Off / ready
C00002/32	Check password ▶ Temporarily deactivate the password protection		0: Off / ready
C00002/33	Delete password ▶ Deactivate password protection/change password		0: Off / ready
Greyed out = display parameter			

Parameters	Info		Lenze setting
Status displays			
C00003	Status of the last device command		-
C00507/1	Password protection - all communication channels • Bit coded display of the active protective functions:		-
	Bit 0	Only access to user menu	
	Bit 1	Parameter write protection	
	Bit 2	Parameter read protection	
	Bit 3 ... 14	Reserved	
	Bit 15	Memory module binding on	
	Note: As the password protection can be configured individually for each single communication channel from version 15.00.00 bit 1 and bit 2 indicate the active protection always with regard to the communication channel used.		
Greyed out = display parameter			

Activate the password protection

The password protection is activated by setting a password.

How to set a password:

1. Enter the desired password in C00505/3.
   - The password must have a length of 4 ... 16 characters.
   - The password may consist of any character. However, this is not sensible. Recommended characters are: lower case letters (a - z), upper case letters (A - Z) and digits (0 - 9).
2. Execute "Set password" device command: C00002/31 = "1: On / start"
   - After successful execution, password status ON is displayed in C00505/3 and password protection takes immediate effect.

Temporarily deactivate the password protection

The "Check password" device command serves to temporarily deactivate the password protection in order to execute password-protected functions.

- The password protection remains deactivated until
  - an invalid password will be entered and checked
    - or -
  - the external 24-V supply of the control electronics is switched off (< 19 V).

How to temporarily deactivate the active password protection:

1. Enter the set password in C00505/3.
2. Execute "Check password" device command C00002/32 = "1: On / start"
   - After a successful check, password status OK is displayed in C00505/3.

Deactivate password protection/change password

The password protection is simply activated by deleting the set password. If you want to change the set password, first delete the set password as well. Then set the new password.

How to delete the set password:

1. Enter the set password in C00505/3.
2. Execute "Delete password" device command C00002/33 = "1: On / start"
   - After a successful deletion, password status OFF is displayed in C00505/3.

2.4.2 Individual password protection for single communication channels

This function extension is available from version 15.00.00!

The 8400 TopLine can be parameterised via the following "communication channels":

- Internally via system blocks (LS_ParReadWrite1-6 and LS_WriteParamList)
- externally via
  - Diagnostics interface X6 ("DIAG")
  - CANopen interface X1 ("CAN on board")
  - Fieldbus interface (MCI)

As in some applications, only certain communication channels have to be protected, the parameters described in the following serve to configure individual access restrictions for each communication channel.

In the Lenze setting, all access restrictions are active for all communication channels when the password is set. This corresponds to the previous behaviour with a set password, i.e. only one write/read access to the parameters of the user menu is possible.

Note!

The configuration can only be changed if no password has been set!
- This means that even if the password has been deactivated temporarily by the "check password" device command, the configuration cannot be changed.
- As long as a password is set, the "Load Lenze setting" device command does not change the configuration either.

The password protection is activated by setting a password.

▶ Activate the password protection (📖 51)

▶ Deactivate password protection/change password (📖 52)

Short overview of the relevant parameters for configuring the password protection:

When the password is set, the respective restriction is active if the corresponding bit has been set:

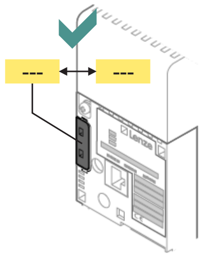
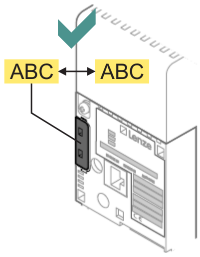
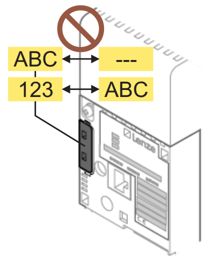
Parameters	Info		Lenze setting
C00506/1	PW protection internal config		0x0007
	Bit 0	Only access to user menu • The menus in the keypad are reduced. • Only the user menu can be used to change parameters.	
	Bit 1	Parameter write protection • System blocks cannot be used to write into write/read parameters.	
	Bit 2	Parameter read protection • System blocks cannot be used to read write/read parameters.	
	Bit 3 ... 15	Reserved	
C00508/1	PW protection config. X6 (DIAG)		0x0006
	Bit 0	Reserved	
	Bit 1	Parameter write protection • The diagnostics interface X6 cannot be used to write into write/read parameters.	
	Bit 2	Parameter read protection • The diagnostics interface X6 cannot be used to read write/read parameters.	
	Bit 3 ... 15	Reserved	
C00509/1	PW protection config. X1 (CAN)		0x0006
	Bit 0	Reserved	
	Bit 1	Parameter write protection • The CANopen interface X1 cannot be used to write into write/read parameters.	
	Bit 2	Parameter read protection • The CANopen interface X1 cannot be used to read write/read parameters.	
	Bit 3 ... 15	Reserved	
C00510/1	PW protection config. MCI		0x0006
	Bit 0	Reserved	
	Bit 1	Parameter write protection • The fieldbus interface (MCI) cannot be used to write into write/read parameters.	
	Bit 2	Parameter read protection • The fieldbus interface (MCI) cannot be used to read write/read parameters.	
	Bit 3 ... 15	Reserved	

The parameter set must be saved to the device safe against mains failure to prevent password protection configured in the device from getting lost due to mains switching.

▶ Save parameter settings in the memory module safe against mains failure (📖 46)

2.4.3 Device personalisation

The inverter and the memory module are married via the device personalisation by means of a binding ID. When the device personalisation is active, all write/read actions between inverter and memory module are only executed if both components have the same binding ID.

Lenze delivery status:	Procedure carried out by the customer:	Impermissible replacement by the end user:
		
No binding ID is set.	Customer sets binding ID for device personalisation.	When device personalisation is active: The replacement of the inverter or memory module causes an error message if the binding ID is incorrect or not available.

If, for instance, a parameter set has been loaded from the memory module with an active device personalisation, saving of this parameter set on another memory module with a different or non-existent binding ID is not possible.

- Thus, copying the parameter set from a personalised memory module to a non-personalised memory module is not possible!

Two types of checks are distinguished:

- If a differing binding ID is detected when the inverter is switched on (during device initialisation):
  - the "Fault" error response is returned.
  - The "PS10: Invalid memory module binding" error message is entered into the logbook.
- If a differing binding ID is detected while a device command for loading/saving the parameter set is executed:
  - Loading/saving is not executed.
  - A corresponding status for the device command is output in C00003.

Short overview of the relevant parameters for device personalisation:

Parameters	Info		Lenze setting
C00505/2	Binding ID • The binding ID must have a length of 4 ... 16 characters. • The binding ID may consist of any character. However, this is not sensible. Recommended characters are: lower case letters (a - z), upper case letters (A - Z) and digits (0 - 9). **Note:** After the execution of one of the device commands listed below, this parameter provides the current binding ID status:		
	off	No binding ID is set.	
	on	Binding ID is set.	
Greyed out = display parameter			

Parameters	Info		Lenze setting
Device commands Before the following device commands are executed, enter the corresponding binding ID in C00505/2.			
C00002/29	Set binding ID ▸ Activate device personalisation		0: Off / ready
C00002/30	Delete binding ID ▸ Deactivate device personalisation/change binding ID		0: Off / ready
Status displays			
C00003	Status of the last device command		-
C00507/1	Password protection - all communication channels • Bit coded display of the active protective functions:		-
	Bit 0	Only access to user menu	
	Bit 1	Parameter write protection	
	Bit 2	Parameter read protection	
	Bit 3 ... 14	Reserved	
	Bit 16	Memory module binding on	
Greyed out = display parameter			

Activate device personalisation

The device personalisation is activated by setting a binding ID.

How to set the binding ID:

1. Enter the desired binding ID in C00505/2.
   - The binding ID must have a length of 4 ... 16 characters.
   - The binding ID may consist of any character. However, this is not sensible. Recommended characters are: lower case letters (a - z), upper case letters (A - Z) and digits (0 - 9).
2. Execute "Set binding ID" device command C00002/29 = "1: On / start"
   - After successful execution, status ON is displayed in C00505/2.

Deactivate device personalisation/change binding ID

The device personalisation is simply deactivated by deleting the set binding ID. If you want to change the set binding ID, first delete the set binding ID as well. Then set the new binding ID.

How to delete the binding ID:

1. Enter the set binding ID in C00505/2.
   - If inverter and memory module do not have the same binding ID, enter the binding ID of the memory module to delete the binding ID of both components.
2. Execute "Delete binding ID" device command C00002/30 = "1: On / start"
   - After a successful deletion, status OFF is displayed in C00505/2.

2.4.4 Unlocking the inverter with a MasterPin

Every inverter has an individual master password called "MasterPin". By entering the MasterPin, an inverter inhibited by the password mechanisms can be reset to the delivery status.

Stop!

When the MasterPin is used, the parameter set is reset to the Lenze setting both in the inverter and in the memory module!

- This results in a permanent loss of the customised parameterisation that must be recreated!
- A reset to the Lenze setting can result in unforeseen level changes at the I/O terminals (e.g. brake control)!

How to restore the delivery status:

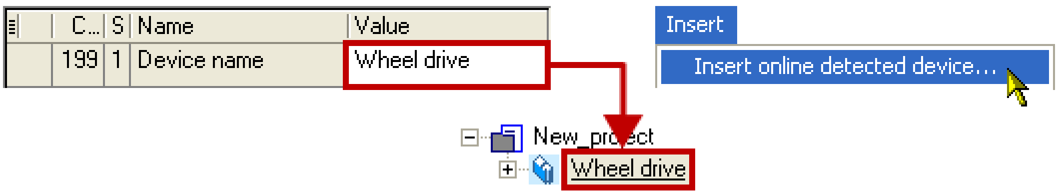
1. Inhibit the inverter if it is enabled, e.g. via the C00002/16 device command.
2. Enter the MasterPin in C00505/1.
   - The MasterPin comprises the last 6 digits of the serial number of the memory module.
3. Execute "Check MasterPin" device command C00002/28 = "1: On / start"

2.5 Device identification

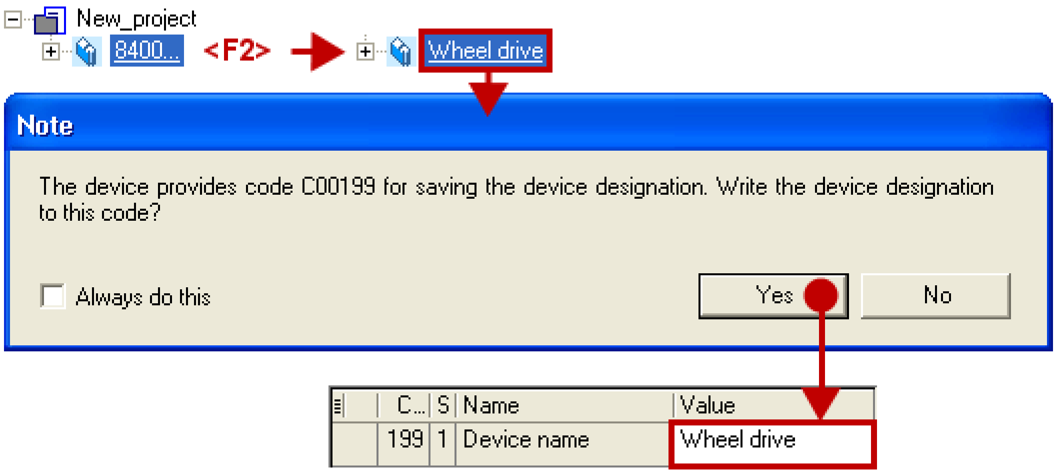
For device identification, any device name (e.g. wheel drive) with max 32 characters can be set in C00199/1 for the inverter and saved in the memory module with mains failure protection.

2.5.1 Automatic acceptance of the device name in the »Engineer«

If a device name is assigned in C00199/1 and the inverter in the »Engineer« is added to the project via the **Insert → Insert device detected online...** function, the device name stored in C00199/1 (here: wheel drive) is used as device designation in the *Project view* instead of the type (8400 TopLine):



This mechanism also functions in reverse direction:
If you rename the inverter in the *project view* via **<F2>**, you will be asked afterwards if you want to take over the changed name in C00199/1:



2.5.2 Extended item designation

This function extension is available from version 12.00.00!

In C00199/2...5, four texts à max. 32 characters can be set for item designation and saved in the memory module safe against mains failure. These additional subcodes are not dependent on the previously described text acceptance mechanism in the »Engineer« for the device name (C00199/1).

3 Commissioning

Danger!

Unexpected motor movements can occur

Under certain conditions the motor may rotate after mains connection.

Possible consequences:

- Persons in the vicinity of the machine or plant risk getting hurt.
- Unexpected starting action may damage the machine or plant.

Protective measures:

- Commissioning with external 24 V supply and without mains voltage. In this case, the inverter can only be parameterised and diagnosed during commissioning.
- Ensure that setpoints are not active.

Tip!

- Information on some of the operating statuses can quickly be obtained via the LED status displays on the front of the inverter. (📖 720)
- **Check firmware:** Particularly with regard to the use of an older inverter (e.g. if the customer is using one from stock) it makes sense to check the software (firmware) version. The software version of the inverter can be seen on the nameplate in the "SW" line and can be determined by reading out code C00099.
- **Restore delivery status:** Set code C00002/1 to "1: On / start" to reset all parameter settings of the device to the Lenze setting. This leaves you with a defined device configuration. ▶ Load Lenze setting (📖 110)

The following chapters describe the commissioning of the available technology applications with the »Engineer«.

Information on how to commission using the keypad (or diagnosis terminal) can be found in the **8400 hardware manual**. The hardware manual has been stored in electronic form on the data carrier supplied with the 8400 inverter.

3.1 Safety instructions with regard to commissioning

General safety instructions

In order to prevent injury to persons or damage to material assets

- check before connecting the mains voltage
  - the wiring for completeness, short circuit, and earth fault
  - the "emergency stop" function of the entire system
  - that the motor circuit configuration (star/delta) is adapted to the output voltage of the inverter
  - the in-phase connection of the motor
- check the setting of the most important drive parameters before enabling the controller:
  - the V/f rated frequency must be adapted to the motor circuit configuration!
  - the drive parameters relevant for your application must be set correctly!
  - the configuration of the I/O terminals must be adapted to the wiring!
- ensure that there are no active speed setpoints before enabling the controller.

Safety instructions with regard to motor operation

Danger!

- For thermal reasons, continuous operation of self-ventilated motors at a low field frequency and rated motor current is not permissible!
  - In the Lenze setting, the Motor temperature monitoring (PTC) is activated. (🕮 313)
  - Activate the Brake resistor monitoring (I2xt) if necessary. (🕮 314)
- C00015 must be used to select 87 Hz operation if a delta-connected asynchronous motor (nameplate data: 400 V ⅄ /230 V △) is to be operated in conjunction with an inverter for a mains voltage of 400 V.

3.2 Notes on motor control

In the Lenze setting, the V/f characteristic control (VFCplus) as motor control is set in C00006 with a linear characteristic.

- V/f characteristic control (VFCplus) is a motor control mode for classic frequency inverter applications on the basis of a simple and robust control procedure for the operation of machines with a linear or quadratic load torque characteristic (e.g. fans).
- The presettings of the parameters ensure that the inverter is immediately ready for operation and the motor works adequately without further parameterisation if an inverter and a 50 Hz asynchronous machine with matching performances are assigned to each other.

Note!

Check the nameplate data against the motor data set in the inverter. Further information is provided in the chapter "Motor selection/Motor data". (🕮 144)

Recommendations for the following application cases:

- If the inverter and motor differ greatly in terms of performance:
  Set the Imax limit (in motor mode) in C00022 to 2x rated motor current.
- If a high starting torque is required:
  When the motor is idling, set a value for Vmin boost in C00016 which ensures that the rated motor current flows at a field frequency of f = 3 Hz (display in C00058).
- For noise optimisation:
  In C00018, set a switching frequency of "16 kHz var./drive-opt.".
- If a high torque must be provided at small speeds without feedback:
  Select "Sensorless vector control (SLVC) as motor control mode in C00006.

Related topics:

▶ Motor control (MCTRL) (🕮 143)

3.3 Preconditions for commissioning with the »Engineer«

For commissioning, you need

- a PC that meets the following system requirements:
  - a processor with 1.4 GHz or higher
  - at least 512 MB RAM and 650 MB free hard disc space
  - Microsoft® Windows® 2000 operating system (from service pack 2 onwards) or Windows® XP
- the Lenze »Engineer« PC software
- a connection to the inverter, e.g. via a USB diagnostic adapter:
  - connect the USB diagnostic adapter to the X6 diagnostic interface.
  - establish a connection between the USB diagnostic adapter and the PC via a free USB port.

Tip!

How to obtain/update the »Engineer« software:

- **Download from the internet:** The full version of the »Engineer StateLevel« is provided free of charge in the internet:
  http://www.Lenze.com → Download → Software downloads
- **Requesting the CD**You can also request the »Engineer« separately on CD free of charge at your Lenze representative. See the "About Lenze" area on our homepage for e.g. the corresponding German address.

3.4 Trouble-shooting during commissioning

When the »Engineer« is used, trouble during commissioning can be detected and eliminated conveniently. Proceed as follows:

- Check whether error messages appear in the »Engineer«.
  - On the **Diagnostics** tab, relevant actual states of the inverter and pending error messages are displayed in a well-arranged visualisation.
- Check the input terminals for their corresponding setpoints.
  - The **Terminal assignment** tab displays the current input/output signals.
- Check the signal flow of the application.
  - For this purpose, click the **Signal flow** button on the **Application parameter** tab. The displayed signal flow shows active setpoints and their further processing.

Related topics:

▶ Diagnostics & error management (📖 719)
▶ LED status displays (📖 720)
▶ Error messages of the operating system (📖 743)

3.5 Commissioning wizard 8400

This function extension is supported by the »Engineer« from version 2.15 onwards!

The **commissioning wizard 8400** serves to carry out a guided commissioning of the inverter based on the Lenze setting of the parameters. The set parameters can then be saved in the inverter with mains failure protection.

Note!

Take all the necessary safety precautions before you carry out the following commissioning steps and switch the device on!

▶ Safety instructions with regard to commissioning (📖 59)

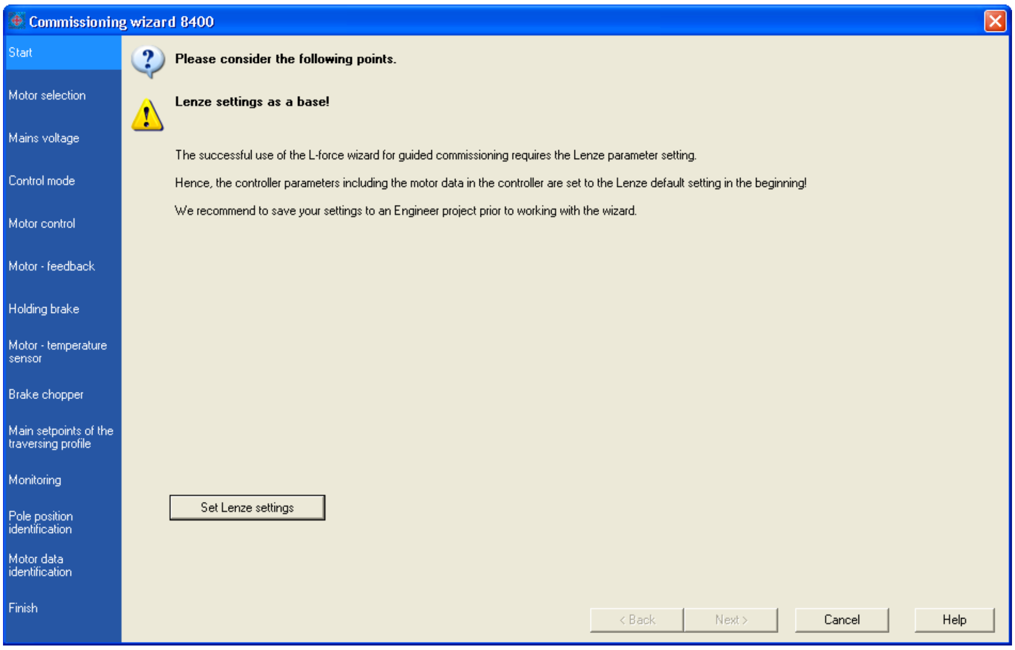
How to carry out a guided commissioning using the »Engineer«:

1. Go to the *Project view* and select the 8400 TopLine inverter.
2. Go online.

   After a connection to the inverter has been established, the following status is displayed in the *Status line*:

3. Click the icon to open the *commissioning wizard 8400* dialog box.
   - Now the commissioning wizard guides you step by step through the setting of the important parameters for a quick commissioning.
   - The **Next** button can only be activated again after all parameter settings in the device have been reset via the **Load Lenze setting** button.



Related topics:

▶ Commissioning of the "Actuating drive speed" technology application (📖 65)

3.6 Manual motor direction of rotation check (manual control)

The »Engineer« function "Manual control" easily allows for controlling the direction of rotation of the motor. The manual control serves to let the motor connected to the inverter with an adjustable speed for an adjustable time.

Note!

Take all the necessary safety precautions before you carry out the following commissioning steps and switch the device on!

▶ Safety instructions with regard to commissioning (📖 59)

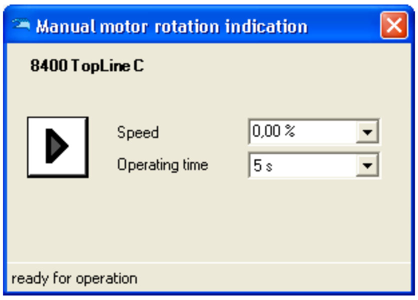
How to carry out a manual motor direction of rotation check using the »Engineer«:

1. Go to the *Project view* and select the 8400 TopLine inverter.
2. Go online.

   After a connection to the inverter has been established, the following status is displayed in the *Status line*:

   ONLINE
3. Click the symbol to inhibit the inverter via device command.
4. Ensure that the following conditions are met:
   - The mains voltage is switched on.
   - No trouble is active.
   - Safe torque off (STO) is not active.
5. Enable inverter via terminal: Set terminal X5/RFR to HIGH level.
6. Click the icon to open the *Manual motor rotation indication* dialog box.

   Note: If the "not ready for operation" status is displayed, check whether all conditions mentioned before (see steps 3 ... 5) have been met.



7. Set the desired speed and runtime.
   (The speed refers to the reference speed set in C00011.)
8. Click the button to let the motor rotate with the set speed for the set runtime.

   By clicking the button, the function can be aborted.

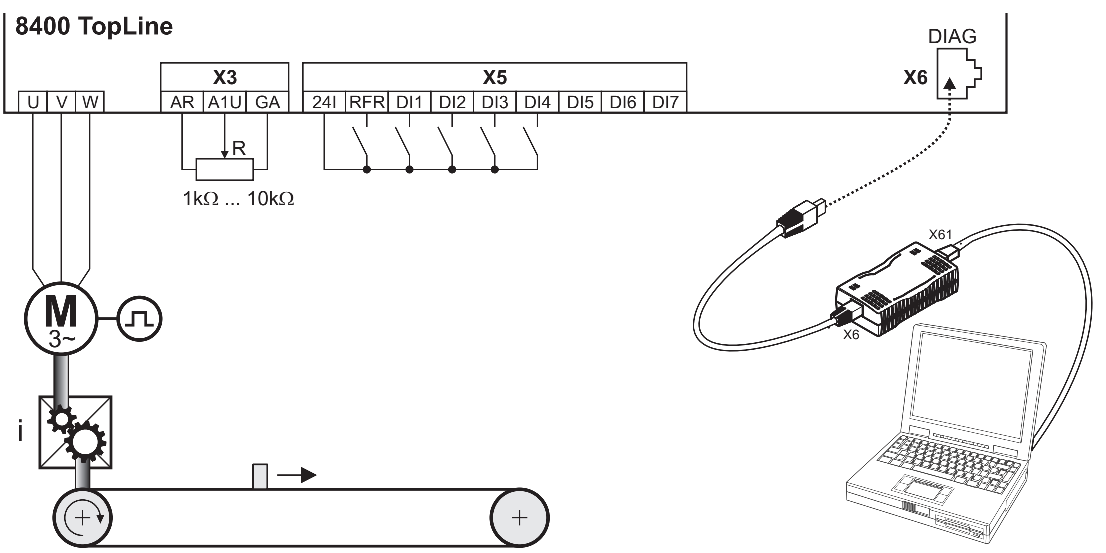
3.7 Commissioning of the "Actuating drive speed" technology application

Note!

Take all the necessary safety precautions before you carry out the following commissioning steps and switch the device on!

▶ Safety instructions with regard to commissioning (📖 59)

System constellation



X3/AR	Reference voltage (10 V) for analog signals
X3/A1U	Speed setpoint selection (slider of setpoint potentiometer R) • Scaling: 10 V ≡ 100 % ≡ 1500 rpm[1] (for a 4-pole motor)
X3/GA	Ground potential (GND) for analog signals
X5/RFR	Controller enable
X6	Diagnostic interface for connecting the USB diagnostic adapter

[3-1] Block diagram for wiring the commissioning example for the "Actuating drive speed" application

Commissioning steps

Find a description of the commissioning steps of the "Actuating drive speed" technology application below.

Please observe the sequence of the steps in the following chapters and follow them through carefully. This will help you to commission your inverter quickly and as safely as possible:

▶ Prepare inverter for commissioning (📖 66)
▶ Creating an »Engineer« project & going online (📖 67)
▶ Parameterising the motor control (📖 68)
▶ Parameterise application (📖 69)

▶ Save parameter settings safe against mains failure (📖 71)
▶ Enable inverter and test application (📖 71)

3.7.1 Prepare inverter for commissioning

1. Power terminal wiring.
   Refer to the mounting instructions supplied with the inverter to find help on how to correctly design the power connections to match the requirements of your device.
2. Wire the control terminals.
   The assignment for your digital inputs should correspond to one of the preconfigured control modes (C00007) for terminal control:

Control mode	Assignment of the digital terminals DI1	DI2	DI3	DI4
Terminals 0	JOG 1/3	JOG 2/3	DCB	Cw/Ccw
Terminals 2	JOG 1/3	JOG 2/3	QSP	Cw/Ccw
Terminals 11	Cw/Ccw	DCB	MPotUp	MPotDown
Terminal 16	JOG 1/3	JOG 2/3	Cw/QSP	Ccw/QSP

Abbreviations used:

JOG	Selection of fixed setpoints 1 ... 3 parameterised in C00039/1...3
DCB	Manual DC-injection braking
Cw/Ccw	CW/CCW rotation
QSP	Quick stop
MPotUp	Motor potentiometer: Increase speed
MPotDown	Motor potentiometer: Reduce speed
Cw/QSP	Fail-safe selection of the direction of rotation in connection with quick stop
Ccw/QSP	

3. Inhibit inverter: Set terminal X5/RFR to LOW level or open contact.
4. Connect USB diagnostic adapter.
5. Switch on voltage supply of the inverter.
   - Without motor operation: Connect external 24 V supply.
   - With motor operation: Connect mains voltage.

If the green "DRV-RDY" LED is blinking and the red "DRV-ERR" LED is off, the inverter is ready for operation and commissioning can proceed.

Related topics:

▶ Automatic restart after mains connection/fault... (📖 130)
▶ LED status displays (📖 720)

3.7.2 Creating an »Engineer« project & going online

You can find detailed information on the general use of the »Engineer« in the online help which you can call with **[F1]**.
- The chapter "Working with projects" describes, among other things, all options of the *Start-up wizard* which are available to create a new »Engineer« project.

The following steps serve to describe a general method for creating a project with the **Select component from catalogue** option. For this purpose, individual components (inverter, motor, etc.) are selected from selection lists.

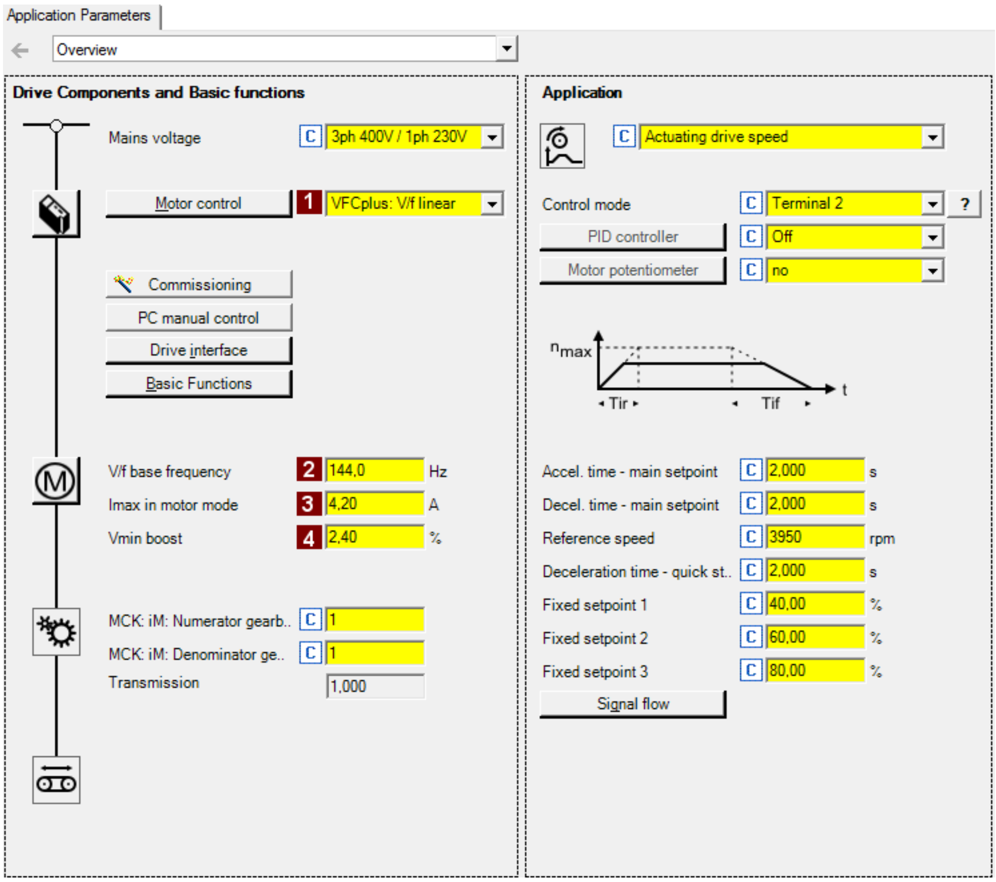
1. Start »Engineer«.
2. Create a new project with the *Start-up wizard* and the **Select component from catalogue** option:
   - In the **Component** step, select the 8400 TopLine inverter.
   - In the **Device modules** step, select the available communication module.
   - In the **Application** step, select the "Actuating drive speed" application. (The application can also be selected any time afterwards via the **Application parameter** tab or C00005.)
   - Select the other components (motor/gearbox) to be added to the project in the **Other components** dialog step.
3. Go online.
   After a connection to the inverter has been established, the following status is displayed in the *Status line*:

4. Transfer parameter set to the device.
   This command serves to overwrite the current parameter settings in the inverter by parameter settings of the »Engineer« project.

3.7.3 Parameterising the motor control

1. Select the **Application parameters** tab from the *Workspace*.
   The motor control parameters, among other things, can be found on the left:



2. In the **1** **Motor control** list field (C00006), select the desired motor control.
3. Adapt the motor control parameters:

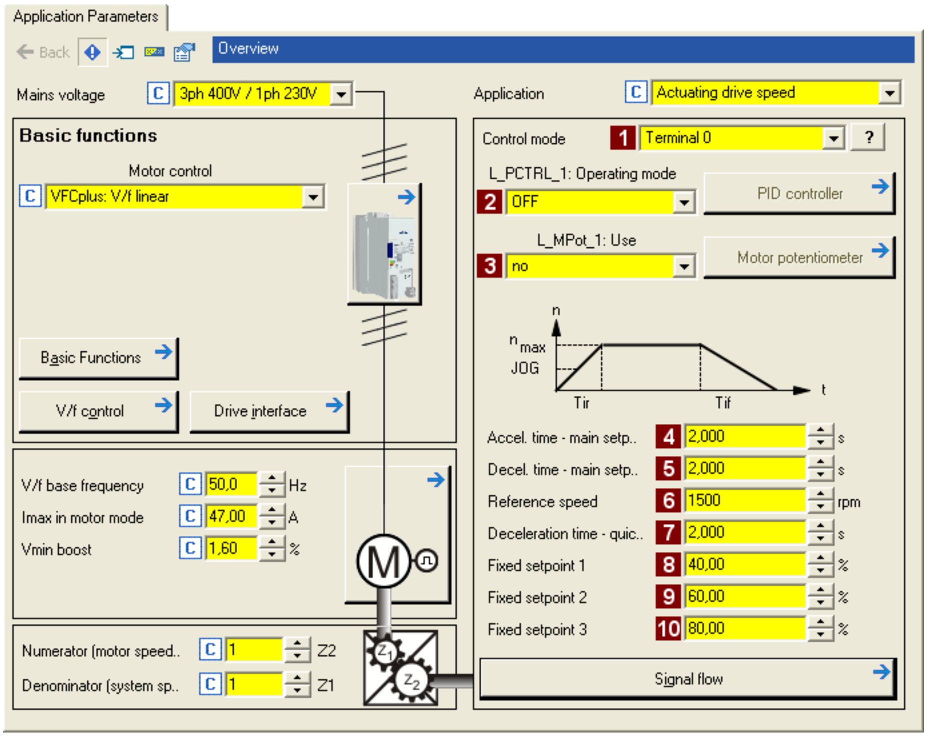
	Parameters	Lenze setting Value	Unit	Info
2	**V/f base frequency** (C00015)	50.0	Hz	▸ Adapting the V/f base frequency (📖 174)
3	**Imax in motor mode** (C00022)	47.00	A	▸ Optimising the Imax controller (📖 178)
4	**Vmin boost** (C00016)	1.60	%	▸ Adapting the Vmin boost (📖 176)

Related topics:

- ▸ Notes on motor control (📖 60)
- ▸ Motor control (MCTRL) (📖 143)

3.7.4 Parameterise application

The application parameters can be found on the right side of the **Application parameter** tab:



1. In the **1** **Control mode** list field (C00007), select the control mode suitable for the wiring of the terminals.
   - The corresponding wiring diagram is displayed in a pop-up window if you click the **?** button right to the list field.
   - For a detailed description, see the chapter "Terminal assignment of the control modes". (📖 465)
2. Optional: Use process controller.
   - For this purpose, select the desired operating mode in the **2** **L_PCTRL_1: Operating mode** list field (C00242).
   - For a detailed description see the L_PCTRL_1 function block. (📖 1695)
   - Go to the parameterisation dialog of the process controller via the **Process controller** button.
3. Optional: Use motor potentiometer.
   - For this purpose, select "1: On" in the **3** **L_MPot_1: Use** list field (C00806).
   - For a detailed description see the L_MPot_1 function block. (📖 1650)
   - Go to the parameterisation dialog of the motor potentiometer via the **Motor potentiometer** button.

4. Adapt the application parameters:

	Parameters	Lenze setting Value	Unit	Info
4	**Accel. time - main setpoint** (C00012)	2.000	s	The setpoint is led via a ramp function generator with linear characteristic. The ramp function generator converts setpoint step-changes at the input into a ramp. ▶ L_NSet_1 (📖 1668)
5	**Decel. time - main setpoint** (C00013)	2.000	s	
6	**Reference speed** (C00011)	1500	rpm	All speed setpoint selections are provided in % and always refer to the reference speed set in C00011. The motor reference speed is indicated on the motor nameplate.
7	**Decel. time - quick stop** (C00105)	2.000	s	If quick stop is requested, motor control is decoupled from the setpoint selection and, within the deceleration time parameterised in C00105, the motor is brought to a standstill (n_{act} = 0). ▶ Activate/deactivate quick stop (📖 114)
8	**Preset setpoint 1** (C00039/1)	40.00	%	A fixed setpoint for the setpoint generator can be activated instead of the main setpoint via the digital DI1 and DI2 inputs. • Fixed setpoints are selected in [%] based on the reference speed (C00011). ▶ L_NSet_1 (📖 1668)
9	**Preset setpoint 2** (C00039/2)	60.00	%	
10	**Preset setpoint 3** (C00039/3)	80.00	%	

Tip!

- Click the **Signal flow** button to go down one dialog level to the signal flow of the application with further possible parameter settings. See chapter "Basic signal flow". (📖 455)
- The preconfigured I/O connection in the selected control mode can be changed via configuration parameters. See chapter "User-defined terminal assignment". (📖 445)

More detailed information on the technology application:

- ▶ TA "Actuating drive speed" (📖 454)
- ▶ Internal interfaces | application block "LA_NCtrl" (📖 457)
- ▶ Process data assignment for fieldbus communication (📖 474)
- ▶ Terminal assignment of the control modes (📖 465)
- ▶ Setting parameters (short overview) (📖 476)
- ▶ Configuration parameters (📖 478)

3.7.5 Save parameter settings safe against mains failure

In order to prevent parameter settings carried out in the device from being lost by mains switching, you have to explicitly save the parameter set with mains failure protection in the device.

Save parameter set.

3.7.6 Enable inverter and test application

Stop!

Before stipulating a speed setpoint, check whether the brake in the form of a holding brake on the motor shaft has been released!

Note!

If the controller is enabled at mains connection and C00142 has activated the "Inhibit at device on" auto-start option (Lenze setting), the inverter remains in the "ReadyToSwitchOn" state.

To be able to change to the "SwitchedOn" status, the controller enable must be deactivated first: set terminal X5/RFR to LOW level.

If the inverter is in the "SwitchedOn" state:

1. Enable inverter: Set terminal X5/RFR to HIGH level or close contact.
   - If there is no other active source for the controller inhibit, the inverter changes from the "SwitchedOn" status to the "OperationEnabled" status.
   - The **Diagnostics** tab and C00158 display all active sources for the controller inhibit.
2. Select the speed setpoint.
   In the "Terminal 0" control mode by selecting a voltage at the analog input via the setpoint potentiometer or by selecting a fixed setpoint via the digital DI1/DI2 inputs:

DI1	DI2	Speed selection
LOW	**LOW**	The setpoint speed is selected via analog input 1 • Scaling: 10 V ≡ 100 % ≡ reference speed (C00011)
HIGH	LOW	Fixed setpoint 1 (C00039/1) is used as setpoint speed. • Lenze setting: 40 % of the reference speed (C00011)
LOW	HIGH	Fixed setpoint 2 (C00039/2) is used as setpoint speed. • Lenze setting: 60 % of the reference speed (C00011)
HIGH	HIGH	Fixed setpoint 3 (C00039/3) is used as setpoint speed. • Lenze setting: 80 % of the reference speed (C00011)

Note!

Observe the actual speed value (display in C00051) as well as the LED status displays.
(📖 720)

Tip!

Other control functions in the "Terminal 0" control mode:

- DI3: HIGH level ≡ Request DC-injection braking
- DI4: HIGH level ≡ Request a change of direction of rotation

Related topics:

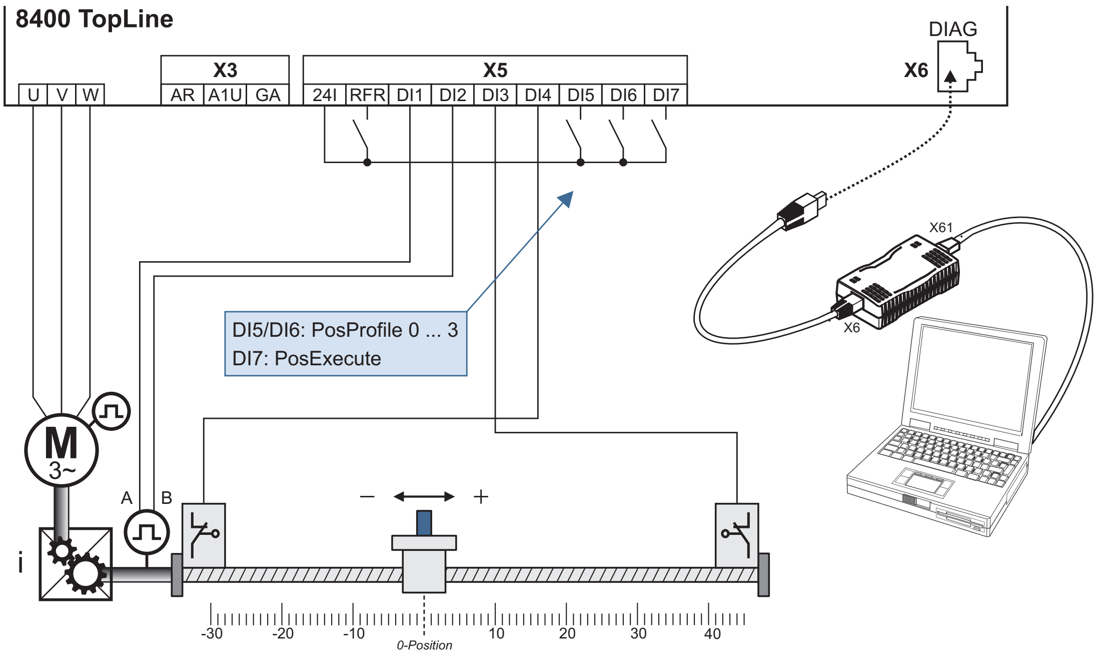
- "Inhibit at device on" auto-start option (📖 130)
- Trouble-shooting during commissioning (📖 62)
- Diagnostics & error management (📖 719)

3.8 Commissioning of the "Table positioning" technology application

Note!

Take all the necessary safety precautions before you carry out the following commissioning steps and switch the device on!

▶ Safety instructions with regard to commissioning (📖 59)

System constellation	
	
X5/RFR	Controller enable
X5/DI1	Position encoder connection, track A
X5/DI2	Position encoder connection, track B
X5/DI3	Hardware limit switch connection in positive direction
X5/DI4	Hardware limit switch connection in negative direction
X5/DI5/DI6	Profile selection and change of operating mode at the same time: 0 | 0 = Profile selection 0 and request for "Speed follower" operating mode 0 | 1 = Profile selection 1 and request for "Homing" operating mode 1 | 0 = Profile selection 2 and request for "Manual jog" operating mode 1 | 1 = Profile selection 3 and request for "Positioning" operating mode
X5/DI7	Start of profile generation
X6	Diagnostic interface for connecting the USB diagnostic adapter

[3-2] Block diagram for wiring the commissioning example for the "Table positioning" application

Commissioning steps

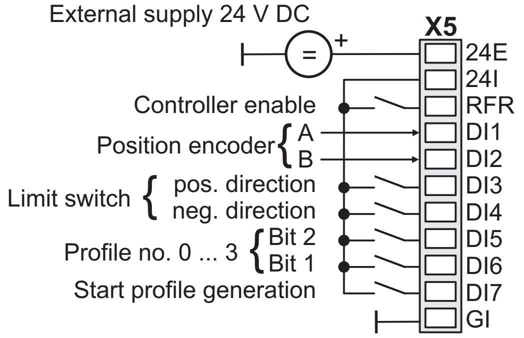
Below find a description of the commissioning steps of the "Table positioning" application shown in illustration [3-2].

Please observe the sequence of the steps in the following chapters and follow them through carefully. This will help you to commission your inverter quickly and as safely as possible:

- Prepare inverter for commissioning (🕮 75)
- Creating an »Engineer« project & going online (🕮 76)
- Parameterising the motor control (🕮 77)
- Parameterise application (🕮 78)
- Save parameter settings safe against mains failure (🕮 86)
- Enable inverter and test application (🕮 86)

3.8.1 Prepare inverter for commissioning

1. Power terminal wiring.
   Refer to the mounting instructions supplied with the inverter to find help on how to correctly design the power connections to match the requirements of your device.
2. Wire the control terminals.
   The system constellation including position encoder and hardware limit switches shown in illustration [3-2] requires wiring according to control mode "Terminals 0":



3. Inhibit inverter: Set terminal X5/RFR to LOW level or open contact.
4. Connect USB diagnostic adapter.
5. Switch on voltage supply of the inverter.
   - Without motor operation: Connect external 24 V supply.
   - With motor operation: Connect mains voltage.

If the green "DRV-RDY" LED is blinking and the red "DRV-ERR" LED is off, the inverter is ready for operation and commissioning can proceed.

Related topics:

▸ Automatic restart after mains connection/fault... (📖 130)
▸ LED status displays (📖 720)

3.8.2 Creating an »Engineer« project & going online

You can find detailed information on the general use of the »Engineer« in the online help which you can call with **[F1]**.
- The chapter "Working with projects" describes, among other things, all options of the *Start-up wizard* which are available to create a new »Engineer« project.

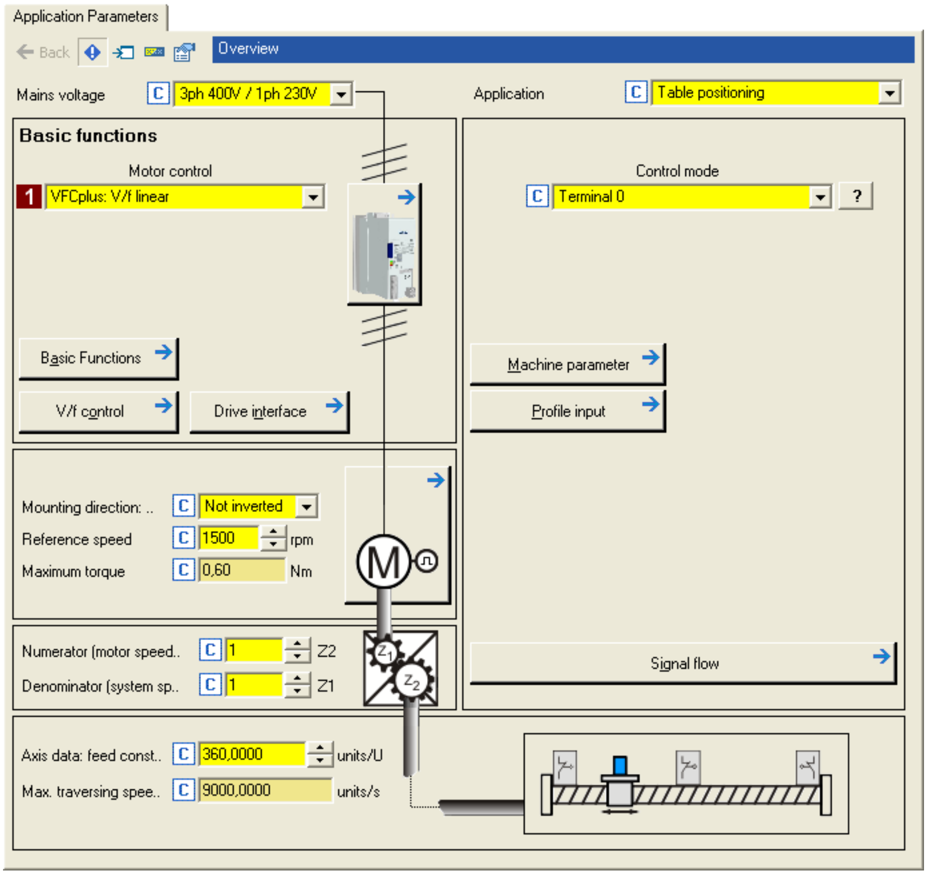
The following steps serve to describe a general method for creating a project with the **Select component from catalogue** option. For this purpose, individual components (inverter, motor, etc.) are selected from selection lists.

1. Start »Engineer«.
2. Create a new project with the *Start-up wizard* and the **Select component from catalogue** option:
   - In the **Component** step, select the 8400 TopLine inverter.
   - In the **Device modules** step, select the available communication module.
   - In the **Application** step, select the "Table positioning" application. (The application can also be selected any time afterwards via the **Application parameter** tab or C00005.)
   - Select the other components (motor/gearbox) to be added to the project in the **Other components** dialog step.
3. Go online.
   After a connection to the inverter has been established, the following status is displayed in the *Status line*:

4. Transfer parameter set to the device.
   This command serves to overwrite the current parameter settings in the inverter by parameter settings of the »Engineer« project.

3.8.3 Parameterising the motor control

1. Select the **Application parameters** tab from the *Workspace*.
   The motor control parameters, among other things, can be found on the left:



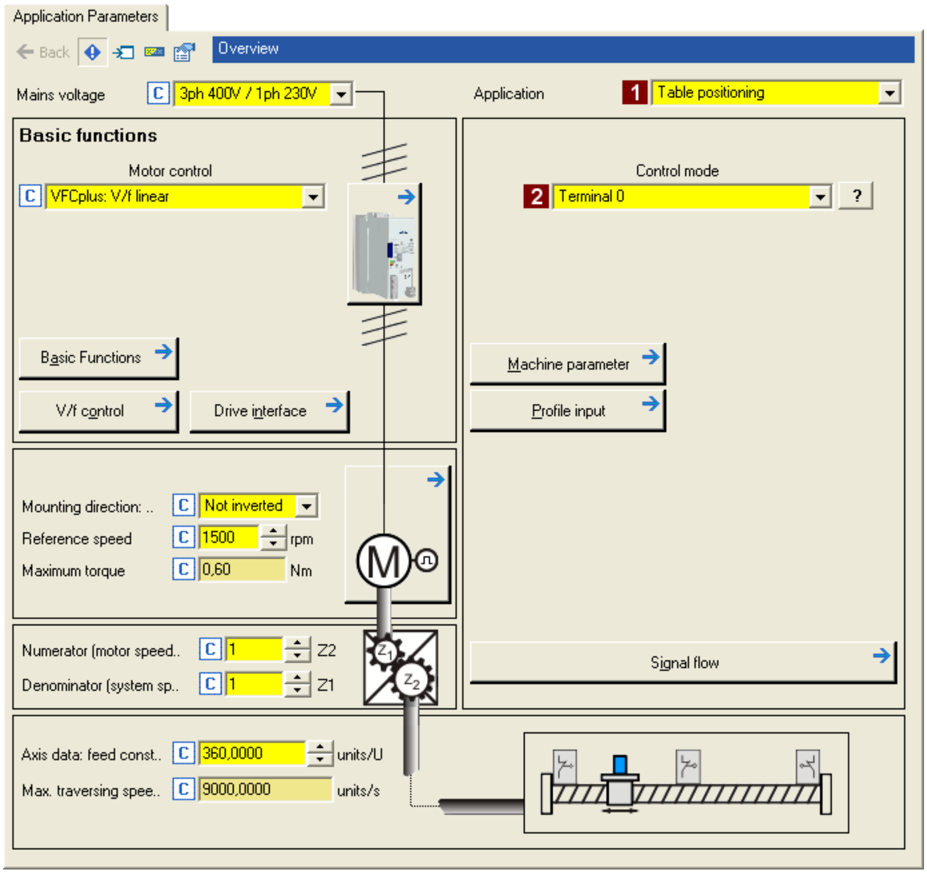
2. In the **1** **Motor control** list field (C00006), select the desired motor control.

Related topics:

- Notes on motor control (📖 60)
- Motor control (MCTRL) (📖 143)

3.8.4 Parameterise application

The application parameters can be found on the right side of the **Application parameter** tab:



1. Select "Table positioning" in the **1** **Application** list field (C00005) (if you have not already done so while creating the project).
   After the "Table positioning" application is selected, the contents of the tab change, e.g. the **Machine parameters** and **Profile entry** buttons are shown now.

2. In the **2** **Control mode** list field (C00007), the preset "Terminals 0" control mode is already suitable for the system constellation shown in illustration [3-2] and need not be changed.

   - The corresponding wiring diagram is displayed in a pop-up window if you click the ? button right to the list field.
   - For a detailed description, see the chapter "Terminal assignment of the control modes". (465)

3.8.4.1 Set machine parameters

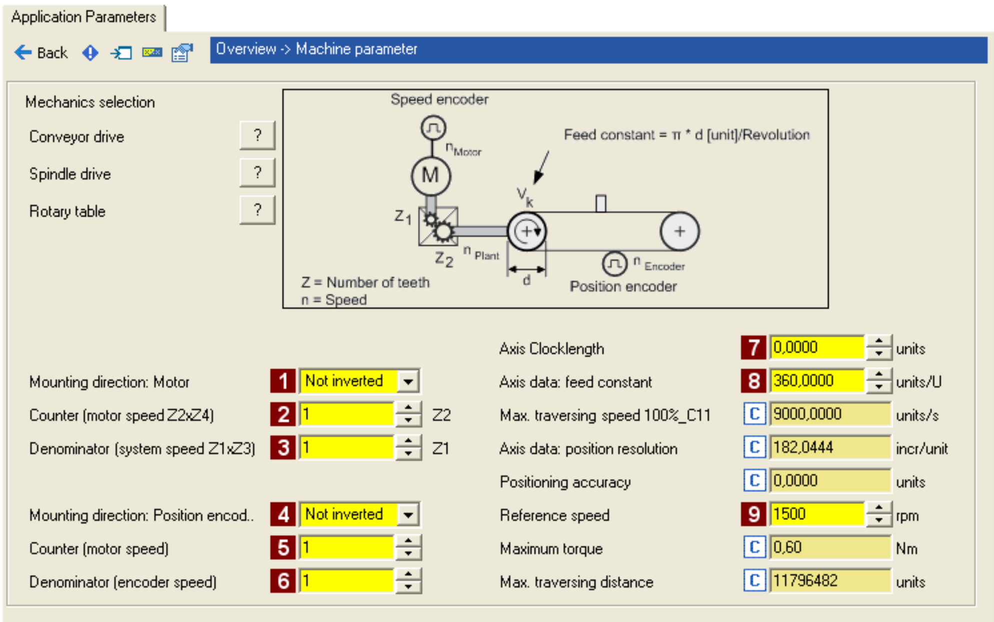
Note!

Setting the machine parameters is a basic prerequisite for the operating modes "Homing", "Manual jog" and "Positioning".

The more precisely the machine parameters are set, the better the results of positioning!

Detailed information on the machine parameters is provided in chapter "Basic drive functions" in subchapter "Machine parameters". (📖 611)

1. Go to the right side of the **Application parameter** tab and click the **Machine parameter** button to change to the *Overview* ➞ *Machine parameter* dialog level:



2. Set the machine parameters according to the system constellation at hand.

	Parameters	Lenze setting Value	Unit	Info
1	**Mounting direction: Motor** (C01206/1)	Not inverted		Inversion if motor mounting is mirrored.
2	**Numerator (Z2)** (C01202/1)	1		Gearbox factor - motor • Entry of the gearbox factor as numerator/denominator ratio (numerator = motor speed and denominator = output speed of gearbox) or from the number of teeth of the gearbox arrangement.
3	**Denominator (Z1)** (C01202/2)	1		
4	**Mounting direction: Position encoder** (C01206/2)	Not inverted		Inversion if position encoder mounting is mirrored.
5	**Counter** (C01203/1)	1		Gearbox factor - position encoder • Entry of the gearbox factor as numerator/denominator ratio, with numerator = motor speed and denominator = position encoder speed.
6	**Denominator** (C01203/2)	1		

Parameters		Lenze setting Value	Unit	Info
7	**Axis cycle** (C01201/1)	0.0000	units	Cycle for Modulo measuring system • The Modulo system is activated by setting a cycle (C01201/1) > 0 units. • When the cycle (C01201/1) is set to 0 units (Lenze setting), the traversing range is unlimited (classical measuring system).
8	**Feed constant** (C01204)	360.0000	units/rev.	The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft. • The value is entered in application units referred to one revolution.
9	**Reference speed** (C00011)	1500	rpm	All speed setpoint selections are provided in % and always refer to the reference speed set in C00011. The motor reference speed is indicated on the motor nameplate.

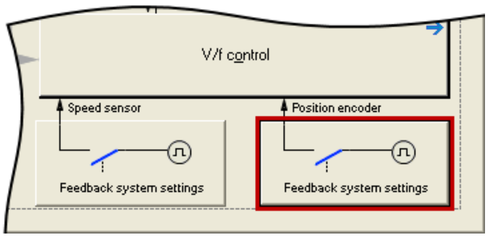
3. After setting the machine parameters, click the **Back** button to change to the *Overview* dialog level.

3.8.4.2 Set the position encoder

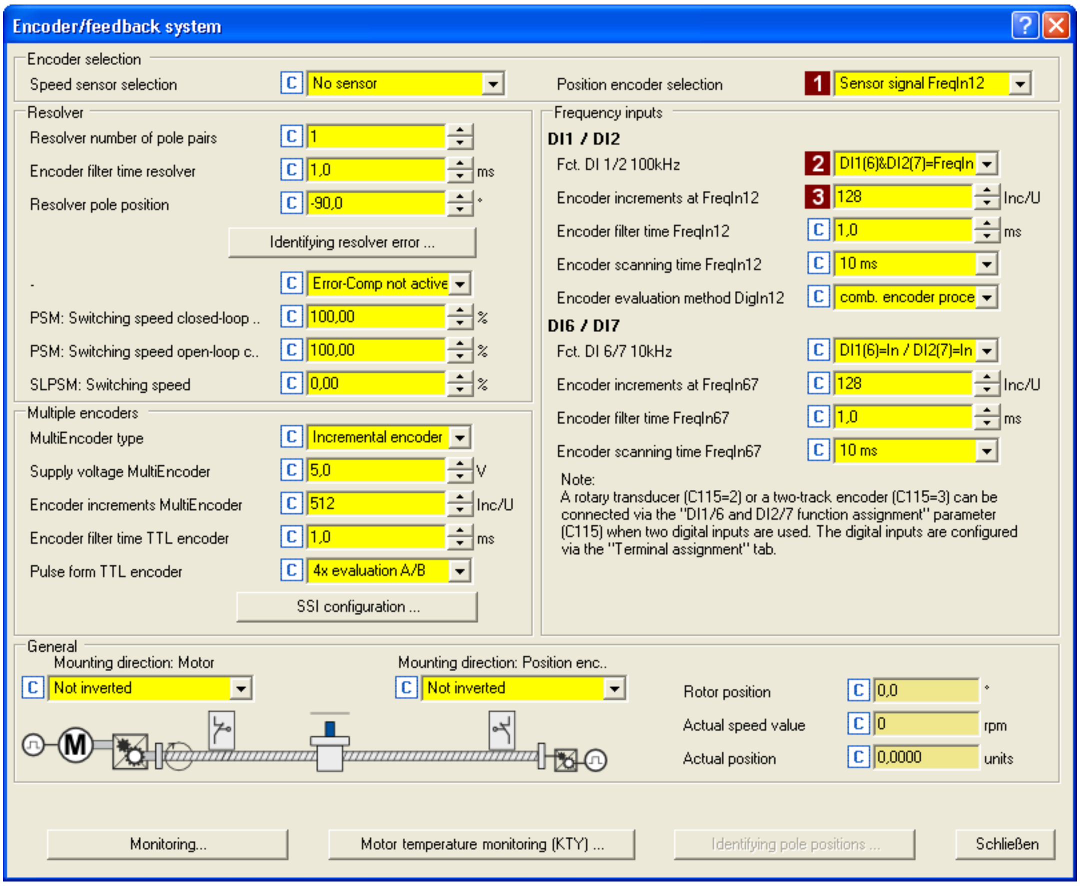
In the Lenze setting, the digital input terminals are configured as "normal" digital inputs. Since in this system constellation DI1 and DI2 are used to connect a two-track position encoder, the function assignment of these two inputs must be changed accordingly.

Detailed information on how to use a resolver/encoder as motor or position encoder can be found in the main chapter "Encoder/feedback system". (📖 330)

1. Go to the right side of the **Application parameter** tab and click the **Signal flow** button to change to the *Overview* → *Signal flow* dialog level.
2. Click the **Feedback system settings** button in the signal flow:



The *Encoder/Feedback system* parameterisation dialog will be shown:



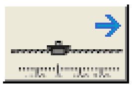
3. In the **1** **Position encoder selection** list field (C00490), select "Encoder signal FreqIn12".
4. In the **2** **Fct. DI 1/2 100kHz** list field (C00115/1), select "DI1(6)&DI2(7)=FreqIn (2-track)".
   This selection ensures that the digital DI1 and DI2 input terminals are configured as frequency inputs.
5. Set the number of position encoder increments in the **3** **Number of encoder increments at FreqIn12** input field (C00420/1).
6. Click the **Close** button to close the parameterisation dialog again.

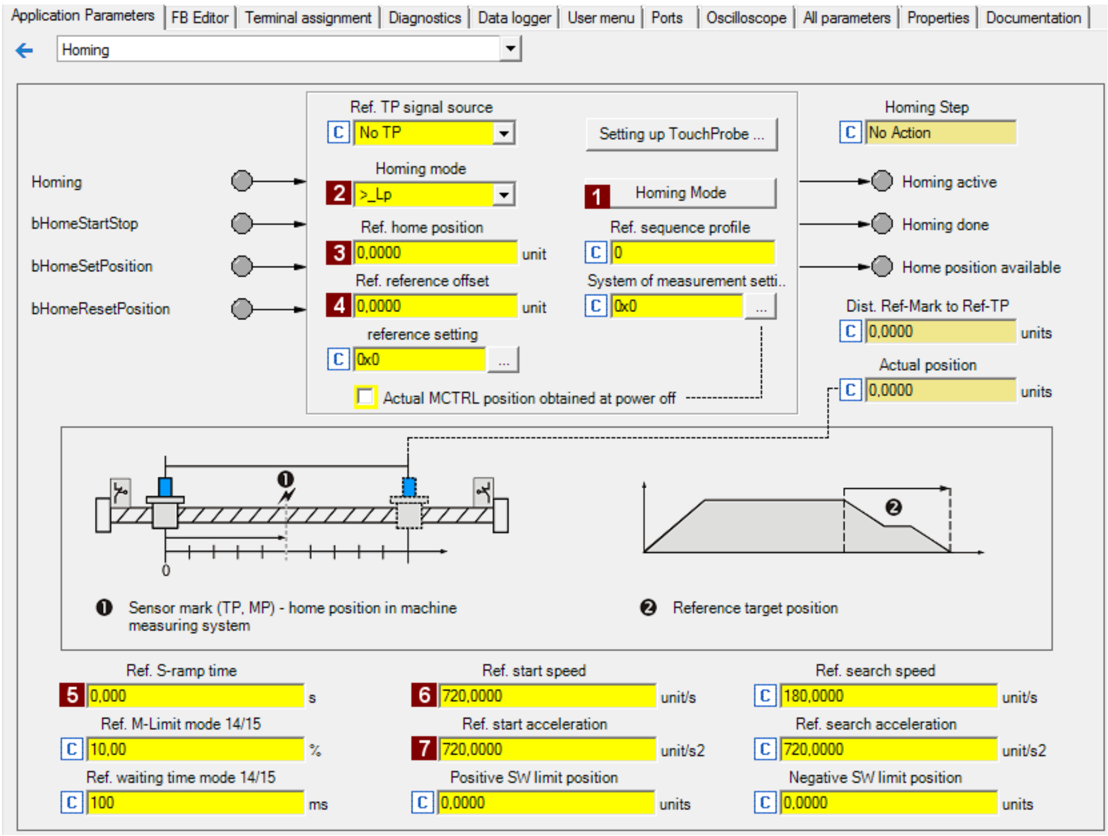
Related topics:

- Digital input terminals (📖 401)
- Using DI1(6) and DI2(7) as frequency inputs (📖 405)
- Encoder/feedback system (📖 330)

3.8.4.3 Set homing

Detailed information on the "Homing" operating mode is provided in chapter "Basic drive functions" in subchapter "Homing". (📖 637)

1. Click the  button (in the "MotionControlKernel" block) in the signal flow to change to the *Overview* → *Signal flow* → *Homing* dialog level:



2. Select the homing mode and hence the type of homing via the **1** **Homing mode** button or directly in the **2** **Homing mode** list field (C01221).
   - A graphic display of all possible homing modes can be seen via the **Homing mode** button.
   - In this system constellation without touch probe, the preset ">_Lp" mode can be used: movement in positive direction, positive edge of the limit switch sets reference.
3. Set the position to be used for loading the set position and the actual position after homing has finished in the **3** **Homing home position** input field (C01227/2).

4. Recommendation: Set an offset in the 4 **Homing home offset** input field (C01227/1) to prevent the machine from stopping at an activated limit switch.
   Background: The load machine can also leave the travel range limit switch. There follows a return to the home position that was set with the positive edge of the travel range limit switch.
5. Set suitable profile parameters for the homing:

Parameters		Lenze setting		Info
		Value	Unit	
5	**S-ramp time** (C01226/1)	0.000	s	S-ramp time for reference search/homing. • Setting "0 s" ≡ no rounding
6	**Start speed** (C01224/1)	720.0000	unit/s	Starting speed for approaching the pre-stop signals.
7	**Start acceleration** (C01225/1)	720.0000	unit/s^2	Starting acceleration for the starting speed ramps.

6. After setting the parameters for the homing, click the **Back** button to change back to the *Overview* → *Signal flow* dialog level.

3.8.4.4 Enter one or more profiles

You can find detailed information on entering profiles in the chapter entitled "Basic drive functions", subchapter "Profile entry". (📖 671)

1. Click the **Profile entry** button (in the "MotionControlKernel" block) in the signal flow to change to the *Overview* → *Signal flow* → *Profile entry* dialog level:

Note!

In the Lenze setting, the profiles are assigned to certain operating modes, i.e. the selection of a profile also results in a change of operating mode:

- If profile 0 is selected: Activation of "Speed follower" operating mode
- If profile 1 is selected: Activation of "Homing" operating mode
- If profile 2 is selected: Activation of "Manual jog" operating mode
- If profile 3 ... 15 is selected: Activation of "Positioning" operating mode

2. Select number 3 in the 1 **Profile number** list field to enter a positioning profile which can be selected via digital inputs DI5 and DI6.

3. Set the profile parameters:

	Parameters	Lenze setting Value	Unit	Info
2	**Profile x: Mode** (C01300/x)	absolute (shortest way)t		Selection of the way in which positioning is to be carried out. ▶ Positioning modes (📖 677)
3	**Profile x: Position** (C01301/x)	360.0000	unit	Target position or distance to be traversed.
4	**Profile x: Speed** (C01302/x)	360.0000	unit/s	Maximum velocity at which the target is to be approached.
5	**Profile x: Acceleration** (C01303/x)	720.0000	unit/s^2	Specification of the maximum velocity variation which is to be used for acceleration.
6	**Profile x: Deceleration** (C01304/x)	720.0000	unit/s^2	Specification of the velocity variation which is to be used for maximum deceleration to standstill.
7	**Profile x: S-ramp time** (C01306/x)	0.000	s	Due to stipulation of an S-ramp time for a profile, the profile is executed with S-shaped ramps, i.e. acceleration and braking processes are initiated smoothly in order to reduce jerk and thus the stress on the drive components.
8	**Profile x: Final speed** (C01305/x)	0.0000	unit/s	This specifies the velocity at which the drive is to start the next profile after reaching the target position.
9	**Profile x: Sequence profile** (C01307/x)	0		Optional: Sequence profile for profile linkage / following block control. After execution of the profile (target position reached), the set following (subsequent) profile is started automatically. In this way, profile chains can be stipulated without additional control processes.
10	**Profile x: TP profile** (C01308/x)	0		Optional: Profile number of the profile (1 ... 15) that is to be executed after a touch probe has been detected. • If "0" is set, there will be no profile stepping through touch probe. • Only relevant for positioning modes with touch-probe.
11	**Profile x: TP source** (C01308/x)	TP-DigIn3		Optional: Selection of the signal source for touch probe detection. • Only relevant for positioning modes with touch-probe.

4. After entering the profile, click the **Back** button to change back to the *Overview → Signal flow* dialog level.

Related topics:

▶ Touch probe detection (📖 435)

▶ Operating mode change with profile number (📖 607)

▶ Positioning (📖 666)

More detailed information on the technology application:

- TA "Table positioning" (📖 514)
- Basic signal flow (📖 515)
- Internal interfaces | application block "LA_TabPos" (📖 519)
- Process data assignment for fieldbus communication (📖 538)
- Terminal assignment of the control modes (📖 529)
- Setting parameters (short overview) (📖 540)
- Configuration parameters (📖 541)

3.8.5 Save parameter settings safe against mains failure

In order to prevent parameter settings carried out in the device from being lost by mains switching, you have to explicitly save the parameter set with mains failure protection in the device.

Save parameter set.

3.8.6 Enable inverter and test application

Stop!

Before stipulating a speed setpoint, check whether the brake in the form of a holding brake on the motor shaft has been released!

Note!

If the controller is enabled at mains connection and C00142 has activated the "Inhibit at device on" auto-start option (Lenze setting), the inverter remains in the "ReadyToSwitchOn" state.

To be able to change to the "SwitchedOn" status, the controller enable must be deactivated first: set terminal X5/RFR to LOW level.

If the inverter is in the "SwitchedOn" state:

1. Enable inverter: Set terminal X5/RFR to HIGH level or close contact.
   - If there is no other active source for the controller inhibit, the inverter changes from the "SwitchedOn" status to the "OperationEnabled" status.
   - The **Diagnostics** tab and C00158 display all active sources for the controller inhibit.
2. Select the respective control signals via the digital inputs (see sections below).

Note!

Observe the actual speed value (display in C00051) as well as the LED status displays. (📖 720)

Homing

1. Request "homing" mode:
   Set digital input DI5 to LOW level and digital input DI6 to HIGH level in order to select the profile 1 and request the "homing" mode at the same time.
2. Start homing:
   Set digital input DI7 to HIGH level ("alternative function PosExecute").
3. Complete homing when home position has been reached:
   Reset digital input DI7 to LOW level.

Positioning

1. Request "positioning" mode:
   Set both digital inputs DI5 and DI6 to HIGH level to select the profile 3 and request the "positioning" mode at the same time.
2. Start positioning:
   Set digital input DI7 to HIGH level ("alternative function PosExecute").
3. Complete positioning if target position has been reached:
   Reset digital input DI7 to LOW level.

Tip!

The active operating mode (C01243) in the signal flow can also be controlled by means of the setting of the switch in the "Motion Control Kernel" block.

Related topics:

- "Inhibit at device on" auto-start option (130)
- Trouble-shooting during commissioning (62)
- Diagnostics & error management (719)

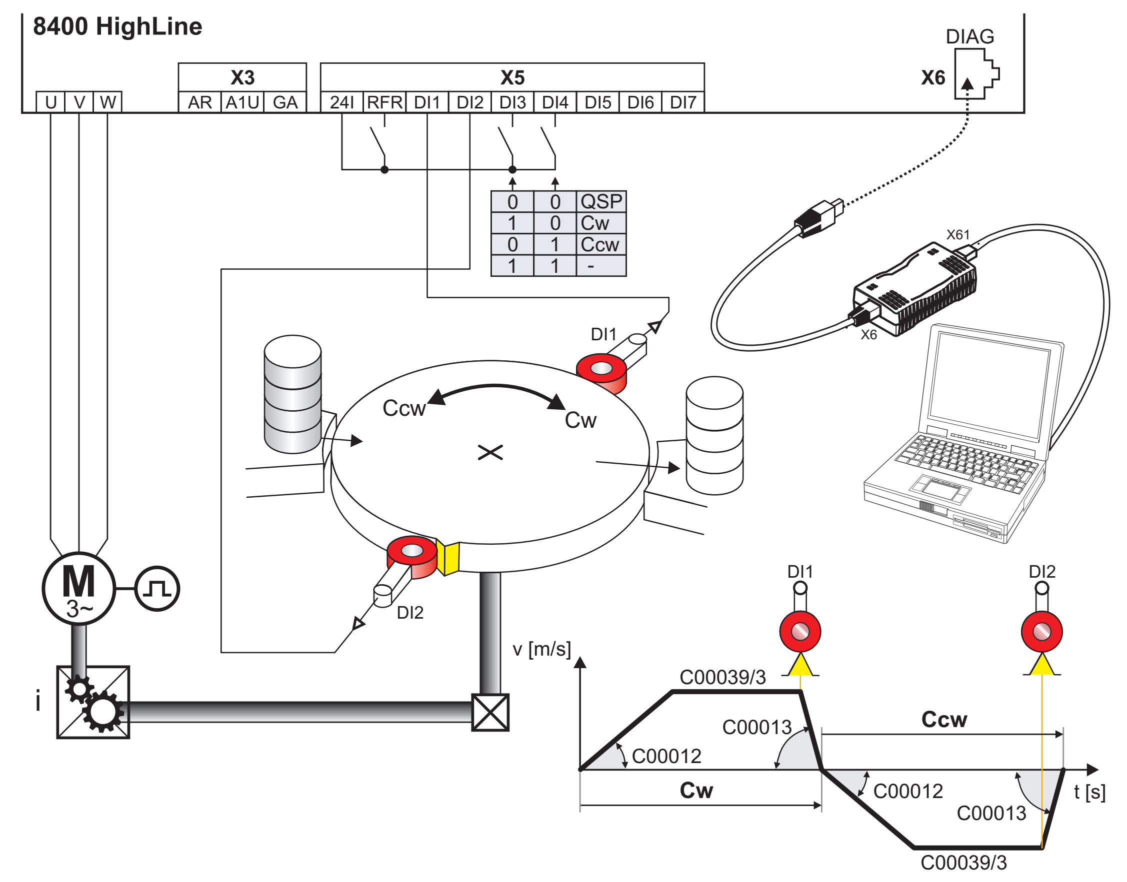
3.9 Commissioning of the "Switch-off positioning" technology application

Note!

Take all the necessary safety precautions before you carry out the following commissioning steps and switch the device on!

▶ Safety instructions with regard to commissioning (📖 59)

System constellation



X4/RFR	Controller enable
X5/DI1	Initiator connection for stop function 1
X5/DI2	Initiator connection for stop function 2
X5/DI3	Quick stop for CW rotation / selection of switch-off position 1
X5/DI4	Quick stop for CCW rotation / selection of switch-off position 2
X6	Diagnostic interface for connecting the USB diagnostic adapter

[3-3] Block diagram for wiring of the commissioning example for the "Switch-off positioning" application

Functional principle of a switch-off positioning without pre-switch off

In case of the switch-off positioning without pre-switch off shown above, it makes sense to use the "Terminals 2" control mode:

1. Set DI3 to HIGH level to activate CW rotation.
2. The drive accelerates along the acceleration ramp (C00012) up to the traversing speed set in C00039/3.
3. After reaching the DI1 contact, the drive comes to a stop with quick stop (QSP) in the target position.
4. Reset DI3 to LOW level and set DI4 to HIGH level to activate CCW rotation now.
5. The drive is accelerated along the acceleration ramp (C00012) up to the traversing speed set in C00039/3.
6. After the DI2 contact has been reached, the drive is braked to standstill with quick stop (QSP) in the initial position.

Tip!
- In order to avoid positioning inaccuracy due to signal propagation delays, the initiators can be directly evaluated by the inverter. Limit switch evaluation can be configured in the inverter. In code C00488/x you can change the method of detecting position signals from level evaluation to edge evaluation.
- In order to prevent unintended movements of the load in the target position, the use of a holding brake is recommended as an alternative to DC-injection braking (limited torque).
- The device terminals and their function assignment do not appear in the FB Editor. The assignment of (hardware) terminals to (software) functions is explained in the chapter "Terminal assignment of the control modes". (📖 555)

Commissioning steps

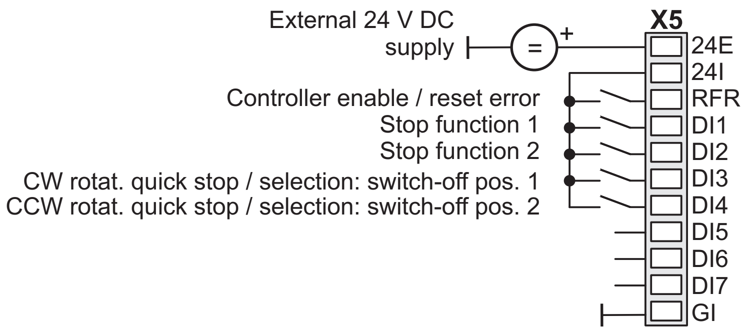
As shown in illustration [3-3], below find a description of the commissioning steps of the "Switch-off positioning" application without pre-switch off.

Please observe the sequence of the steps in the following chapters and follow them through carefully. This will help you to commission your inverter quickly and as safely as possible:

- Prepare inverter for commissioning (📖 90)
- Creating an »Engineer« project & going online (📖 91)
- Parameterising the motor control (📖 92)
- Parameterise application (📖 93)
- Save parameter settings safe against mains failure (📖 95)
- Enable inverter and test application (📖 95)

3.9.1 Prepare inverter for commissioning

1. Power terminal wiring.
   Refer to the mounting instructions supplied with the inverter to find help on how to correctly design the power connections to match the requirements of your device.
2. Wire the control terminals.
   In case of the application shown in illustration [3-3], switch-off positioning without pre-switch off, wiring according to the "Terminals 2" control mode makes sense:



3. Inhibit inverter: Set terminal X5/RFR to LOW level or open contact.
4. Connect USB diagnostic adapter.
5. Switch on voltage supply of the inverter.
   - Without motor operation: Connect external 24 V supply.
   - With motor operation: Connect mains voltage.

If the green "DRV-RDY" LED is blinking and the red "DRV-ERR" LED is off, the inverter is ready for operation and commissioning can proceed.

Related topics:

- Automatic restart after mains connection/fault... (📖 130)
- LED status displays (📖 720)

3.9.2 Creating an »Engineer« project & going online

You can find detailed information on the general use of the »Engineer« in the online help which you can call with **[F1]**.

- The chapter "Working with projects" describes, among other things, all options of the *Start-up wizard* which are available to create a new »Engineer« project.

The following steps serve to describe a general method for creating a project with the **Select component from catalogue** option. For this purpose, individual components (inverter, motor, etc.) are selected from selection lists.

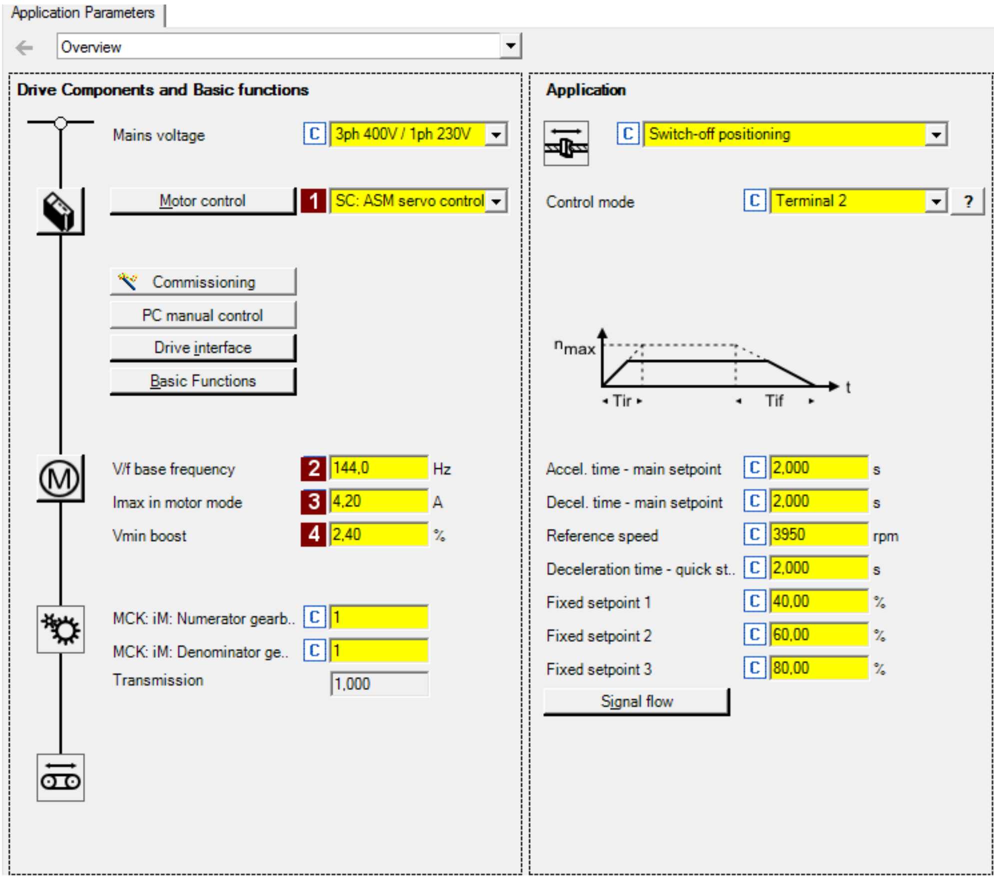
1. Start »Engineer«.
2. Create a new project with the *Start-up wizard* and the **Select component from catalogue** option:
   - In the **Component** step, select the 8400 TopLine inverter.
   - In the **Device modules** step, select the available communication module.
   - In the **Application** step, select the "Switch-off positioning" application. (The application can also be selected any time afterwards via the **Application parameter** tab or C00005.)
   - Select the other components (motor/gearbox) to be added to the project in the **Other components** dialog step.
3. Go online.
   After a connection to the inverter has been established, the following status is displayed in the *Status line*:

4. Transfer parameter set to the device.
   This command serves to overwrite the current parameter settings in the inverter by parameter settings of the »Engineer« project.

3.9.3 Parameterising the motor control

1. Select the **Application parameters** tab from the *Workspace*.
   The motor control parameters, among other things, can be found on the left:



2. In the **1** **Motor control** list field (C00006), select the desired motor control.
3. Adapt the motor control parameters:

Parameters		Lenze setting		Info
		Value	Unit	
2	**V/f base frequency** (C00015)	50.0	Hz	▶ Adapting the V/f base frequency (📖 174)
3	**Imax in motor mode** (C00022)	47.00	A	▶ Optimising the Imax controller (📖 178)
4	**Vmin boost** (C00016)	1.60	%	▶ Adapting the Vmin boost (📖 176)

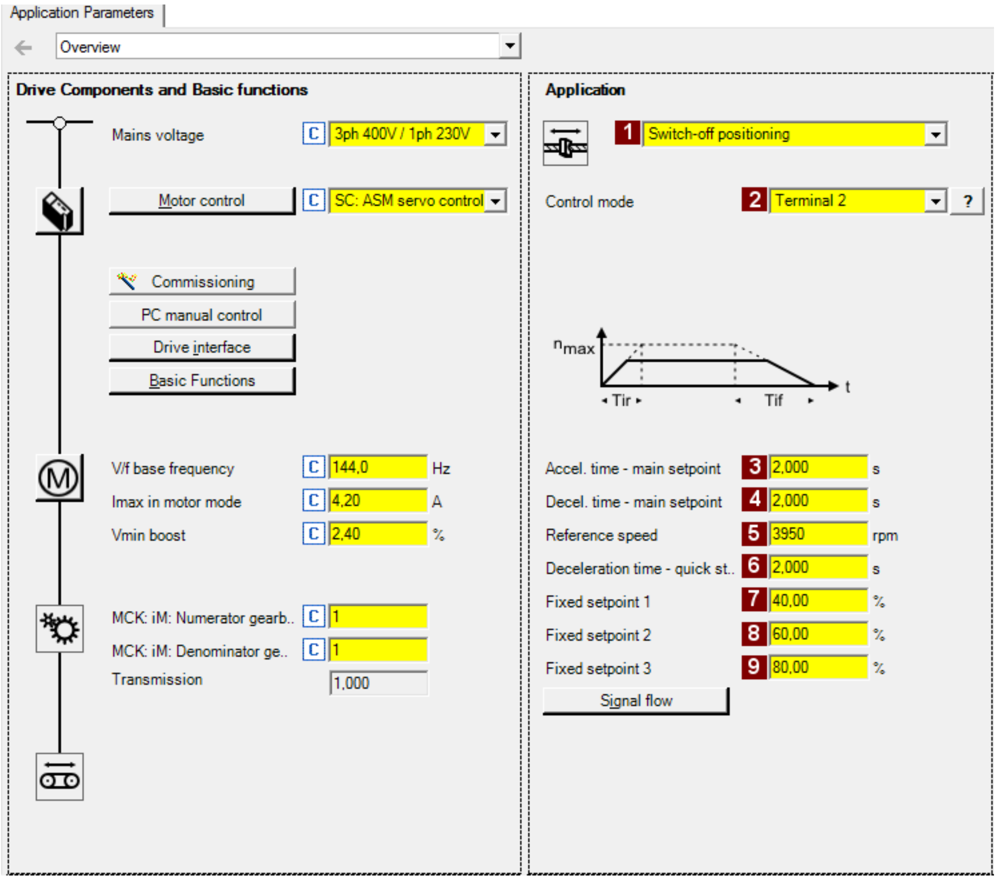
Related topics:

▶ Notes on motor control (📖 60)

▶ Motor control (MCTRL) (📖 143)

3.9.4 Parameterise application

The application parameters can be found on the right side of the **Application parameter** tab:



1. In the **1** **Application** list field (C00005), select the "Switch-off positioning" application (if you have not already done so while creating the project).
   After the "Switch-off positioning" application is selected, the contents of the tab change, e.g. the **Process controller** and **Motor potentiometer** buttons are not shown any more.
2. In the **2** **Control mode** list field (C00007) and in case of illustration [3-3], for the shown switch-off positioning without pre-switch off the "Terminals 2" control mode must be selected.
   - The corresponding wiring diagram is displayed in a pop-up window if you click the ? button right to the list field.
   - For a detailed description, see the chapter "Terminal assignment of the control modes". (465)

3. Adapt the application parameters:

	Parameters	Lenze setting Value	Unit	Info
3	**Accel. time - main setpoint** (C00012)	2.000	s	The setpoint is led via a ramp function generator with linear characteristic. The ramp function generator converts setpoint step-changes at the input into a ramp. Note: These settings only apply if no other ramp times have been selected at the L_NSet FB!
4	**Decel. time - main setpoint** (C00013)	2.000	s	
5	**Reference speed** (C00011)	1500	rpm	All speed setpoint selections are provided in % and always refer to the reference speed set in C00011. The motor reference speed is indicated on the motor nameplate.
6	**Decel. time - quick stop** (C00105)	2.000	s	If quick stop is requested, motor control is decoupled from the setpoint selection and, within the deceleration time parameterised in C00105, the motor is brought to a standstill (n_{act} = 0). ▸ Activate/deactivate quick stop (📖 114)
7	**Preset setpoint 1** (C00039/1)	40.00	%	Fixed setpoints are selected in [%] based on the reference speed (C00011). **Fixed setpoint 2 must be smaller than fixed setpoint 3!** Otherwise, the drive will be started with a low speed and accelerated after the pre-switch off.
8	**Preset setpoint 2** (C00039/2)	60.00	%	
9	**Preset setpoint 3** (C00039/3)	80.00	%	

Tip!

- Click the **Signal flow** button to go down one dialog level to the signal flow of the application with further possible parameter settings. See chapter "Basic signal flow". (📖 546)
- The preconfigured I/O connection in the selected control mode can be changed via configuration parameters. See chapter "User-defined terminal assignment". (📖 445)
- Low-jerk traversing profiles can be implemented by means of S-shaped ramps.
- In the case of high breakaway torques combined with horizontal motion sequences, "Sensorless vector control (SLVC)" can be used as motor control (C00006).
- For reversal of rotation direction (bidirectional motion), comprehensive configuration options are available in the inverter (e.g. by means of the L_DFlipFlop function block).

More detailed information on the technology application:

▸ TA "Switch-off positioning" (📖 544)
▸ Internal interfaces | application block "LA_SwitchPos" (📖 547)
▸ Process data assignment for fieldbus communication (📖 564)
▸ Terminal assignment of the control modes (📖 555)
▸ Setting parameters (short overview) (📖 566)
▸ Configuration parameters (📖 568)

3.9.5 Save parameter settings safe against mains failure

In order to prevent parameter settings carried out in the device from being lost by mains switching, you have to explicitly save the parameter set with mains failure protection in the device.

Save parameter set.

3.9.6 Enable inverter and test application

Stop!

Before stipulating a speed setpoint, check whether the brake in the form of a holding brake on the motor shaft has been released!

Note!

If the controller is enabled at mains connection and C00142 has activated the "Inhibit at device on" auto-start option (Lenze setting), the inverter remains in the "ReadyToSwitchOn" state.

To be able to change to the "SwitchedOn" status, the controller enable must be deactivated first: set terminal X5/RFR to LOW level.

If the inverter is in the "SwitchedOn" state:

1. Enable inverter: Set terminal X5/RFR to HIGH level or close contact.
    - If there is no other active source for the controller inhibit, the inverter changes from the "SwitchedOn" status to the "OperationEnabled" status.
    - The **Diagnostics** tab and C00158 display all active sources for the controller inhibit.
2. Select the respective control signals via the digital inputs.

Note!

Observe the actual speed value (display in C00051) as well as the LED status displays. (📖 720)

Related topics:

▸ "Inhibit at device on" auto-start option (📖 130)
▸ Trouble-shooting during commissioning (📖 62)
▸ Diagnostics & error management (📖 719)

3.10 PC manual control

For the purpose of testing and demonstration and when an online connection has been established, the PC manual control enables the manual control of various drive functions from the »Engineer«.

Supported drive functions:

- Speed control (follow speed setpoint)
- Activate/deactivate quick stop
- Set/reset home position
- Manual jog
- Positioning (relative or absolute)

More control functions:

- Reset error message
- Set digital/analog outputs (in preparation)

Diagnostic functions:

- Display of the actual speed value and motor current (as time characteristic)
- Display of the current device status
- Display of the status determining error
- Display of the status of the digital/analog inputs (in preparation)

3.10.1 Activate PC manual control

STOP **Stop!**

PC manual control must be explicitly activated by the user.

If PC manual control is activated, the inverter is inhibited via device command (C00002/16) first.

Note!

With active PC manual control:

The online connection between PC and controller is monitored by the inverter.

- If the online connection is interrupted longer than the timeout time set (Lenze setting: 2 s):
  - The error response "Fault" takes place, i.e. the motor becomes torqueless and coasts, if it has not yet been in standstill.
  - The "Ck16: Time overflow manual control" error message is entered into the logbook.

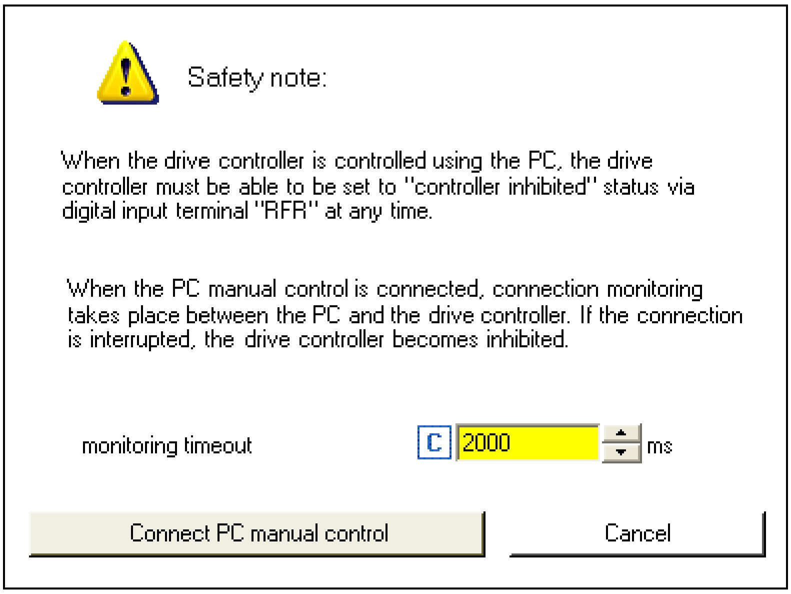
PC manual control provides the **Motion Control Kernel** and the motor interface with all required control signals and setpoint signals.

- The available application (function block interconnection) is now decoupled from these interfaces, but is continued to be processed and remains unchanged.
- It does not matter what type of motor control is set in C00006.

How to activate the PC manual control:

1. If an online connection to the inverter has not been established yet:

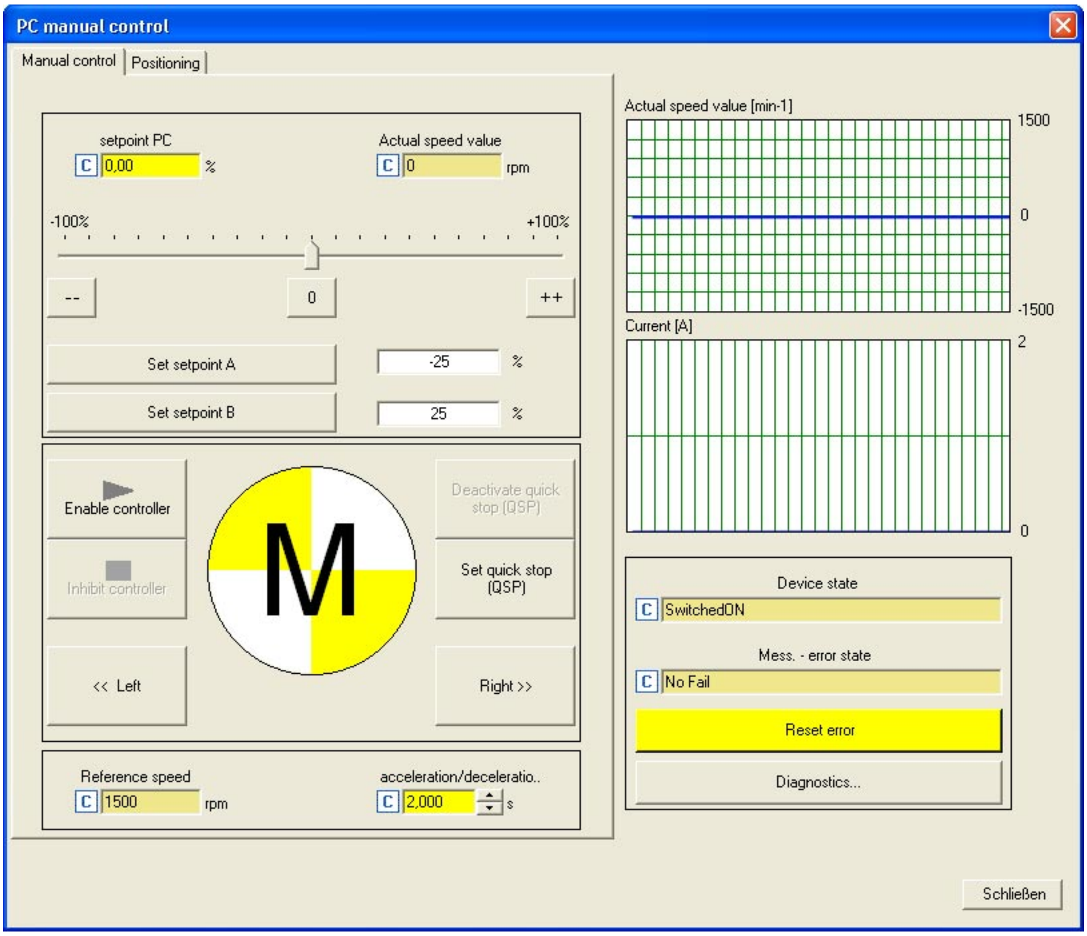
   Go online.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the "**PC manual control**" button.
   - First, the following safety note is displayed:



   - Click the **Cancel** button to abort the action and close the dialog box.
   - The **Timeout monitoring** input field serves to adapt the timeout time for the connection monitoring between PC and inverter.
4. To acknowledge the note and activate PC manual control:
   Click the **Activate PC manual control** button.
   - The inverter is inhibited via device command (C00002/16).
   - The *PC manual control* operator dialog is displayed.

PC manual control - operator dialog

On the left-hand side, the *PC manual control* operator dialog includes several tabs which serve to select various control functions. On the right-hand side, setpoint and status displays are provided for diagnostic purposes:



Note!

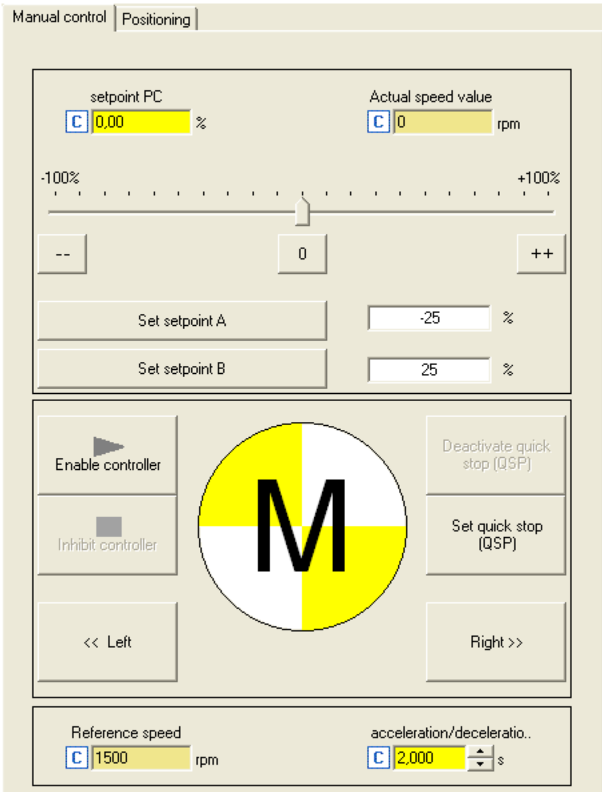
PC manual control can be exited any time by clicking the **Close** button.

If you exit PC manual control or change to another tab, the inverter is inhibited via device command (C00002/16), i.e. the motor becomes torqueless and is coasting unless it already is at standstill.

The different versions are described in the following chapters.

3.10.2 Speed control

The **Speed control** serves to easily rotate the drive without setting control parameters or feedback systems in the "Speed follower" mode:

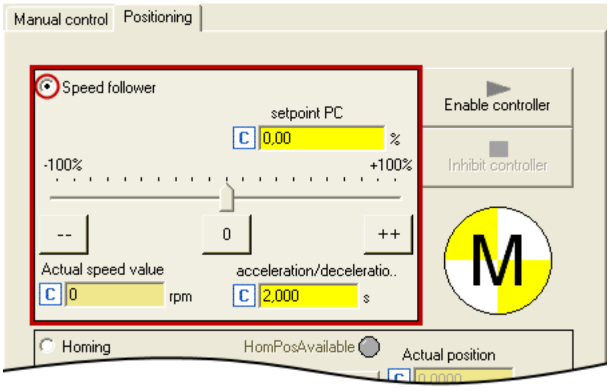


How to easily rotate the motor:

1. Set the desired speed setpoint in [%] based on the reference speed, e.g. directly in the **Setpoint PC** input field or via the slider.
   - Via the **--** / **0** / **++** buttons, the currently set speed setpoint can be reduced/increased in steps of 10 percent or set to zero.
   - Via the **Set setpoint A/B** buttons, the speed setpoint can be set to a previously set constant value A/B.
2. To start the speed follower:
   Enable the inverter via the **Enable controller** button.
   - Please observe that the inverter will not be enabled if other sources of controller inhibit (e.g. RFR terminal) are active.
   - The enabled drive now follows the defined speed setpoint.
   - In order to prevent shocks or overload at higher setpoint changes, the speed setpoint is lead via a linear ramp generator with adjustable acceleration/deceleration time.
   - Via the **Inhibit controller** button, the inverter can be inhibited again, i.e. the motor becomes torqueless and is coasting unless it already is at standstill.

Further functions:

- If the **Set quick stop (QSP)** button is clicked, the motor is braked to a standstill within the deceleration time parameterised in C00105.
  - Via the **Deactivate quick stop (QSP)** button, the quick stop can be deactivated.
- Via the **<< CCW** and **CW >>** buttons, the direction of rotation can be changed.
- The "speed follower" operating mode can be activated via the second **Positioning** tab without the functions described before:

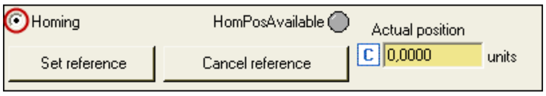


3.10.3 Set/reset home position

The measuring system in the machine is selected by means of homing and the 0 position is set within the possible physical travel range.

Detailed information on the "Homing" operating mode is provided in chapter "Basic drive functions" in subchapter "Homing". (🕮 637)

A change to the "Homing" operating mode is made by selecting the correspondent option on the **Positioning** tab:



How to set the reference manually with standing drive:

Click the **Set reference** button.

The current actual position in the machine measuring system now corresponds to the home position set in C01227/2 (Lenze setting: 0.0000 units).

Stop!

If you set the reference with PC manual control, the drive is referenced with it.

If another home position is used for normal operation:

Click **Cancel reference** and reset the status signals *HomPosAvailable* and *HomePosDone* in the MCK status word in order that no mechanical problems occur in normal operation.

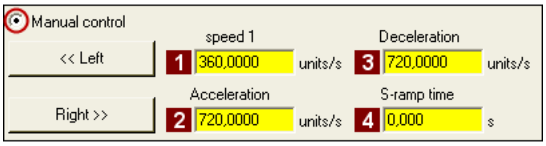
- Setpoints and actual positions remain untouched until a renewed reference setting or homing.

3.10.4 Manual jog

In the "Manual jog" mode, the drive can be traversed manually in a clockwise or anticlockwise direction.

You can find detailed information on the "Manual jog" mode in the chapter entitled "Basic drive functions" in the subchapter "Manual jog". (📖 658)

A change to the "manual jog" operating mode is made by selecting the correspondent option on the **Positioning** tab:



- The profile parameters stored in the parameter set for the basic "manual jog" function are used:

Parameters		Lenze setting Value	Unit	Info
1	**Speed 1** (C01231/1)	360.0000	units/s	Manual jog speed
2	**Acceleration** (C01232/1)	720.0000	units/s^2	For accelerating and decelerating, different values can be set so that smooth starting and quick stopping of the drive can be implemented.
3	**Deceleration** (C01232/2)	720.0000	units/s^2	
4	**S-ramp time** (C01233/1)	0.000	s	In order to reduce jerking, the two ramps can be set in such a way that they are s-shaped. This is done by entering a relative S-ramp time.

How to traverse the drive manually in inching mode:

1. If the inverter is still inhibited, enable the inverter via the **Enable controller** button.
   - Please observe that the inverter will not be enabled if other sources of controller inhibit (e.g. RFR terminal) are active.

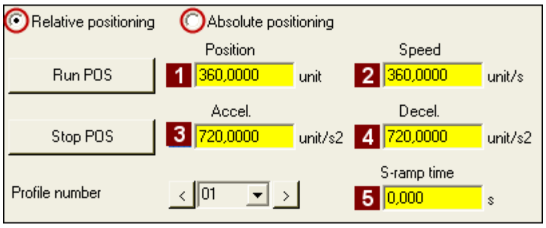
   If the inverter is in the "OperationEnabled" device status:
2. Press the **<< Left** or **Right >>** button (and keep it pressed) to traverse the drive with the corresponding direction of rotation.

3.10.5 Positioning (relative or absolute)

Positioning means that a workpiece/tool or material is moved from a starting position to a defined destination.

You can find detailed information on the "Positioning" mode in the chapter entitled "Basic drive functions" in the subchapter "Positioning". (📖 666)

A change to the "Positioning" mode and a simultaneous selection of the positioning mode (relative or absolute) is made by selecting the corresponding option on the **Positioning** tab:



- The profile parameters stored in the parameter set for the basic "Positioning" function are used:

Parameters		Lenze setting		Info
		Value	Unit	
1	**Position** (C01301/1...15)	360.0000	units	Target position or distance to be traversed. • With "Relative positioning": Distance to starting position (current position). • With "Absolute positioning": Distance to defined zero position.
2	**Velocity** (C01302/1...15)	360.0000	units/s	Maximum velocity at which the target is to be approached.
3	**Acceleration** (C01303/1...15)	720.0000	units/s^2	Specification of the maximum velocity variation which is to be used for acceleration.
4	**Deceleration** (C01304/1...15)	720.0000	units/s^2	Specification of the velocity variation which is to be used for maximum deceleration to standstill.
5	**S-ramp time** (C01306/1...15)	0.000	s	A profile is traversed with S-shaped ramps by defining an S-ramp time for it.

Note!

In order that no complex travel movements occur, no switching to a sequence profile takes place that has been set in the profile data!

You can find detailed information on entering profiles in the chapter entitled "Basic drive functions", subchapter "Profile entry". (📖 671)

How to carry out positioning:

1. Select the profile to be executed (1 ... 15) in the **Profile number** list field.
2. If the inverter is still inhibited, enable the inverter via the **Enable controller** button.
   - Please observe that the inverter will not be enabled if other sources of controller inhibit (e.g. RFR terminal) are active.

   If the inverter is in the "OperationEnabled" device status:
3. Press the **Run POS** button to start positioning.
   - The **Stop POS** serves to stop active positioning at any time.

4 Device control (DCTRL)

This chapter provides information on internal device control as well as the device commands which can be executed via the subcodes of C00002.

- The device control causes the inverter to take defined device statuses.
- The device control provides a multitude of status information in many ways:
  - Visually via the LED status displays on the front of the inverter. (📖 721)
  - As text messages in the Logbook. (📖 728)
  - As process signals via the outputs of the LS_DriveInterface system block. (📖 135)
  - Via diagnostic / display parameters which are included in the »Engineer« parameter list as well as in the **Diagnostics** category in the keypad.

Note!

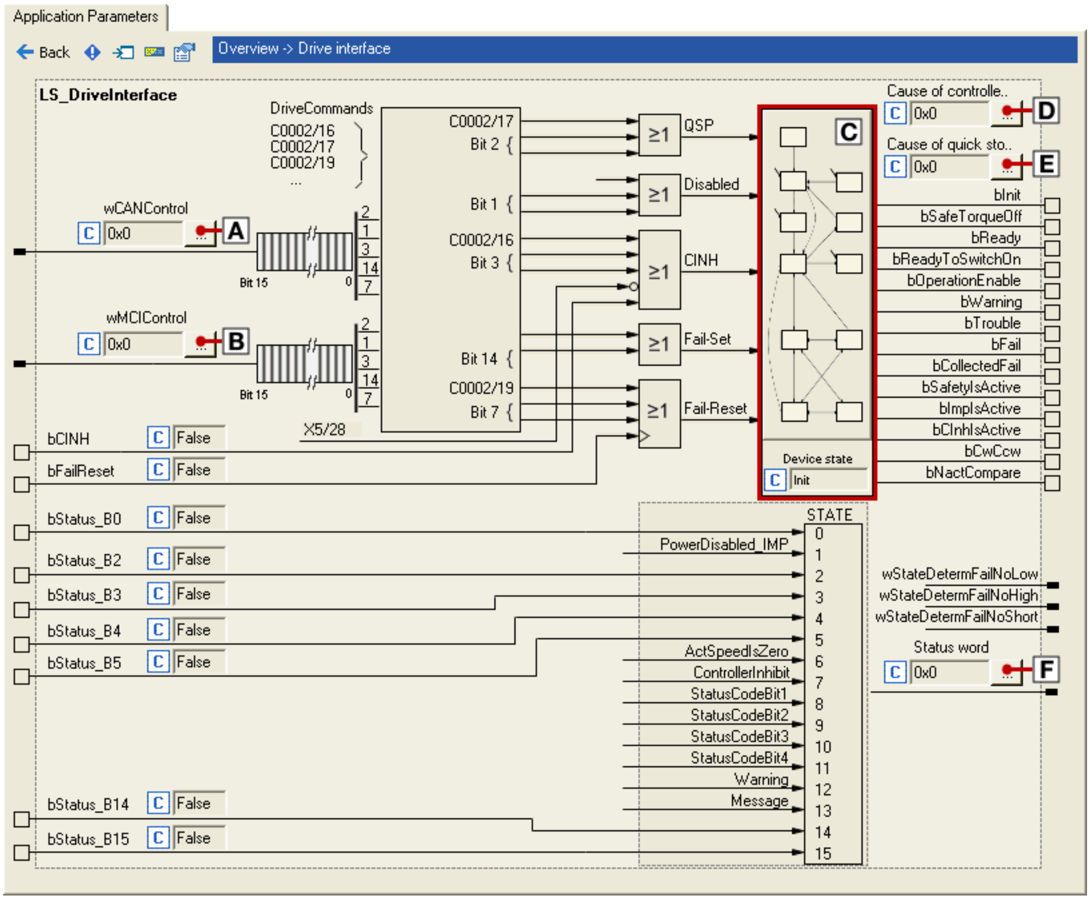
The device states of the inverter are based on the operating states of the CiA402 standard. ▶ Device state machine and device states (📖 119)

How to get to the parameterisation dialog of the device control:

1. Go to the *Project view* of the »Engineer« and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the **Drive interface** button.

Parameterisation dialog in the »Engineer«

The parameterisation dialog shows the input / output signals and the internal signal flow of the LS_DriveInterface system block which displays the device control in the function block editor:



Range / Meaning		Display parameter
A	Display of the control word via system bus (CAN)	C00136/2
B	Display of the control word via communication module (e.g. PROFIBUS)	C00136/1
C	Display of the internal state machine and the current device status	C00137
D	Display of all active sources of a controller inhibit	C00158
E	Display of all active sources of a quick stop	C00159
F	Display of the status word of the device control	C00150

4.1 Device commands (C00002/x)

This chapter describes the device commands of the inverter which are provided in the subcodes of C00002 and can be carried out using the keypad or, alternatively, the »Engineer« when an online connection has been established.

The device commands serve, among other things, to directly control the inverter to organise parameter sets, and to call diagnostic services.

Regarding the execution of the device commands, a distinction is drawn between:

- Device commands which have an immediate effect on control (e.g. "Activate quick stop")
  - After being called in C00002/x, these device commands provide static status information ("On" or "Off").
- Device commands with longer execution duration (several seconds)
  - After being called in C00002/x, these device commands provide dynamic status information ("Work in progress 20%" → "Work in progress 40%", etc.).
  - The execution of the device command has not finished successfully until the "Off / ready" status information is provided in C00002/x.
  - In the event of an error, the "Action cancelled" status information is provided in C00002/x . In this case, further details can be obtained from the status of the device command executed last which is displayed in C00003.

STOP

Stop!

Before the supply voltage is switched off after a device command has been transmitted via C00002/x, the device command must be checked for successful completion on the basis of the status information provided in C00002/x!

- This is of particular importance for device commands which save data to the memory module of the device. Incomplete storage processes may lead to data inconsistencies in the memory module.

Note!

- Before activating device commands by a master control, wait for the "Ready" signal of the inverter.
- The device will reject a write process to C00002/x if the value is >1 and issue an error message.
- C00003 displays the status of the device command that was executed last.

Activate device command

When an online connection has been established, simply use the »Engineer« to activate a device command by selecting the corresponding option from the **Parameters** tab in C00002/x ("0: off" or "1: On / start").

- Alternatively, the device command can also be activated via e.g. keypad or through a master control by writing to C00002/x.
- Some of the frequently used device commands (such as "Save parameter set") can also be executed via the *Toolbar* icons of the »Engineer« when an online connection has been established:

Symbol	Function
	Enable inverter
	Inhibit inverter
	Save parameter set (for 8400: Save all parameter sets)

Note!

Device commands that can be executed via the *Toolbar* of the »Engineer« always affect the element currently selected in the *Project view* including all subelements!

- If no inverter but a system module is selected in the *Project view*, the corresponding device command will be activated in all lower-level inverters having an online connection with the »Engineer«.

Before the desired action is carried out, a confirmation prompt appears first, asking whether the action is really to be carried out.

Short overview of device commands

Device commands described in this chapter:

C00002 Subcode:	Device command	Controller inhibit required	Status information
1	Load Lenze setting	●	dynamic
6	Load all parameter sets	●	dynamic
11	Save all parameter sets		dynamic
16	Enable/inhibit inverter		static
17	Activate/deactivate quick stop		static
19	Reset error		static
21	Delete logbook		static
27	Device search function		static

Device commands described in other chapters:

C00002 Subcode:	Device command	Controller inhibit required	Status information
23	Identify motor parameters ▶ Automatic motor data identification	●	dynamic
25	Identify resolver error ▶ Optimising resolver behaviour	●	dynamic
26	CAN reset node ▶ Reinitialising the CANopen interface		static
28	Check MasterPin ▶ Unlocking the inverter with a MasterPin	●	static
29	Set binding ID ▶ Device personalisation		static
30	Delete binding ID ▶ Device personalisation		static
31	Set password ▶ Password protection		static
32	Check password ▶ Password protection		static
33	Delete password ▶ Password protection		static
34	Identify pole position (360°) ▶ Pole position identification (PPI)	●	dynamic

4.1.1 Load Lenze setting

The C00002/1 = "1: On / start" device command resets the parameters to the Lenze setting which are saved in the inverter firmware.

- Can only be executed if the controller is inhibited; otherwise, the feedback C00002/1 = "6: No access - controller inhibit" will be returned.
- All parameter changes made since the last saving of the parameter set will get lost!
- This device command has an effect on the settings of the parameters of the operating system, application and module.

Note!

When the Lenze setting C00002/1 is loaded, all communication parameters are reset as well. After the mains is switched on, the Lenze setting is accepted and the inverter might not be accessible anymore via the communication module.

From version 18.00.00 onwards, C01004 (Load Lenze setting without:) serves to prevent all communication parameters from being reset when the Lenze setting C00002/1 is loaded.

In order that the communication parameters are not reset while loading the Lenze setting, you must parameterise C01004 before mains switching:
- When using a communication module: C01004:set bit 0 = 1.
- When using CAN OnBoard: C01004:set bit 1 = 1.

How to load the Lenze setting:

1. If the inverter is enabled, it must be inhibited, e.g. by executing the "Enable/Inhibit inverter" device command "(C00002/16 = "0: Off / ready").
2. Execute the "Load Lenze setting" device command:
   C00002/1 = "1: On / start"

   The loading process may take a couple of seconds. After the device command has been called, C00002/1 returns dynamic status information ("Work in progress 20 %" → "Work in progress 40 %" → "Work in progress 60 %", etc.).

Notes on execution time

The execution time of this device command usually is 2.5 s, independent of the communication module used. If - in addition to the device-internal time monitoring - a monitoring of the execution time is to be implemented for this device command in a higher-level control, the monitoring time has to be 2 ... 3 times higher.

4.1.2 Load all parameter sets

The C00002/6 = "1: On / start" device command reloads all parameter settings from the memory module to the inverter.

- Can only be executed if the controller is inhibited; otherwise, the feedback C00002/6 = "6: No access - controller inhibit" will be returned.
- All parameter changes made since the last saving of the parameter set will get lost!
- This device command has an effect on the settings of the parameters of the operating system, application and module.

Note!

The inverter is currently provided with one data record for all parameters, i.e. every parameter has a value. Several data records per inverter are in preparation.

The basic function Parameter change-over provides a change-over between four sets with different parameter values for up to 32 freely selectable parameters. (914)

How to load the parameter settings from the memory module:

1. If the inverter is enabled, it must be inhibited, e.g. by executing the "Enable/Inhibit inverter" device command "(C00002/16 = "0: Off / ready").
2. Execute the "Load all parameter sets" device command: C00002/6 = "1: On / start"

   The loading process may take a couple of seconds. After the device command has been called, C00002/6 returns dynamic status information ("Work in progress 20 %" ➞ "Work in progress 40 %" ➞ "Work in progress 60 %", etc.).

Notes on execution time

The execution time of this device command depends on the communication module used. If - in addition to the device-internal time monitoring - a monitoring of the execution time is to be implemented for this device command in a higher-level control, the monitoring time has to be 2 ... 3 times higher than the times specified in the following table:

Communication module used	Typical execution time		
	StateLine C	HighLine C	TopLine C
Without communication module	3.5 s	3.5 s	3.5 s
PROFIBUS® or INTERBUS	3.5 s	3.5 s	3.5 s
EtherCAT®, PROFINET® or EtherNet/IP™	4.0 s	3.5 s	4.0 s
Ethernet POWERLINK	15.0 s	15.0 s	15.0 s

4.1.3 Save all parameter sets

If parameter settings are changed in the inverter, those changes will be lost after mains switching of the inverter unless the settings have been saved explicitly.

The C00002/11 = "1: On / start" device command saves the current parameter settings safe against mains failure to the memory module of the inverter.

Note!

When the device is switched on, all parameters are automatically loaded from the memory module to the main memory of the inverter.

Observe the following to avoid data inconsistencies which cause errors when the parameters are loaded from the memory module:

During the storage process:

- Do not switch off the supply voltage!
- Do not remove the memory module from the device!

The inverter is currently provided with one data record for all parameters, i.e. every parameter has a value. Several data records per inverter are in preparation.

How to save the parameter settings to the memory module:

Execute the "Save all parameter sets" device command:
C00002/11 = "1: On / start"

The storage process may take a couple of seconds. After the device command has been called in C00002/11, dynamic status information ("Work in progress 20%" ➜ "Work in progress 40%" ➜ "Work in progress 60%", etc.) is returned.

Tip!

- This device command can also be activated via the icon in the *Toolbar*.
- The "Load Lenze setting" device command (C00002/1 = "1: On / start") resets the parameter settings to the delivery status of the device.

Notes on execution time

The execution time of this device command depends on the communication module used. If - in addition to the device-internal time monitoring - a monitoring of the execution time is to be implemented for this device command in a higher-level control, the monitoring time has to be 2 ... 3 times higher than the times specified in the following table:

Communication module used	Typical execution time		
	StateLine C	HighLine C	TopLine C
Without communication module	2.0 s	2.5 s	2.5 s
PROFIBUS® or INTERBUS	3.5 s	3.5 s	4.0 s
EtherCAT®, PROFINET® or EtherNet/IP™	4.5 s	4.5 s	4.5 s
Ethernet POWERLINK	11.0 s	11.0 s	11.0 s

4.1.4 Enable/inhibit inverter

The C00002/16 = "1: On / start" device command enables the inverter, provided that no other source of an inverter inhibit is active.

The C00002/16 = "0: Off / ready" device command inhibits the inverter again, i.e. the power output stages in the inverter are inhibited and the speed/current controllers of the motor control are reset.

- The motor becomes torqueless and coasts, if it has not yet been in standstill.
- When the controller is inhibited, the status output *bCInhActive* of the LS_DriveInterface system block is set to TRUE.
- When the controller inhibit request is reset, the drive synchronises to the actual speed. For this purpose,
  - If the flying restart circuit is activated in C00990, the flying restart function parameterised in C00991 is used for the synchronisation to the rotary or standing drive. ▸ Flying restart function (📖 280)
  - In the case of an operation with feedback, the actual speed is read out by the encoder system.
  - In the case of a sensorless vector control (SLVC), the actual speed from the motor model of the motor control is used for the synchronisation.
- C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit:

Bit	Cause/Source of controller inhibit
Bit 0	Terminal controller enable
Bit 1	CAN control word
Bit 2	MCI control word
Bit 3	SwitchOn
Bit 4	Application (LS_DriveInterface system block: *bCInh* input)
Bit 5	**Device command** (C00002/16)
Bit 6	Error with error response "Fault" or "Trouble"
Bit 7	Internal signal
Bit 8	Reserved
Bit 9	Reserved
Bit 10	AutoStartLock
Bit 11	Motor parameter identification
Bit 12	Automatic brake operation
Bit 13	DCB-IMP
Bit 14	Reserved
Bit 15	Reserved

Tip!

The inverter can also be enabled or inhibited via the and *toolbar* icons.

4.1.5 Activate/deactivate quick stop

The C00002/17 = "1: On / start" device command activates the quick stop function, i.e. the motor control is separated from the setpoint selection, and within the deceleration time parameterised inC00105 the motor is brought to a standstill (n_{act} = 0).

Parameters	Info	Lenze setting	
		Value	Unit
C00105	Decel. time - quick stop	2.000	s

- The motor is kept at a standstill during closed-loop operation.
- A pulse inhibit is set if the auto-DCB function has been activated via C00019.
- From version 02.00.00, further options for the quick stop function can be activated in C00104/1.
  ▶ Optional settings (📖 115)
- C00159 provides a bit coded representation of all active sources/triggers of a quick stop:

Bit	Cause/source for quick stop
Bit 0	Reserved
Bit 1	CAN control word (bit 2)
Bit 2	MCI control word (bit 2)
Bit 3	Reserved
Bit 4	Application (LS_MotorInterface system block: *bQspOn* input)
Bit 5	**Device command** (C00002/17)
Bit 6	Device error with "TroubleQSP" error response
Bit 7	Internal signal
Bit 8	Reserved
Bit 9	Reserved
Bit 10	Operating system
Bit 11	Reserved
Bit 12	MCK (System block LS_MotionControlKernel: Input *bQspOn*)
Bit 13	Reserved
Bit 14	Reserved
Bit 15	Reserved

Deactivate quick stop again

The C00002/17 = "0: Off / ready" device command deactivates the quick stop again, provided that no other source of a quick stop is active.

4.1.5.1 Optional settings

From version 02.00.00, the options described in the following for the quick stop function can be activated in C00104/1. In the Lenze setting, no option is activated.

Standstill position-controlled

When bit 0 is set in C00104/1, the position control is activated at standstill.

- The basic conditions are shown in the Flow diagram for the quick stop function.
- More conditions for a proper function:
  The inputs *nPosCtrlOutLimit_a* and *nPosCtrlPAdapt_a* at the LS_MotorInterface system block have to be set to 100 % (default setting for all technology applications).

Ramp position-controlled

If bit 1 is set in C00104/1, the position control is activated while ramping down.

- The basic conditions are shown in the Flow diagram for the quick stop function.
- More conditions for a proper function:
  The inputs *nPosCtrlOutLimit_a* and *nPosCtrlPAdapt_a* at the LS_MotorInterface system block have to be set to 100 % (default setting for all technology applications).

Continue following error

(from version 15.00.00)

If bit 2 is set in C00104/1, the current following error is continued if the quick stop with position control is activated while ramping down.

Starting value is the setpoint speed

(from version 15.00.00)

If bit 3 is set in C00104/1, the starting value for quick stop is the setpoint speed. If the bit is not set, the starting value for quick stop is the actual speed if this does not derive more than 5 rpm from the setpoint. This serves to prevent a jump to a setpoint with a high deviation.

Use position encoder

(from version 15.00.00)

If bit 4 is set in C00104/1, the position encoder is used instead of the speed encoder for quick stop with position control.

- The basic conditions are shown in the Flow diagram for the quick stop function.

Standstill depending on the actual speed

(from version 16.00.00)

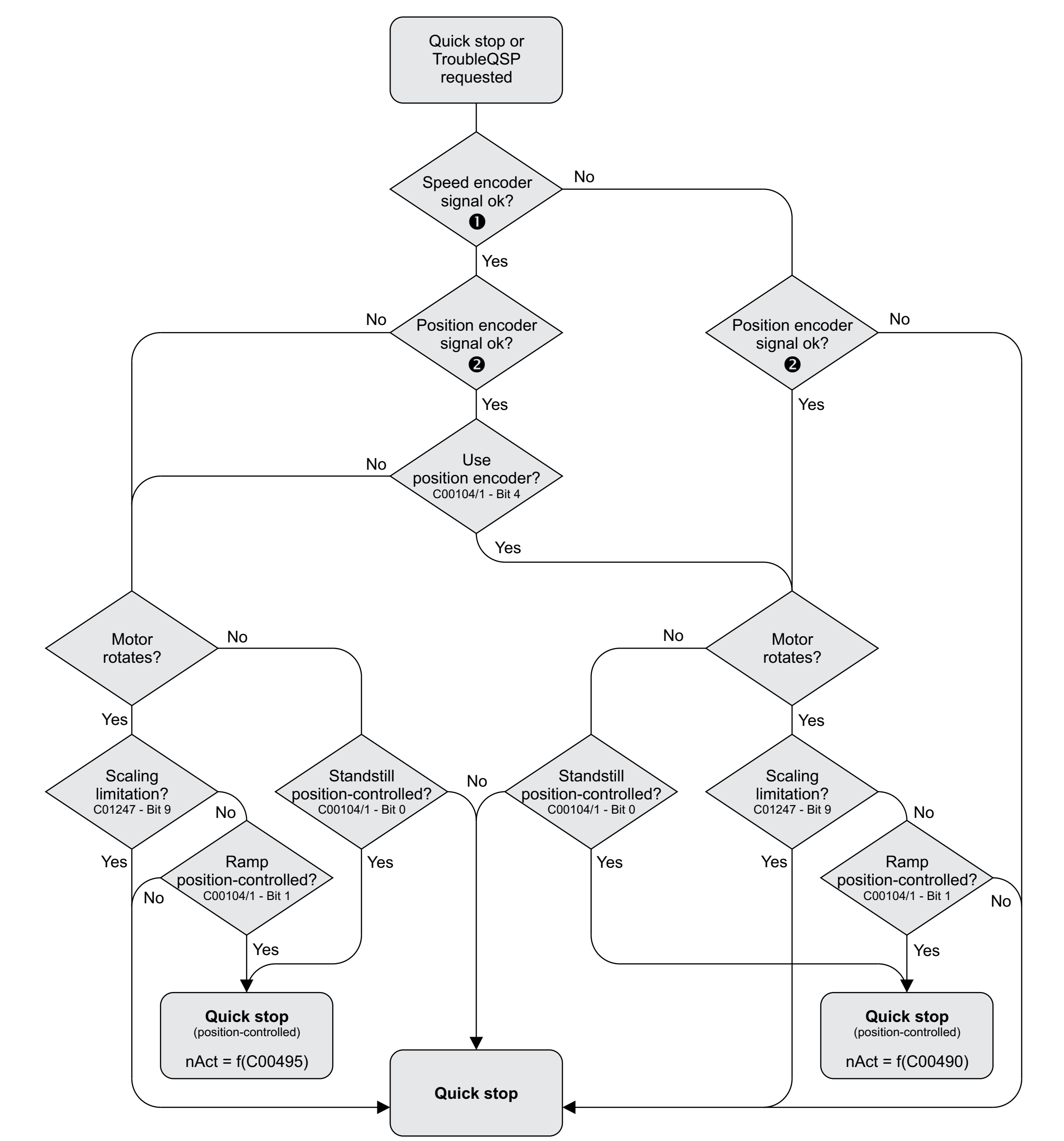
Extended option for quick stop with position control at standstill by setting C00104/1, Bit5 . When this setting is activated, the standstill of the motor shaft is detected under consideration of the speed setpoint and the actual speed.

The standstill of the motor shaft is reached when

- Bit5 = FALSE (Lenze setting)
  - Speed setpoint = 0
- Bit5 = TRUE
  - speed setpoint = 0 AND 0 ≤ actual speed value ≤ 1 rpm

4.1.5.2 Flow diagram

Flow diagram for quick stop function (from version 15.00.00)



❶	The "speed encoder signal ok" status calls for the following requirements: • Speed sensor is set in C00495 • Speed sensor is running (no open circuit)
❷	The "position encoder signal ok" status calls for the following requirements: • Position encoder is set in C00490 • Position encoder is running (no open circuit)

🛈
Although the use of SSI absolute value encoders and 2-track HTL encoders is possible, the error detection system of both encoders always returns "OK" as error status !

4.1.6 Reset error

The C00002/19 = "1: On / start" device command acknowledges an existing error message if the error cause has been eliminated and thus the error is no longer pending.

- After the reset (acknowledgement) of the current error, further errors may be pending which must also be reset.
- The status determining error is displayed in C00168.
- The current error is displayed in C00170.

Tip!

An error message can also be acknowledged by activating the **Reset error** button in the **Diagnostics** tab.

Detailed information on error messages can be found in the "Diagnostics & error management" chapter. (719)

4.1.7 Delete logbook

The C00002/21 = "1: On / start" device command deletes all logbook entries.

Tip!

To display the logbook in the »Engineer«, click the **Logbook** button on the **Diagnostics** tab.

In the *Logbook* dialog box, it is also possible to delete all logbook entries by clicking the **Delete** button.

Detailed information on the logbook can be found in the "Diagnostics & error management" chapter. (📖 719)

4.1.8 Device search function

In some applications where inverters are installed in control cabinets or are positioned in a spacious plant, it is often difficult to locate a device connected online for e.g. maintenance work. There is an established online connection with the device but you do not know where the inverter is located physically.

The C00002/27 = "1: On / start" device command serves to carry out an "optical location":

- For the time set in C00181/1, all four status LEDs at the front of the inverter flash. Afterwards, the function is turned off automatically.
- If the device command is executed again within the set time period, the duration is extended accordingly.
- The setting C00002/27 = "0: Off / ready" serves to abort or switch off the function.
- Adjustable time period: 0 ... 6000 s (Lenze setting: 5 s)

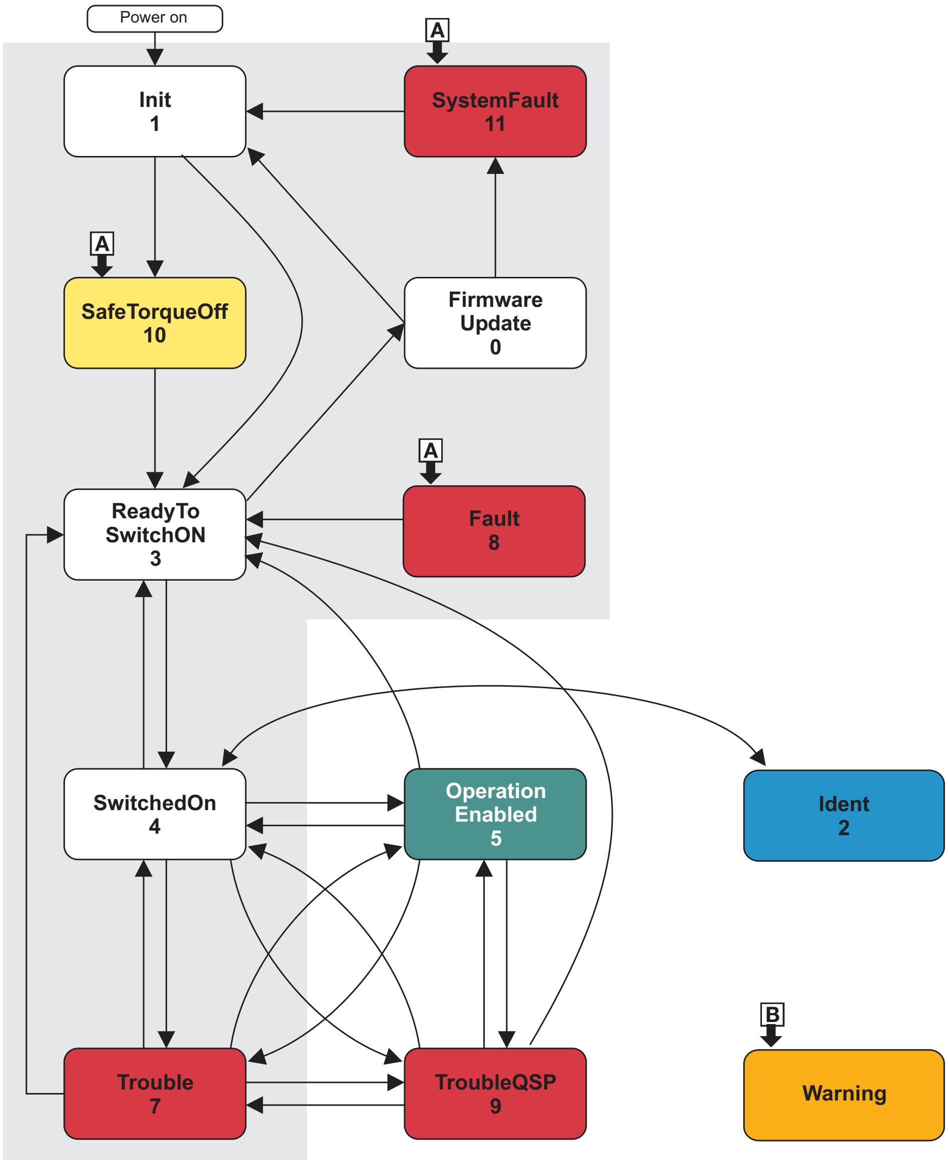
Tip!

The device search function can also be activated via the *toolbar* icon.

4.2 Device state machine and device states

The behaviour of the inverter is mainly determined by the current device status within the device state machine. Which device status is active and which device status is next depends on certain control signals (e.g. for controller inhibit and quick stop) and status parameters.

Device state machine



Grey field: Pulse inhibit

A	Can be reached from all statuses.
B	"Warning" contradicts the definition of a device status. In fact, it is a message which is to call attention to the device status the warning exists for. "Warning" can occur in parallel to other device states.

- The arrows between the device states mark possible state changes.
- The digits stand for the status ID (see table below).

- The change from one status to the other is carried out within a 1-ms cycle. If within this time there are several requests for status changes, the status with the higher priority is processed first (see table below).
- The C00137 displays the current device status.
- C00150 (status word) provides a bit coded representation of the current device status via bits 8 ... 11 (see table below).

ID	Device status (Display in C00137)	Priority	Status bits (Display in C00150)				Meaning
			Bit 11	Bit 10	Bit 9	Bit 8	
0	FirmwareUpdate	-	0	0	0	0	Firmware update function is active
1	Init	-	0	0	0	1	Initialisation active
2	Ident	-	0	0	1	0	Identification active
3	ReadyToSwitchOn	Prio 5	0	0	1	1	Device is ready to start
4	SwitchedOn	Prio 4	0	1	0	0	Device is switched on
5	OperationEnabled	Prio 1	0	1	0	1	Operation
6	-	-	0	1	1	0	-
7	Trouble	Prio 3	0	1	1	1	Trouble active
8	Fault	Prio 7	1	0	0	0	Error active
9	TroubleQSP	Prio 2	1	0	0	1	TroubleQSP is active
10	SafeTorqueOff	Prio 6	1	0	1	0	Safe torque off is active
11	SystemFault	Prio 8	1	0	1	1	System fault active • This device state is not used since the inverter lacks the activating event.

[4-1] Device statuses, priorities, and meaning of the status bits in the status word

4.2.1 FirmwareUpdate

Note!

This function may only be executed by qualified Lenze personnel!

4.2.2 Init

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
OFF	OFF	Init	0	0	0	1

The inverter is in this status immediately after switching on its 24 V supply voltage.

In the "Init" status, the operating system is initialised and all device components (communication module, memory module, power section, etc.) are identified. When identifying the power section, it is checked first if it is switched on or if the required voltage lies within the tolerance zone, respectively.

- The inverter is inhibited, i.e. the motor terminals (U, V, W) of the inverter are deenergised.
- The digital and analog inputs are not yet evaluated at this time.
- The bus systems (CAN, PROFIBUS etc.) do not work yet, i.e. communication is not possible.
- The application is not yet processed.
- The monitoring functions are not active yet.
- The inverter cannot be parameterised yet and no device commands can be carried out yet.

Note!

If the 24V voltage supply is in the valid range (>19V) and the initialisation is finished, the device changes automatically to the "ReadyToSwitchOn" status.

If only the 24V voltage supply is available during the mains connection, the error message "LU: Undervoltage in the DC bus" is also entered into the logbook of the inverter.

4.2.3 Ident

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
(green)	OFF	Ident	0	0	1	0

The inverter has been provided with functions for automatically detecting the motor parameters, the resolver error and the pole position of a synchronous motor. If such an "identification" is active, the inverter is in the "Ident" status.

The "Ident" device status can only be reached from the "SwitchedOn" device status, i.e. the inverter must be inhibited first so that identification can be started afterwards via the corresponding device command:

Device command	Function	Detailed information
C00002/23	Identify motor parameters	▶ Automatic motor parameter identification
C00002/25	Identify resolver error	e ▶ Optimise resolver behaviour
C00002/34	Identify pole position (360°)	▶ Pole position identification

STOP

Stop!

During the identification of the motor parameters or the pole position, the inverter does not respond to setpoint changes or control processes (e.g. speed setpoints, quick stop, torque limitations).

While an identification is being executed,

- the application remains active,
- all system interfaces (IO, bus systems, etc.) remain active,
- error monitoring remains active,
- the inverter is controlled independently of the setpoint sources.

After the identification is completed, the status changes back to "SwitchedOn".

4.2.4 SafeTorqueOff

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
	OFF	SafeTorqueOff	1	0	1	0

Note!

Up to and including version 13.xx.xx the following applies:

This device status is only possible in connection with an integrated safety system and if a power section supply is available!

From version 14.00.00 the following applies:

This device status is only possible in connection with an integrated safety system!

Integrated safety systems with Inverter Drives 8400

Inverters of the 8400 series can be equipped with the integrated "Safe torque off (STO)" safety system.

The integrated safety system can be used on machines for the protection of persons.

The drive function is still carried out by the inverter. The safety system provides safe inputs. If the safety system is activated, it executes control functions according to EN 60204-1 directly in the inverter in case of errors.

Safety state

If the inverter is switched off by the safety system, the device changes to the "SafeTorqueOff" status.

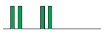
If the safety system deactivates the "Safe torque off (STO)" request, the device changes to the "ReadyToSwitchOn" status.

Detailed information on the integrated safety system can be found in the hardware manual!

The hardware manual contains important notes on the safety system which must be observed!

The hardware manual has been stored in electronic form on the data carrier supplied with the 8400 inverter.

4.2.5 ReadyToSwitchOn

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
	OFF	ReadyToSwitchOn	0	0	1	1

The inverter is in this device status directly after the initialisation has been completed!

- The bus systems are running and the terminals and encoders are evaluated.
- The monitoring functions are active.
- The inverter can be parameterised.
- The application is basically executable.

Note!

- The "ReadyToSwitchOn" state is not only activated after the mains connection, but also after the deactivation of "Trouble", "Fault" or "SafeTorqueOff".
- If C00142 activates the autostart option "Inhibit at device on" (Lenze setting), explicit deactivation of the controller inhibit after mains connection is always required for the inverter to change from the "ReadyToSwitchOn" status to the "SwitchedOn" status.
- If only the 24V voltage supply is available during the mains connection, the error message "LU: Undervoltage in the DC bus" is entered into the logbook of the inverter and the inverter remains in the "ReadyToSwitchOn" status.

Danger!

If the "Inhibit at device on" auto-start option has been deactivated in C00142, the "ReadyToSwitchOn status switches directly to the "SwitchedOn" status after mains connection.

▶ Automatic restart after mains connection/fault... (📖 130)

4.2.6 SwitchedOn

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
	OFF	SwitchedON	0	1	0	0

The drive is in this device status if the DC bus voltage is applied and the inverter is still inhibited by the user (controller inhibit).

- The bus systems are running and the terminals and encoders are evaluated.
- The monitoring functions are active.
- The application is basically executable.

If the controller inhibit is deactivated, the devices changes to the "OperationEnabled" status and the motor follows the setpoint defined by the active application.

Tip!

C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.

Depending on certain conditions, a status change takes place based on the "SwitchedOn" device status:

Change condition	Changeover to the device status
Control bit "EnableOperation" of all control channels = "1" **AND** terminal RFR = HIGH level (controller enable)	OperationEnabled
Control bit "SwitchOn" of a control channel = "0".	ReadyToSwitchOn
Identification requested.	Ident
Undervoltage in the DC bus.	Trouble/Fault (depending on C00600/1)
Error with error response "Trouble" occurs.	Trouble
Error with error response "TroubleQSP" occurs.	TroubleQSP

Related topics:

▶ wCANControl/wMCIControl control words (📖 138)

4.2.7 OperationEnabled

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
(green)	OFF	OperationEnabled	0	1	0	1

The inverter is in this device status if the controller inhibit is deactivated and no error ("Trouble" or "TroubleQSP") has occurred.

If the operation is enabled and the magnetisation in case of servo control (SC) and sensorless vector control (SLVC) has been completed, the motor follows the setpoint determined by the active application.

Depending on certain conditions, a status change takes place based on the "OperationEnabled" device status.

Change condition	Changeover to the device status
Control bit "EnableOperation" of a control channel = "0" **OR** terminal RFR = LOW level (controller inhibit).	SwitchedOn
Control bit "SwitchOn" of a control channel = "0".	ReadyToSwitchOn
Undervoltage in the DC bus.	Trouble/Fault (depending on C00600/1)
Error with error response "Trouble" occurs.	Trouble
Error with error response "TroubleQSP" occurs.	TroubleQSP

Related topics:

▶ wCANControl/wMCIControl control words (📖 138)

4.2.8 TroubleQSP

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
		TroubleQSP	1	0	0	1

This device status will be active as soon as a monitoring mode responds the error response "TroubleQSP" has been parameterised for.

- The drive is decelerated to standstill with torque within the deceleration time parameterised for quick stop independently of the defined setpoint and can be kept there.
- The device status can only be abandoned by acknowledging the error if the error cause is removed.
- When the controller is inhibited, it is possible to jump to the "SwitchedOn" status even during the error status since the controller inhibit function has a higher priority. As long as the error is pending and has not been acknowledged, the status is changed back to the "TroubleQSP" status when the controller is enabled afterwards.

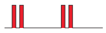
Depending on certain conditions a status change takes place based on the "TroubleQSP" device status.

Change condition	Changeover to the device status
Control bit "SwitchOn" of a control channel = "0".	ReadyToSwitchOn
Control bit "EnableOperation" of all control channels = "1" **AND** terminal RFR = HIGH level (controller enable) **AND** error is reset by the control bit "ResetFault" **AND** no more errors are pending.	OperationEnabled
Control bit "EnableOperation" of a control channel = "0" **OR** terminal RFR = LOW level (controller inhibit) **AND** error is reset by the control bit "ResetFault" **AND** no more errors are pending.	SwitchedOn
A message is active in the system.	Trouble

Related topics:

▸ wCANControl/wMCIControl control words (📖 138)
▸ Basics on error handling in the inverter (📖 719)
▸ Error messages of the operating system (📖 743)

4.2.9 Trouble

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
OFF		Trouble	0	1	1	1

This device status becomes active as soon as a monitoring mode responds for which the error response "Trouble" has been parameterised.

- The motor has no torque (is coasting) when the inverter is inhibited.
- The "Trouble" device status is automatically exited if the error cause has been removed.

Note!

If in C00142 the "Inhibit at trouble" is activated, explicit deactivation of the controller inhibit is required before this status can be abandoned.

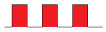
Depending on certain conditions a status change takes place based on the "Trouble" device status.

Change condition	Changeover to the device status
The error cause is no longer active.	ReadyToSwitchOn
Control bit "EnableOperation" of all control channels = "1" **AND** terminal RFR = HIGH level (controller enable) **AND** the message has been cancelled.	OperationEnabled
Control bit "EnableOperation" of a control channel = "0" **OR** terminal RFR = LOW level (controller inhibit) **AND** the message has been cancelled.	SwitchedOn
In the system, there is an error configured on "TroubleQSP". **AND** the message has been cancelled.	TroubleQSP

Related topics:

- wCANControl/wMCIControl control words (📖 138)
- Basics on error handling in the inverter (📖 719)
- Error messages of the operating system (📖 743)

4.2.10 Fault

"DRV-RDY" LED	LED "DRV-ERR"	Display in C00137	Display in status word 1 (C00150)			
			Bit 11	Bit 10	Bit 9	Bit 8
OFF		Fault	1	0	0	0

This device status will be active as soon as a monitoring mode responds the error response "Fault" has been parameterised for.

- The motor has no torque (is coasting) when the inverter is inhibited.
- The error must explicitly be reset ("acknowledged") in order to exit the device status, e.g. by the device command "Reset error" or via the control bit "ResetFault" in the control word *wCanControl* or *wMCIControl*.

Note!

If an undervoltage in the DC bus of the inverter occurs (error message "LU"), the device changes to the "Trouble" status.

An additional error of higher priority leads the device into the "Fault" status.

According to the Device state machine, the device changes to the "ReadyToSwitchOn" status after acknowledging the error although the undervoltage is still available!

If the "Inhibit at fault" auto-start option has been activated in C00142, explicit deactivation of the controller inhibit is required before the status can be abandoned.

Related topics:

▸ wCANControl/wMCIControl control words (📖 138)
▸ Basics on error handling in the inverter (📖 719)
▸ Error messages of the operating system (📖 743)

4.3 Automatic restart after mains connection/fault...

.../Error/undervoltage/loading of the Lenze setting

In C00142, the starting performance of the inverter after mains connection, undervoltage, loading of the Lenze setting as well as a "Trouble" or a "Fault" reset can be parameterised individually:

Auto-start option (C00142)		Lenze setting
Bit 0	"Inhibit at device on" auto-start option	1 ≡ Inhibit is active
Bit 1	Auto-start option "Inhibit at trouble"	0 ≡ Inhibit is not active
Bit 2	Auto-start option "Inhibit at fault"	0 ≡ Inhibit is not active
Bit 3	Auto-start option "Inhibit at undervoltage"	1 ≡ Inhibit is active
Bit 4	Auto-start option "Inhibit at Lenze setting"	1 ≡ Inhibit is active
Bit 5	Reserved	0
Bit 6		
Bit 7		

Note!

In the Lenze setting , automatic restart after mains connection, undervoltage, and loading of the Lenze setting is inhibited.

4.3.1 "Inhibit at device on" auto-start option

The auto-start option "Inhibit at device on" prevents the change to the "SwitchedOn" status after mains connection if the controller is already enabled at mains connection.

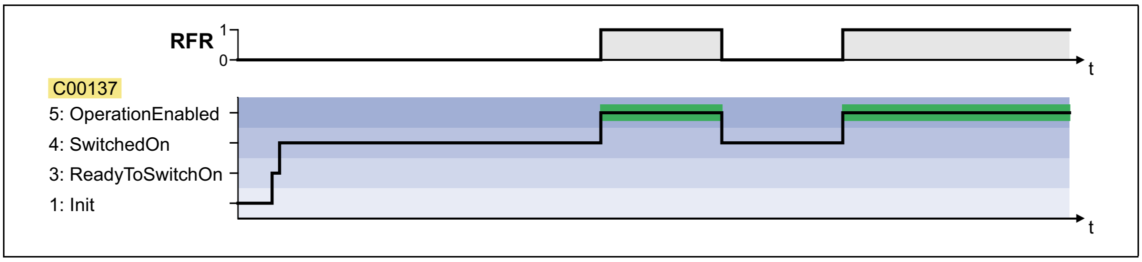
Danger!

If the "Inhibit at device on" auto-start option has been deactivated in C00142, (bit 0 = 0), the motor can directly start to run if the controller is enabled after mains connection!

The following three cases describe the behaviour of the inverter after mains connection depending on whether the controller is enabled and the set auto-start option. Here, it is assumed that after mains connection, no errors and trouble occur in the inverter and the "EnableOperation" control bit in the *wDriveControl* is set to "1".

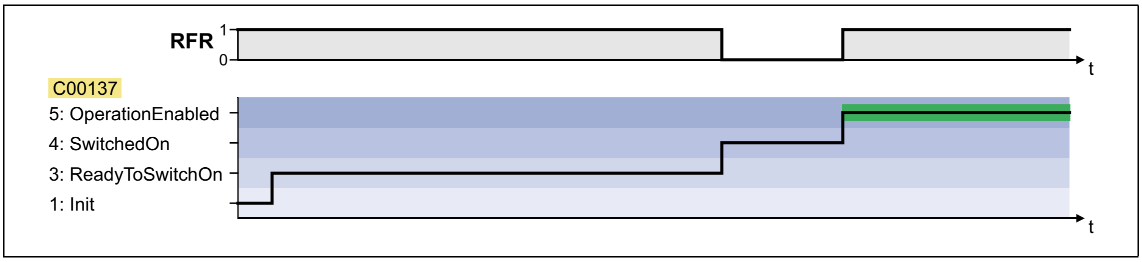
Case 1: No controller enable at mains connection

If the controller is not enabled at mains connection, the inverter remains in the "SwitchedOn" status. Only with the controller enable, the device changes to the "OperationEnabled" status, independent of the set auto-start option:



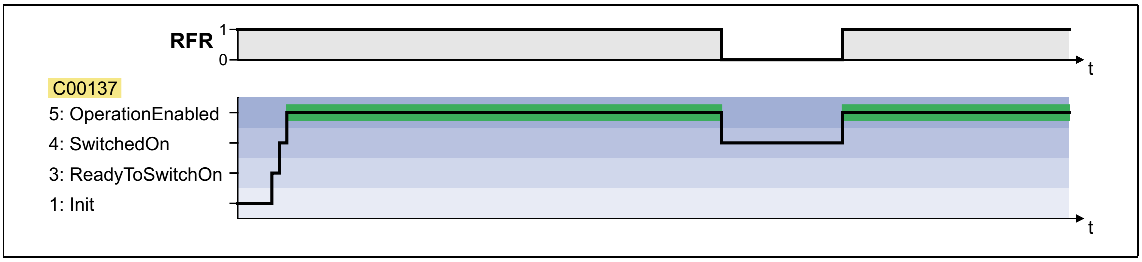
Case 2: Controller enable at mains connection and "Inhibit at device on" activated

If the controller is enabled at mains connection and the auto-start option "Inhibit at device on" is activated, the inverter remains in the "ReadyToSwitchOn" status. For changing to the "SwitchedOn" status, the controller enable must first be deactivated. Only when the controller is enabled again afterwards, the status changes to "OperationEnabled":



Case 3: Controller enable at mains connection and "Inhibit at device on" deactivated

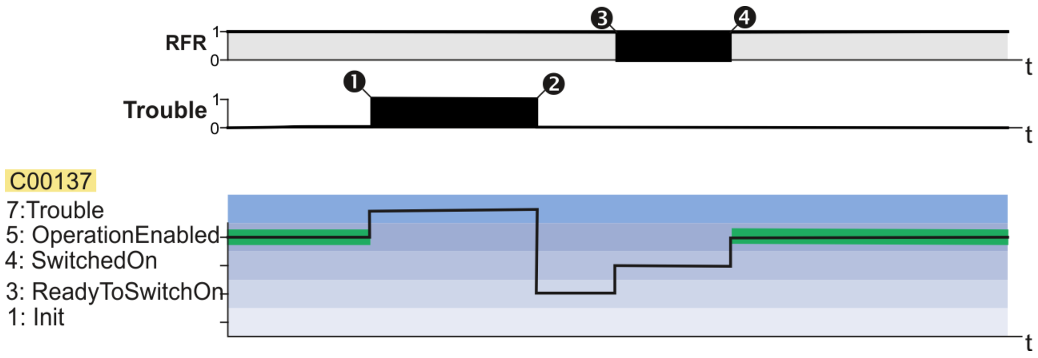
If in C00142 the autostart option "Inhibit at device on" is deactivated (bit 0 = 0), the status first changes from "ReadyToSwitchOn" to "SwitchedOn" and then to "OperationEnabled" after mains connection with an enabled controller:



4.3.2 Auto-start option "Inhibit at trouble"

The auto-start option "Inhibit at trouble" (Bit 1 = 1) does the following: After the state "Trouble", it prevents the system switching to the state "SwitchedOn" if the controller is already enabled after the state "Trouble".

To be able to switch to the "SwitchedOn" state, the controller enable must first be deactivated after the "Trouble" state. Only when the controller is enabled again afterwards does the state change to "OperationEnabled"

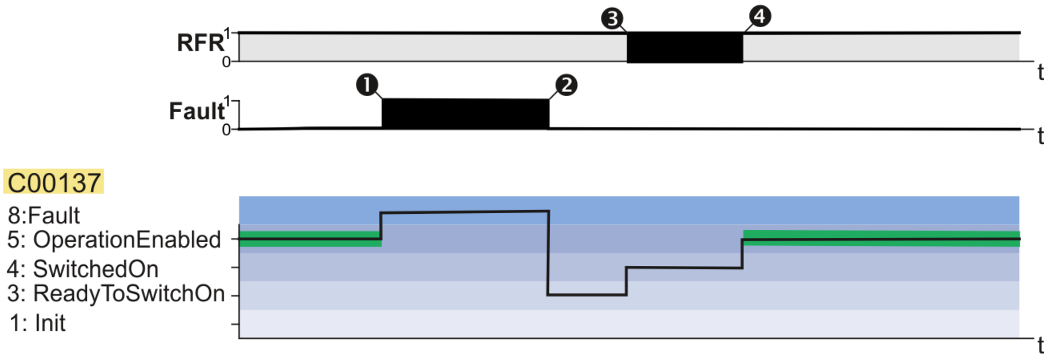


❶ An error configured for Trouble switches the drive into the **Trouble** state.
❷ The cause of Trouble is eliminated.
❸ Controller inhibit via signal change by the user
❹ Controller enable via signal change by the user

4.3.3 Auto-start option "Inhibit at fault"

The auto-start option "Inhibit at trouble" (Bit 2 = 1) does the following: After the state "Fault", it prevents the system switching to the state "SwitchedOn" if the controller is already enabled after the state "Fault".

To be able to switch to the "SwitchedOn" state, the controller enable must first be deactivated after the "Fault" state. Only when the controller is enabled again afterwards does the state change to "OperationEnabled"

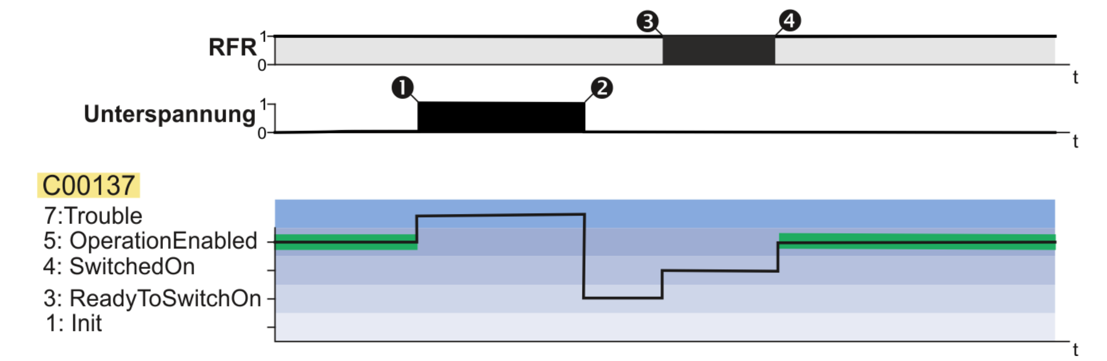


❶ An error configured for Fault switches the drive into the Fault **state.**
❷ The reason for the fault has been eliminated and a fault reset is performed.
❸ Controller inhibit via signal change by the user
❹ Controller enable via signal change by the user

4.3.4 Auto-start option "Inhibit at undervoltage"

The auto-start option "Inhibit at undervoltage" (Bit 3 = 1) prevents the system switching to the state "SwitchedOn" after an instance of undervoltage if the controller is already enabled after the instance of untervoltage.

To be able to switch to the "SwitchedOn" state, the controller enable must first be deactivated after the instance of undervoltage. Only when the controller is enabled again afterwards does the state change to "OperationEnabled"

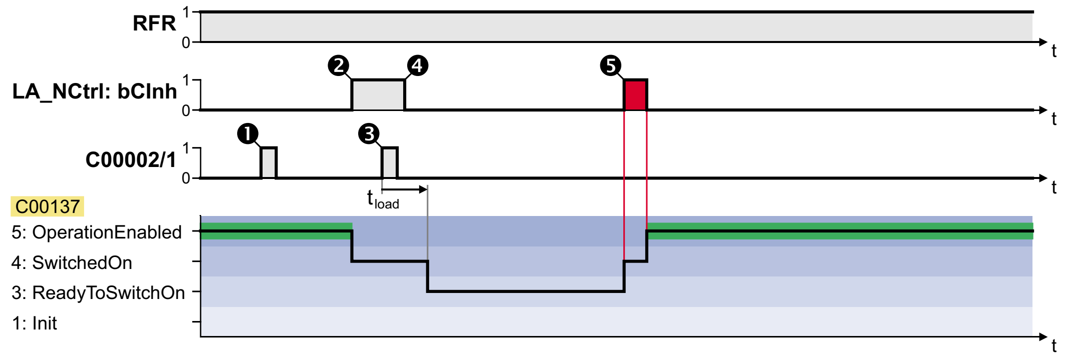


❶ The detection of undervoltage results in the state Trouble.
❷ The mains voltage (DC-bus voltage) returns.
This deactivates the undervoltage detection.
❸ Controller inhibit via signal change by the user
❹ Controller enable via signal change by the user

4.3.5 Auto-start option "Inhibit at Lenze setting"

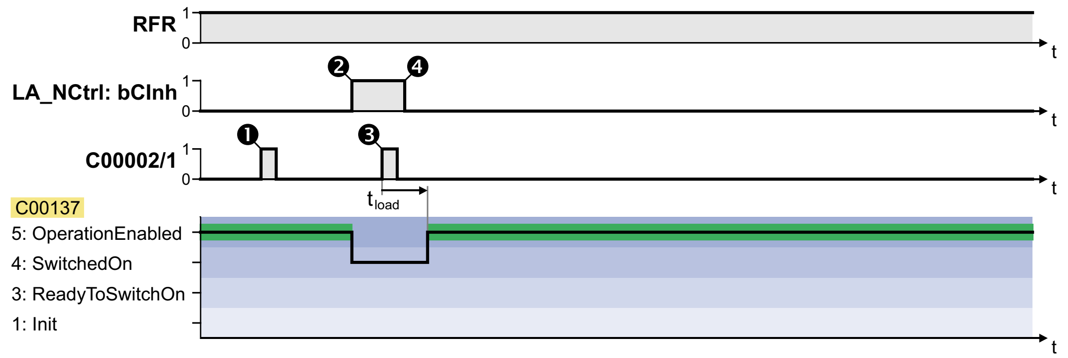
The "Inhibit at Lenze setting" auto-start option configurable via bit 4 of C00142 prevents the change to the "SwitchedOn" status after the Lenze setting has been loaded and the controller is enabled.

For a change to the "SwitchedOn" status, the controller enable must first be deactivated after the Lenze setting has been loaded. Only if the controller is enabled again afterwards, the status changes to "OperationEnabled":



❶ Loading of the Lenze setting not possible since controller inhibit not set.
❷ Controller inhibit via signal change by the user
❸ Loading of the Lenze setting via device command C00002/1 = "1"
❹ Controller enable via signal change by the user
❺ Controller inhibit and subsequent controller enable via signal change by the user
t_{load} = time required for loading the Lenze setting

[4-1] Example 1: Behaviour with activated auto-start option "Inhibit at Lenze setting" (C00142: Bit 4 = "1")

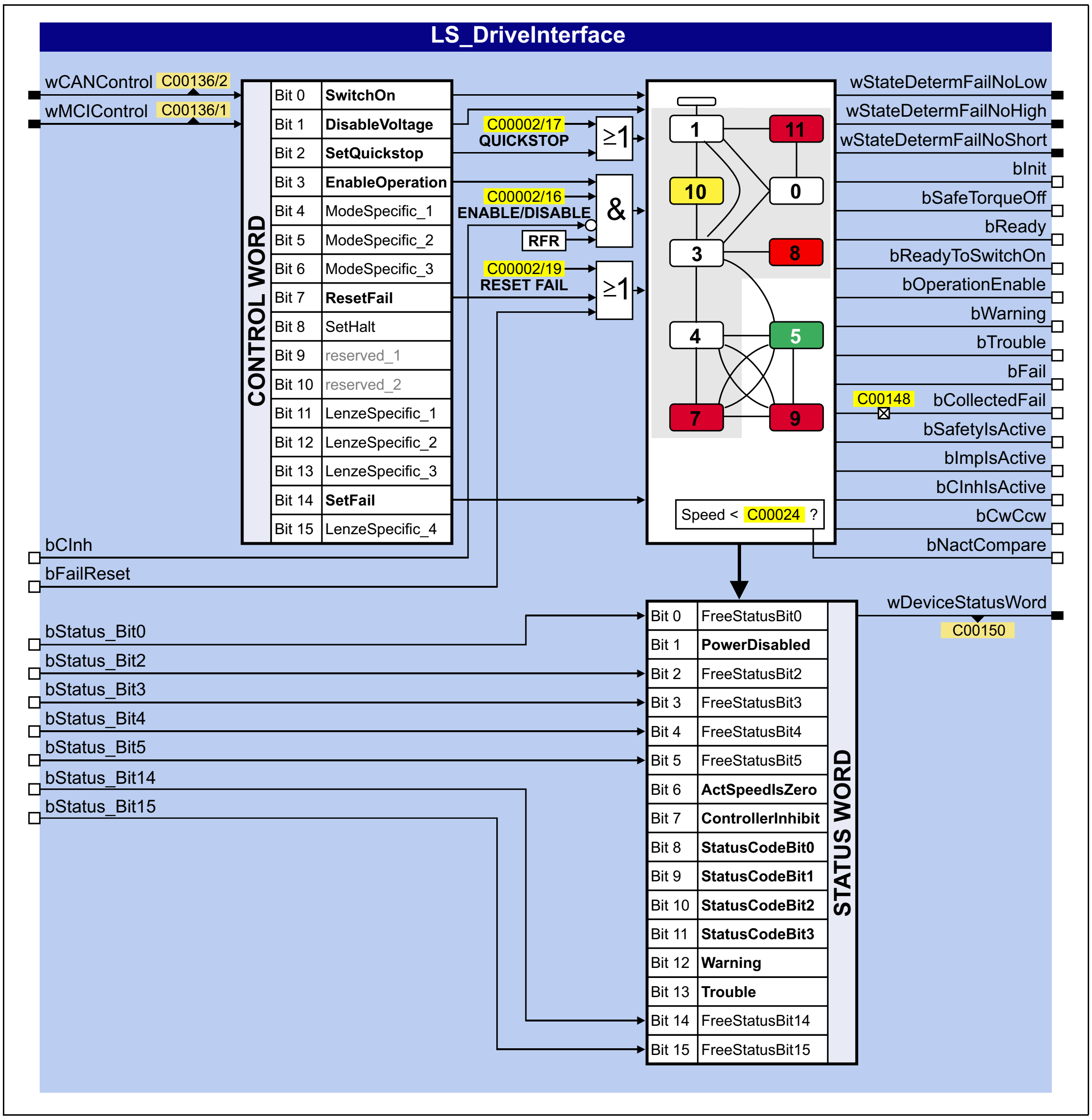


❶ Loading of the Lenze setting not possible since controller inhibit not set.
❷ Controller inhibit via signal change by the user
❸ Loading of the Lenze setting via device command C00002/1 = "1"
❹ Controller enable via signal change by the user
t_{load} = time required for loading the Lenze setting

[4-2] Example 2: Behaviour with deactivated auto-start option "Inhibit at Lenze setting" (C00142: Bit 4 = "0")

4.4 Internal interfaces | "LS_DriveInterface" system block

The **LS_DriveInterface** system block displays the device control in the FB Editor.



inputs

<table>
<tr><th>Designator
DIS code | data type</th><th colspan="2">Information/possible settings</th></tr>
<tr><td>wCANControl
C00136/2 | WORD</td><td colspan="2">Control word via system bus (CAN)
• The inverter controlled by a master control (e.g. IPC) receives its control word by the CANopen system bus interface. The process data word is provided at this input by the upstream port block LP_CanIn1.
• For a detailed description of the individual control bits, see chapter "wCANControl/wMCIControl control words". (138)</td></tr>
<tr><td>wMCIControl
C00136/1 | WORD</td><td colspan="2">Control word via communication module (e.g. PROFIBUS)
• The inverter controlled by a master control (e.g. IPC) receives its control word by a plugged-in communication module. The process data word is provided at this input by the upstream port block LP_MciIn1.
• For a detailed description of the individual control bits, see chapter "wCANControl/wMCIControl control words". (138)</td></tr>
<tr><td rowspan="3">bCInh
C00833/36 | BOOL</td><td colspan="2">▸ Enable/inhibit inverter (113)</td></tr>
<tr><td>FALSE</td><td>Enable inverter: The inverter switches to the "OperationEnabled" device status if no other source for controller inhibit is active.
• C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.</td></tr>
<tr><td>TRUE</td><td>Inhibit inverter (controller inhibit): The inverter switches to the "SwitchedOn" device status.</td></tr>
<tr><td rowspan="2">bFailReset
C00833/37 | BOOL</td><td colspan="2">▸ Reset error message (747)</td></tr>
<tr><td>FALSE↗TRUE</td><td>The current error is reset.</td></tr>
<tr><td>bStatus_Bit0
bStatus_Bit2
bStatus_Bit3
bStatus_Bit4
bStatus_Bit5
bStatus_Bit14
bStatus_Bit15
C00833/38 ... 44 | BOOL</td><td colspan="2">Freely assignable bits in the status word of the inverter
• You can use these bits for returning information to the master control (e.g. IPC).</td></tr>
<tr><td>bFree_1
bFree_2
bFree_3
bFree_4
C00833/45 ... 48 | BOOL</td><td colspan="2">Reserved inputs have no function</td></tr>
</table>

outputs

Designator DIS code | data type	Value/meaning	

wDeviceStatusWord
C00150 | WORD
	Status word of the inverter (based on DSP-402)
	• The status word contains all information relevant for controlling the inverter.
	• The status word is sent as a process data word to the master control via a port block:
	  • Port block LP_CanOut1 when the CANopen system bus interface is used or
	  • Port block LP_MciOut when a plugged-in communication module is used (e.g. PROFIBUS).
	• For a detailed description of each status bit see chapter "wDeviceStatusWord status word". (📖 140)	

wStateDetermFailNoLow
WORD
	Display of the status determining error (32-bit error number, Low-Word)
	• If the "Use I16BitFailNo" (bit 15 = "1") option is activated in C00148, the short 16-bit error number (*wStateDetermFailNoShort*) is provided via this output as well.
	  • In this case, the *wStateDetermFailNoHigh* output is "0".
	  • Advantage: The bus transfer of the error numbers is possible via a data word without changing the interconnection of the technology application.	

wStateDetermFailNoHigh WORD	Display of the status determining error (32-bit error number, High-Word)	
wStateDetermFailNoShort WORD	Display of the status determining error (16-bit error number)	
bInit BOOL	TRUE	"Init" device state is active
bSafeTorqueOff BOOL	TRUE	"SafeTorqueOff" device state is active • *bSafeTorqueOff* will only be output if the DC bus is loaded. • The STO feedback signal is always pending at the X80/GO terminal.
bReady BOOL	TRUE	"SwitchedOn" device state is active
bReadyToSwitchOn BOOL	TRUE	"ReadyToSwitchOn" device state is active
bOperationEnable BOOL	TRUE	"OperationEnabled" device state is active
bWarning BOOL	TRUE	A warning is indicated
bMessage BOOL	TRUE	"Trouble" device state is active
bFail BOOL	TRUE	"Fault" device state is active
bCollectedFail BOOL	TRUE	Group error: A device status according to the group error configuration in C00148 has occurred, the drive is not able to follow the setpoint selection.
bSafetyIsActive BOOL	TRUE	In preparation
bImpIsActive BOOL	TRUE	Pulse inhibit is active
bCInhIsActive BOOL	TRUE	Controller inhibit is active
bCwCcw BOOL	FALSE	Motor rotates in CW direction
	TRUE	Motor rotates in CCW direction
bNactCompare BOOL	TRUE	During open-loop operation: Speed setpoint < comparison value (C00024, C00025/1)
		During closed-loop operation: actual speed value < comparison value (C00024, C00025/1)

Option "Lock bFail at TroubleQSP"

The TroubleQSP device status gets active as soon as a monitoring function responds that has been parameterised for the "TroubleQSP" error response. Since the *bFail* status output is not set in this case, it is not recognisable after pulse inhibit due to e.g. automatic brake operation (as well for a higher-level control), why the drive is standing und does not start when the setpoint is selected. Only after an error reset, a setpoint is accepted again.

From version 02.00.00: If the "Lock bFail at TroubleQSP" option is activated (bit 14 = "1") in C00148, the *bFail* status output is also set to TRUE if the device status is TroubleQSP.

4.4.1 wCANControl/wMCIControl control words

The inverter is controlled by a master control (e.g. IPC) via the *wCanControl* or *wMCIControl* control word, respectively.

- *wCANControl*: Control word via system bus (CAN)
  - The process data word is provided at the *wCanControl* input via the upstream LP_CanIn1 port block.
  - Display parameter: C00136/2
- *wMCIControl*: Control word via a plugged-in communication module (e.g. PROFIBUS)
  - The process data word is provided at the *wMCIControl* input via the upstream LP_MciIn1 port block.
  - Display parameter: C00136/1
- The bit assignment for the *wCanControl*/*wMCIControl* control words can be seen from the table below.

Note!

The assignment of bits 11 ... 13 and bit 15 depends on the technology application selected in C00005!

- See description of the corresponding technology application.

Bit	Name	Function
Bit 0	SwitchOn	1 ≡ Change to the "SwitchedOn" device status • This bit has to be set in the CAN **AND** MCI control word in order that the drive changes to the "SwitchedOn" device state. In order to reach the "ReadyToSwitchOn" state, it is sufficient to set the bit to 0 in of the two control words.
Bit 1	DisableVoltage	1 ≡ Inhibit inverter control (pulse inhibit)
Bit 2	SetQuickStop	Activate quick stop (QSP) ▸ Activate/deactivate quick stop (🕮 114)
Bit 3	EnableOperation	1 ≡ Enable inverter (RFR) • This bit must be set in CAN **AND** in the MCI control word, otherwise the controller will be inhibited.
Bit 4	ModeSpecific_1	Reserved (currently not assigned)
Bit 5	ModeSpecific_2	
Bit 6	ModeSpecific_3	
Bit 7	ResetFault	1 ≡ Reset fault (trip reset) • Acknowledge error message (if the error cause has been eliminated).
Bit 8	SetHalt	1 ≡ Activate stop function • Stop drive via stopping ramp (in preparation).
Bit 9	reserved_1	Reserved (currently not assigned)
Bit 10	reserved_2	
Bit 11	LenzeSpecific_1	Assignment depends on the selected technology application • See description of the corresponding technology application.
Bit 12	LenzeSpecific_2	
Bit 13	LenzeSpecific_3	
Bit 14	SetFail	1 ≡ Set error (trip set)
Bit 15	LenzeSpecific_4	Assignment depends on the selected technology application • See description of the corresponding technology application.

Tip!

If a bus control is not wanted (e.g. in case of control via terminals):

Connect both control word inputs with the *wDriveCtrl* output signal of the LS_ParFix system block. This output signal has the fixed value "9", which corresponds to the following assignment:

- Bit 0, SwitchOn = 1
- Bit 3, EnableOperation = 1
- All others: 0

4.4.2 wDeviceStatusWord status word

The *wDeviceStatusWord* status word provided by the control system contains all information relevant for controlling the inverter.

- The status word is sent as a process data word to the master control via a port block:
  - The **LP_CanOut1** port block if "CAN on board" is used or
  - the **LP_MciOut1** port block if a plugged-in communication module is used (e.g. PROFIBUS).
- Display parameter: C00150
- The bit assignment of the *wDeviceStatusWord* status word can be seen from the table below.

Bit	Name	Status
Bit 0	FreeStatusBit0	Free status bit 0
Bit 1	PowerDisabled	1 ≡ Inverter control inhibited (pulse inhibit is active)
Bit 2	FreeStatusBit2	Free status bit 2 (not assigned, freely assignable)
Bit 3	FreeStatusBit3	Free status bit 3 (not assigned, freely assignable)
Bit 4	FreeStatusBit4	Free status bit 4 (not assigned, freely assignable)
Bit 5	FreeStatusBit5	Free status bit 5 (not assigned, freely assignable)
Bit 6	ActSpeedIsZero	During open-loop operation: 1 ≡ Speed setpoint < Comparison value (C00024)
		During closed-loop operation: 1 ≡ Actual speed value < Comparison value (C00024)
Bit 7	ControllerInhibit	1 ≡ Inverter is inhibited (controller inhibit is active)
Bit 8	StatusCodeBit0	Bit coded display of the active device status ▸ Device state machine and device states (see table [4-1])
Bit 9	StatusCodeBit1	
Bit 10	StatusCodeBit2	
Bit 11	StatusCodeBit3	
Bit 12	Warning	1 ≡ a warning is indicated
Bit 13	Trouble	1 ≡ Inverter is in the "Trouble" device status • E.g. if an overvoltage has occurred.
Bit 14	FreeStatusBit14	Free status bit 14 (not assigned, freely assignable)
Bit 15	FreeStatusBit15	Free status bit 15 (not assigned, freely assignable)

4.5 Energy saving mode

This function extension is available from version 17.00.00 onwards!

In energy saving mode, the energy demand of the inverter can be adapted to the most diverse environments and applications.

Via C1704, various functions can be utilised in a user-defined fashion, bringing the energy consumption of the inverter down to a minimum value:

- inhibiting the power output stages (controller inhibit)
- entering the energy saving mode using quick stop
- switching off the LEDs
- switching off all outputs

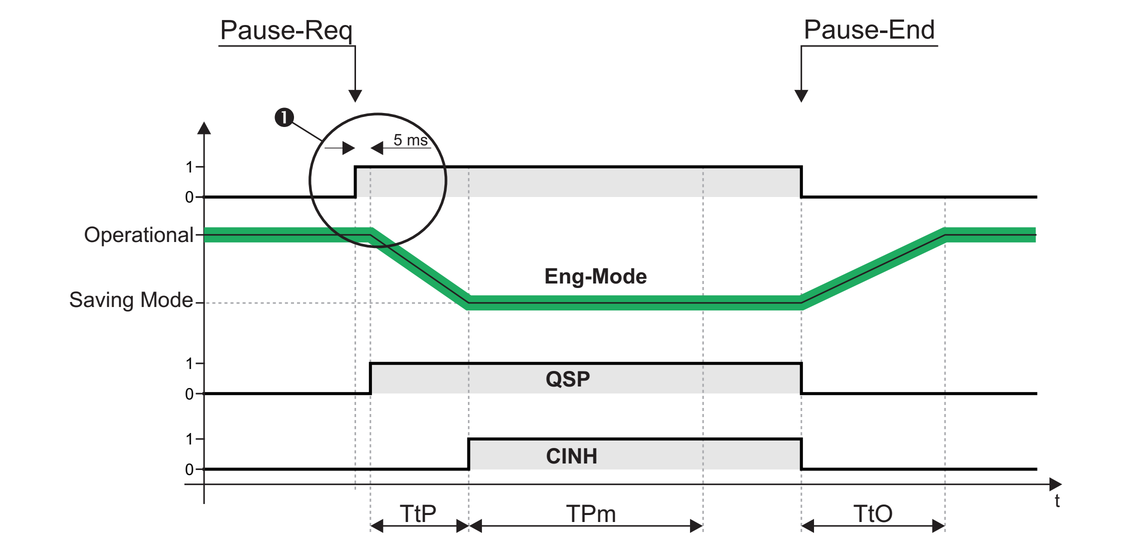
If the use of the energy saving mode is not desired, C1704 provides the possibility of inhibiting this operating status.

The functions for the energy saving mode provide the basis for implementing the **PROFIenergy** PROFINET profile.

Tip!

Detailed information about the **PROFIenergy** PROFINET profile can be obtained from the PROFINET specifications.

Activating / deactivating the energy saving mode



❶	To transfer the information to the FB application, the operating system requires a delay time of 5 ms.
TtP	Time to pause (C01702/1) Time required for entering the energy saving mode.
TPm	Time pause min (C01701/1) Minimum time for which the inverter is to remain in energy saving mode.
TtO	Time to operate (C01703/1) Time required for exiting the energy saving mode

The energy saving mode is activated via the **PROFIenergy** PROFINET profile as follows:

1. Via a "Pause-Req" command, entry into the energy saving mode is requested.
   - At the same time, a dead time is transferred with the command.
2. If the sum of the times set in C1701/1, C1702/1, and C1703/1 is smaller than the dead time requested, the inverter starts entering the energy saving mode.
3. Via a "Pause-End" command, this operating status can be exited again.

5 Motor control (MCTRL)

This chapter provides information on the parameter setting of the inverter's internal motor control. The 8400 TopLine inverter both supports synchronous and asynchronous motors.

Topics:

Basic settings:

- Motor selection/Motor data
- Selecting the control mode
- Defining current and speed limits

Description of the motor control types:

- V/f characteristic control (VFCplus)
- V/f characteristic control - energy-saving (VFCplusEco)
- V/f control (VFCplus + encoder)
- Sensorless vector control (SLVC)
- Sensorless control for synchronous motors (SLPSM)
- Servo control (SC)

Parameterisable additional functions:

- Selection of switching frequency
- Operation with increased rated power
- Current-dependent stator leakage inductance Ppp(I)
- Flying restart function
- DC-injection braking
- Slip compensation
- Oscillation damping
- Phase sequence reversal for correcting misconnected UVW motor phases
- Field weakening for synchronous motors

Further topics:

- Position control/additive speed specification
- Braking operation/brake energy management
- Monitoring

Internal interfaces (process signals):

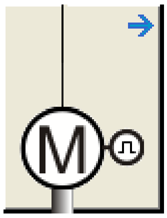
- Internal interfaces | System block "LS_MotorInterface"
- Internal status signals | System block "LS_DeviceMonitor"

5.1 Motor selection/Motor data

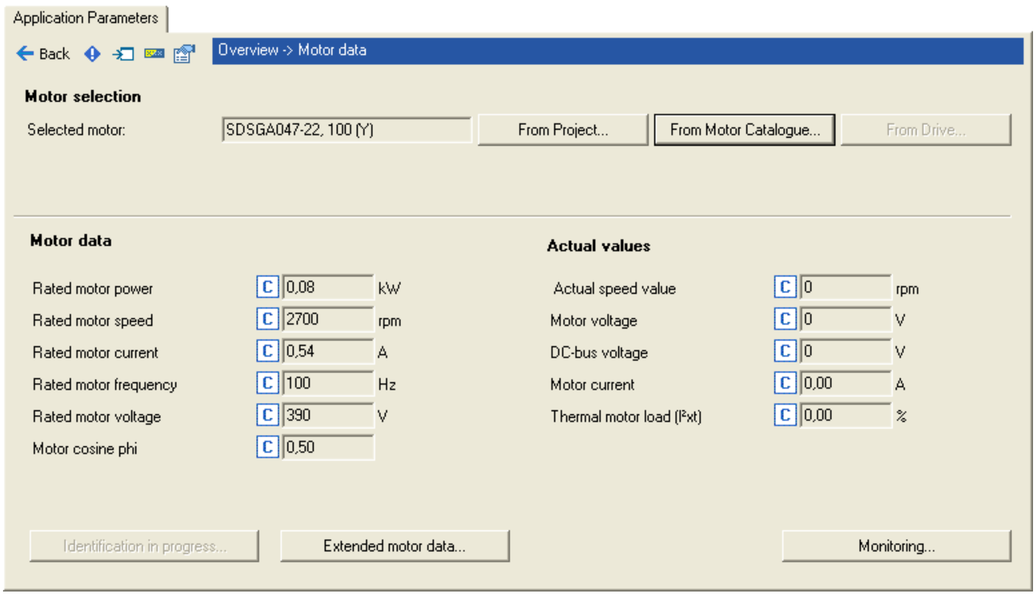
The motor data term comprises all parameters that only depend on the motor and that only characterise the electrical behaviour of the machine. The motor data are independent of the application in which the inverter and the motor are used.

Proceed as follows to open the dialog for parameterising the motor data:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the following button:



Parameterisation dialog in the »Engineer«



- Via the **From Motor Catalogue** button, the motor catalogue can be opened to select another motor. ▸ Selecting a motor from the motor catalogue in the »Engineer« (📖 149)
- Via the **From inverter**... button, the motor data set in the inverter can be copied to the »Engineer« when an online connection has been established.
- When an online connection has been established to the inverter, the **Identification in progress...** button serves to automatically identify different motor data. ▸ Automatic motor data identification (📖 151)
- The **Encoder/feedback system...** button serves to get to the settings for the encoder/feedback system, if available. ▸ Encoder/feedback system (📖 330)

Stop!

Motors with electronic nameplates (ETS) must not be operated on 8400 inverters!

Note!

In the future, Lenze motors will only be equipped with motor temperature sensors PT1000. This is taken into account in the current motor catalogue for »EasyStarter« or »Engineer«.

When motors are replaced in existing plants or in new plants with previous applications, please check whether a motor with KTY or PT1000 is used (motor nameplate or order designation). If a motor with PT1000 is used, adapt the parameterisation of the device.

- Up to version V16.xx.xx, you must use the Specific characteristic for the motor temperature sensor.
- From version V17.00.00 onwards, you can select PT1000 in C01190.

Note!

Sensorless vector control (SLVC) and sensorless control for synchronous motors (SLPSM) in particular requires the motor data parameters to be set. The motor data comprise the data of the motor nameplate and the data of the motor equivalent circuit.

If the motor has been selected via the motor catalogue of the »Engineer« or the motor data have been adapted offline using the »Engineer«, all motor data must then be copied to the inverter and saved power-failure-proof to the memory module (device command: C00002/11) when an online connection has been established.

Note!

Setting a rated motor frequency with a decimal position

If the motor has a rated motor frequency with a decimal position (e.g. motor nameplate data "23.7 Hz"):

- Up to and including version 02.xx.xx, the following motor nameplate data must be increased by a factor of 10:
  - C00089: Rated motor frequency
    (a value of "23.7 Hz", for instance, must be increased to 237 Hz.)
  - C00081: Rated motor power
  - C00087: Rated motor speed
  - C00090: Rated motor voltage
- From version 12.00.00, the rated motor frequency must be entered without decimal position to clearly identify the right motor type. (a value of "23.7 Hz", for instance, must be set to 23 Hz in C00089)
  - C01000 displays the set motor type (ASM or PSM).
  - C00969 displays the set number of motor pole pairs.
- From version 14.00.00 onwards, the motor type (ASM or PSM) can also be set manually in C01001/1 for a clear identification.

Note!

If a rated motor speed is set for an asynchronous motor (ASM) in C00087 which corresponds to a very high slip speed, the number of pole pairs is identified incorrectly. Moreover, a synchronous motor (PSM) is detected instead of an asynchronous motor (ASM) at certain slip speeds. Detailed information and possible remedies can be found here:

▶ Application notes for asynchronous motors with high slip speed (📖 158)

Motor data

In the parameterisation dialog, the data of the motor nameplate for the selected motor are displayed under "Motor data".

Parameters	Information
C00081	Rated motor power
C00087	Rated motor speed
C00088	Rated motor current
C00089	Rated motor frequency
C00090	Rated motor voltage
C00091	Motor cos φ

Actual values

When an online connection to the inverter has been established, the following actual values are displayed in the parameterisation dialog under "Actual values":

Parameters	Information
C00051	Actual speed value
C00052	Motor voltage
C00053	DC-bus voltage
C00054	Motor current
C00066	Thermal motor load (I2xt)
Greyed out = display parameter	

Adapting motor data manually

If a third party manufacturer's motor is used, the displayed motor data can exactly be adapted to the real motor by clicking the **From project**... button and selecting the "Own motor settings" entry from the **Motor selection** dialog box afterwards. For this purpose, the data of the motor nameplate and the equivalent circuit diagram must be available.

Tip!

For a better concentricity factor, we recommend to perform motor parameter identification of the third party manufacturer's motor first. The motor parameters can be manually adapted afterwards.

Improving the concentricity factor includes

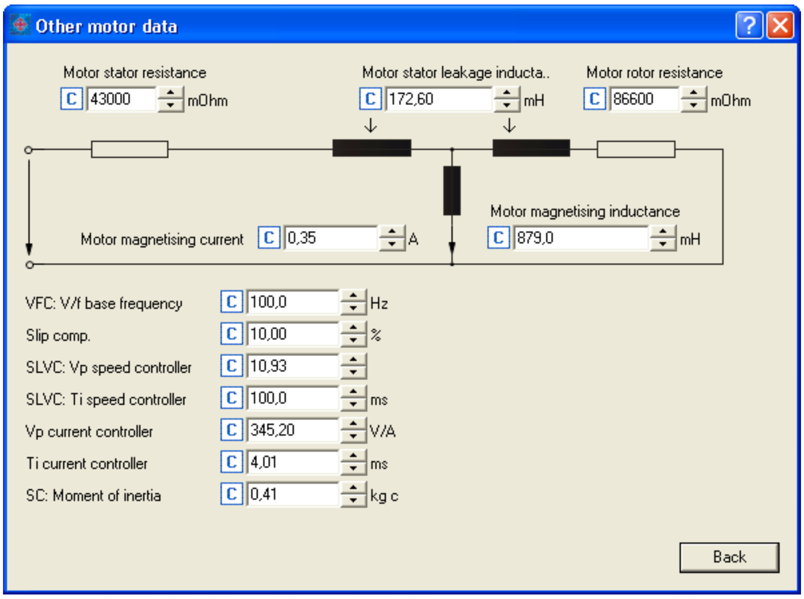
- the adjustment of the inverter error characteristic to the drive system and
- the knowledge of the motor cable resistance.

Both factors are determined in the course of motor parameter identification.

▶ Automatic motor data identification (📖 151)

Other motor data

Click the **Other motor data...** button and go to the *Other motor data* dialog box including the motor equivalent circuit (in the following for an asynchronous motor):

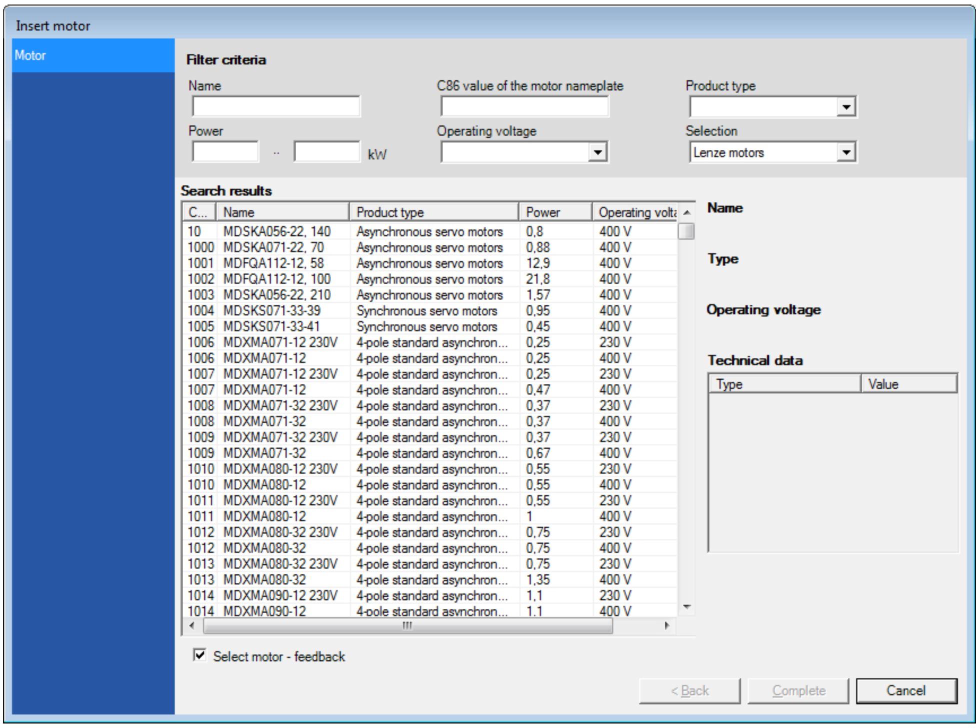


Parameters	Info	ASM	PSM
C00084	Motor stator resistance	●	●
C00085	Motor stator leakage inductance	●	●
C00082	Motor rotor resistance	●	
C00095	Motor magnetising current	●	
C00092	Motor magnetising inductance	●	
C00015	VFC: V/f base frequency	●	●
C00021	Slip compensation	●	
C00070/1	SLVC: Vp speed controller	●	
C00071/1	SLVC: Ti speed controller	●	
C00075	Vp current controller	●	●
C00076	Ti current controller	●	●
C00273	Moment of inertia	●	●
C00016	VFC: Vmin boost	●	●
C00070/2	SC: Vp speed controller	●	●
C00071/2	SC: Ti speed controller	●	●
C00070/3	SLPSM: Vp speed controller	●	●
C00071/4	SLPSM: Ti speed controller	●	●
C00072	SC: Tdn speed controller	●	●
C00011	Appl.: Reference speed	●	●
C00022	Imax in motor mode	●	●
C00966	VFC: Time const. slip comp.	●	
C00982	VFC-ECO: Voltage reduction ramp	●	
C00073/1	VFC: Vp Imax controller	●	●
C00077	SC: Vp field controller	●	
C00078	SC: Tn field controller	●	

- From version 12.00.00 onwards, C01000 displays the set motor type (ASM or PSM).
- Generally, a synchronous motor without speed feedback can also be operated with the V/f characteristic control (VFCplus) control mode. The parameters for this control mode (e.g. V/f base frequency) thus also have an according influence on synchronous motors.

5.1.1 Selecting a motor from the motor catalogue in the »Engineer«

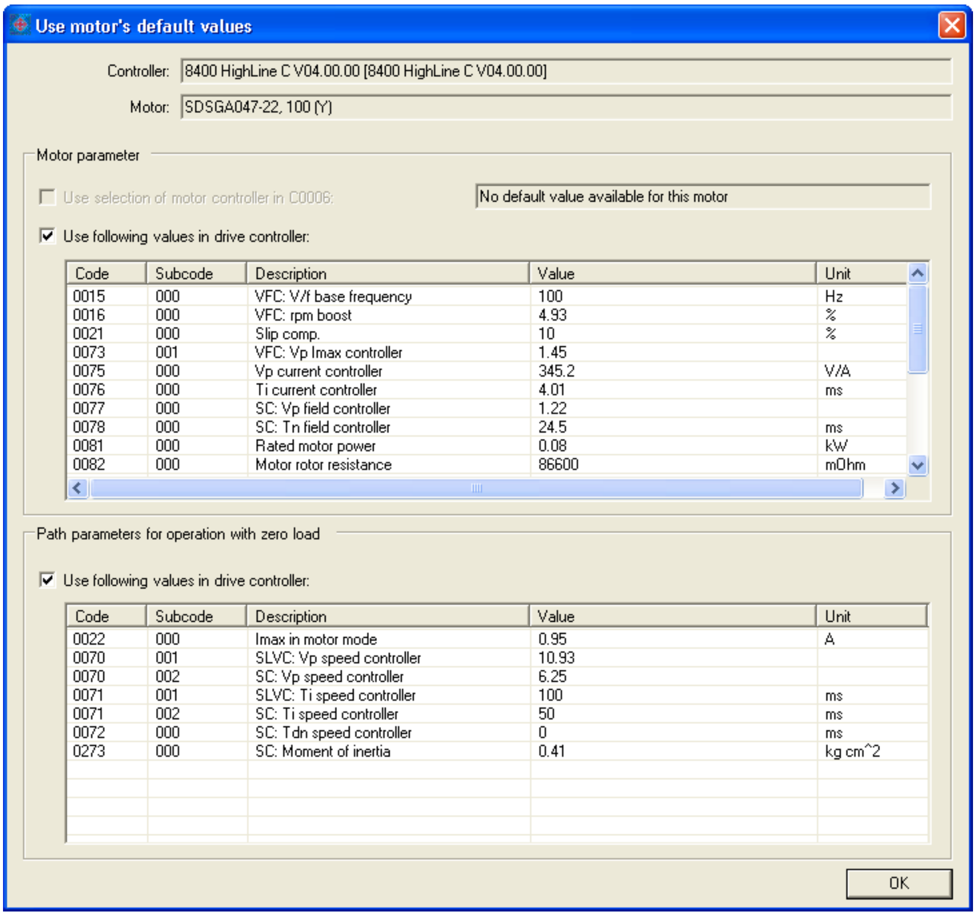
If you, when inserting the inverter into the project in the dialog step "Other components", put a checkmark in the control field **Motor**, you can select as a further dialog step the motor for the inverter from the motor catalogue:



- Alternatively, the motor can be inserted into the project at a later time via the **Insert a component** command.
- Go to the **Application parameters** tab in the *Overview* ➔ *Motor data* dialog level and click the **From motor catalogue...** button to also reach the motor catalogue for the selection of another motor.

Accepting the default values of the motor

If a motor is selected from the motor catalogue at a later time, the *Use motor's default values* dialog box is displayed afterwards which includes all motor data of the selected motor. Please select here which of the default values are to be copied to the inverter:



- The listed motor parameters are already optimally preset for the selected Lenze motor. An adaptation is not required.
- The "plant parameter" term comprises all parameters that result from the combination of motor and load. These characterise the transfer behaviour of the entire controlled system.
  - The plant parameters depend on the application in which the inverter and motor are used.
  - When a Lenze motor is selected in the »Engineer«, plant parameters are suggested for this motor for a load-free operation.

Tip!

If a third party manufacturer's motor is used, select a Lenze motor from the motor catalogue first which is similar in terms of current, voltage and speed rating. Adapt the preselected motor data exactly to the real motor afterwards.

5.1.2 Automatic motor data identification

The motor parameter identification serves to automatically identify the inverter characteristic, the influences of the motor cable and various motor parameters.

Note!

We strongly recommend motor parameter identification before the initial commissioning of the sensorless vector control (SLVC) and the sensorless control for synchronous motors (SLPSM).

Automatically identified motor parameters

Parameters	Info	ASM	PSM
C00015	V/f base frequency	●	●
C00016	V_{min} boost	●	●
C00021	Slip compensation	●	
C00082	Motor rotor resistance	●	
C00083	Motor rotor time constant	●	
C00084	Motor stator resistance	●	●
C00085	Motor stator leakage inductance	●	●
C00092	Motor magnetising inductance	●	
C00095	Motor magnetising current	●	

Automatic calculation of the current controller and field controller parameters

From version 12.00.00 onwards: Following successful motor parameter identification, the current controller and field controller parameters listed in the below table are calculated as well.

- If these parameters are not to be calculated, bit 4 of C02865/1 must be set to "1".

Parameters	Info	ASM	PSM
C00073/1	VFC: Vp Imax controller	●	●
C00075	Vp current controller	●	●
C00076	Ti current controller	●	●

OPTIONAL: Automatic calculation of the speed controller parameters

From version 12.00.00 onwards: Following successful motor parameter identification, the speed controller parameters listed in the below table can also be calculated automatically.

- If these parameters are to be calculated, bit 6 of C02865/1 must be set to "5".

Parameters	Info	ASM	PSM
C00070/1	SLVC: Vp speed controller	●	
C00071/1	SLVC: Ti speed controller	●	
C00070/3	SLPSM: Vp speed controller	●	●
C00071/3	SLPSM: Ti speed controller	●	●

- Care must be taken that the constant mass inertia of the drive (mass inertia of motor, gearbox, shaft and constant load) is entered as exactly as possible in C00273 to calculate the speed controller parameters as dynamically as possible.
  - Mass inertias that are not constant (e.g. changing loads of reels or different loads of hoists) must be entered in C00919/1.
  - If the mass inertia set in C00273 is too low, the speed controller parameters are calculated less dynamically.
  - If the mass inertia set in C00273 is too high, speed controller operation is unstable.
- If the mass inertia in C00273 is set to "0", the setting of bit 5 in C02865 has no effect on the speed controller parameter calculation. In this case, the speed controller parameters will not be calculated automatically after motor parameter identification.

OPTIONAL: Automatic calculation of other controller parameters

From version 12.00.00 onwards: Following successful motor parameter identification, the controller parameters listed in the below table can also be calculated automatically.

- If these parameters are to be calculated, bit 6 of C02865/1 must be set to "6".

Parameters	Info	ASM	PSM
C00011	Appl.: Reference speed	●	●
C00022	Imax in motor mode	●	●
C00497	Nact filter time constant	●	●
C00966	VFC: Time const. slip comp.	●	
C00982	VFC-ECO: Voltage reduction ramp	●	

Automatic calculation of the field weakening controller parameters for ASM servo control

From version 15.00.00, the field weakening controller parameters are automatically calculated as well after a successful motor parameter identification. Moreover, the special functions of the internal motor control listed in the table are activated.

- If these parameters are not to be calculated/activated, bit 6 has to be set to "1" in C02864/1.

Parameters	Info	ASM	PSM
C00577	SC: Vp field weakening controller	●	
C00578	SC: Tn field weakening controller	●	
Activation of special functions of the internal motor control			
C02864/1	Bit 5 is set to "1" → Field/field weakening controllers are called in a 500-µs cycle.*	●	
	Bit 7 is set to "1" → In case of servo control for asynchronous motors, the rated motor speed C00087 is assumed for the calculation of the slip from the nameplate (C02879/Bit 1 = "0" for a warm machine.* ▸ Slip calculation from motor equivalent circuit diagram data (📖 261)	●	
* If bit 6 is set to "1" in C02864/1, bit 5 and bit 7 are reset to "0".			

Principal sequence of the motor parameter identification

1. The motor stator resistance (C00084) is measured.
2. The inverter error characteristic is measured.
3. The motor stator leakage inductance (C00085) is measured.
4. The V/f base frequency (C00015) is calculated.
5. The slip compensation (C00021) is calculated.
6. The V_{min} boost (C00016) is detected.
7. Only for asynchronous motors:
   The motor magnetising inductance (C00092) and the motor rotor resistance (C00082) are measured.
   The motor magnetising current (C00095) is measured.

   From version 12.00.00:
8. If C02865/1 - bit 4 = "0" (Lenze setting):
   The current controller and field controller parameters are calculated. (See table.)
9. If C02865/1 - bit 5 = "1":
   The speed controller parameters are calculated. (See table.)
10. If C02865/1 - bit 6 = "1":
    Other controller parameters are calculated. (See table.)

    From version 15.00.00:
11. If C02864/1 - bit 6 = "0" (Lenze setting):
    The field weakening controller parameters are calculated and certain special functions of the internal motor control are activated. (See table.)

Selection of the identification mode

From version 10.00.00 onwards, two identification modes are available in C02867/1:

- "1: Basic identification" (previous mode)
  - Only for asynchronous motors
  - Duration approx. 30 s
- "2: extended identification"
  - Stands out due to increased accuracy of the determined motor parameters.
  - Also supports synchronous motors and asynchronous motors with a power of more than 11 kW.
  - Duration approx. 80 s

Tip!

In the Lenze setting, a setting of "0: automatic" is selected in C02867/1. This setting ensures that the inverter automatically selects the optimum procedure for motor parameter identification.

Preconditions for executing the motor parameter identification

- The motor parameter identification must be carried out when the motor is cold!
- The load machine may remain connected. Holding brakes, if present, may remain in the braking position.
- In case of a synchronous motor:
  - The shaft must be able to rotate freely (must not be locked).
  - An identification is only possible with the extended motor parameter identification.
- For an asynchronous motor from a power of 11 kW:
  An identification is only possible with the extended motor parameter identification.
- The inverter has to be ready for enable (has to be in the "SwitchedOn" state).
  Inhibit the inverter if it is enabled, e.g. via the C00002/16 device command or a LOW signal at the X4/RFR terminal.
- The motor has to be at standstill.

Required parameter settings before the motor parameter identification is executed

- For an extended motor parameter identification, a motor control suitable for the motor has to be set in C00006. When the identification starts, this setting determines whether an asynchronous motor or a synchronous motor has to be identified. ▶ Selecting the control mode (📖 159)
- The motor parameters listed in the table below are excluded from automatic identification and must therefore be adapted to the used motor before motor parameter identification is carried out (see motor nameplate).

Parameters	Information
C00081	Rated motor power
C00087	Rated motor speed
C00088	Rated motor current (according to the connection method ⅄/△)
	NoteThe amplitude of the rated motor current (C00088) is injected to identify the stator resistance. If the rated motor current amounts to less than 60 % of the rated inverter current, at least 60 % of the rated inverter current will be injected to ensure sufficient motor parameter identification accuracy.
C00089	Rated motor frequency (according to the connection method ⅄/△)
C00090	Rated motor voltage (according to the connection method ⅄/△)
C00091	Motor cos φ

- If the motor nameplate data are entered in the »Engineer« motor catalogue instead of selecting a motor, it is recommendable to use the extended motor parameter identification (C02867/1 = "2: extended identification").
- Especially for third-party motors, it is also recommendable to carry out the slip calculation with the motor equivalent circuit diagram data (C02879/1). If the slip calculation is derived from the motor nameplate data, this can negatively affect the stability and accuracy of the drive behaviour because the rated speed specified on the motor nameplate often is rather inaccurate.
- The available motor cable must be specified in terms of length and cross-section:

Parameters	Information
C00915	Motor cable length
C00916	Motor cable cross-section

  The motor cable resistance resulting from these settings is displayed in C00917.

- For the measurement of the required variables, the motor is energised via the inverter terminals U, V and W during the motor parameter identification. The corresponding current controller is preset in the Lenze setting so that a optimal controller behaviour is achieved with an asynchronous motor power-adapted to the inverter.
  Thanks to optimisation, the current controller can be set via the following parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C00075	Vp current controller	7.00	V/A
C00076	Ti current controller	10.61	ms

- Switching frequency for the motor parameter identification:
  - Up to and including version 13.xx.xx, the motor parameter identification is executed with a switching frequency of 4 kHz.
  - From version 14.00.00, the motor parameter identification can also be executed with a switching frequency of 8 kHz instead of 4 kHz. For this purpose, the option "Motor ident.: Switching frequency 8 kHz" (Bit 4 = "1") has to be set in C02864/1.
    Example of how to use this option: Between the output of the inverter and the motor, a sinusoidal filter is connected which may only be operated with a minimum switching frequency of 8 kHz. (See also the section "Preventing a decrease of the switching frequency".) (📖 273)

  Generally, a switching frequency of 4 kHz is recommended for the motor parameter identification as it serves to obtain the most accurate results.

Premature abort of the motor parameter identification

Stop!

If motor parameter identification is aborted, unstable drive behaviour may be the result!

The motor parameter identification can be aborted in the following cases:

- If a special motor (e.g. mid-frequency motor) or a servo motor is used.
- If there is a large deviation between inverter and motor power.

In case of a simple motor parameter identification, we recommend the following:

- to reduce the P component Vp of the current controller (C00075) e.g. by halving.
- to increase the time constant Ti of the current controller (C00076) e.g. by doubling.

In case of the extended motor parameter identification, the current controller parameters are determined automatically. If the identification is aborted all the same, the current controller parameters set in C00075 and C00076 can be used by parameterising C02866 to "1".

Another cause for the abort of the motor parameter identification could be the implausibility of the entered nameplate data, e.g. the entry P = 0 kW for the motor power.

Execute automatic motor parameter identification

Danger!

During motor parameter identification, the motor is energised via the outputs U, V and W of the inverter!

- Observe the corresponding safety instructions!
- With an idling motor, a small angular offset may occur at the motor shaft.
- During the motor parameter identification, rotations occur in case of a synchronous motor.

How to carry out automatic motor parameter identification:

1. Activate motor parameter identification via the C00002/23 = "1: On / start" device command.
2. Enable inverter.
   - The inverter changes to the "Ident" device status.
   - Motor parameter identification starts.
   - The progress of the identification run can be seen in C00002/23.
   - The identification is completed if the "0: Off / ready" message is displayed in C00002/23.
   - After successful identification, it changes back to the "SwitchedOn" device status.
3. Inhibit inverter again.

5.1.3 Application notes for asynchronous motors with high slip speed

Note!

If a rated motor speed is set for an asynchronous motor (ASM) in C00087 which corresponds to a very high slip speed, the number of motor pole pairs is identified incorrectly.

Moreover, a synchronous motor is (PSM) is detected instead of an asynchronous motor (ASM). In this case, the "Id7" error message can occur since motor and motor control type do not match.

Examples, possible impacts and remedies are described in the following sections.

Examples:

Number of pole pairs	Mechanical synchronous speed	Rated motor frequency (C00089)	Incorrect number of pole pairs if the rated motor speed (C00087) is lower:
1	3000 rpm	50 Hz	1986 rpm
2	1500 rpm	50 Hz	1195 rpm
3	1000 rpm	50 Hz	854 rpm
4	750 rpm	50 Hz	665 rpm

Possible impacts depending on the motor control type:

Motor control type	Possible impacts on the drive
Servo control (SC) - PSM	No impact if the motor data have been entered correctly.
Servo control (SC) - ASM	The motor can accelerate up to maximum speed of the motor.
V/f control (VFCplus + encoder)	
Sensorless vector control (SLVC)	The output speed is considerably too high. Example: p=2, f_n=50 Hz, n_n=1195 min^{-1} → Speed is 50 % too high
V/f characteristic control (VFCplus)	

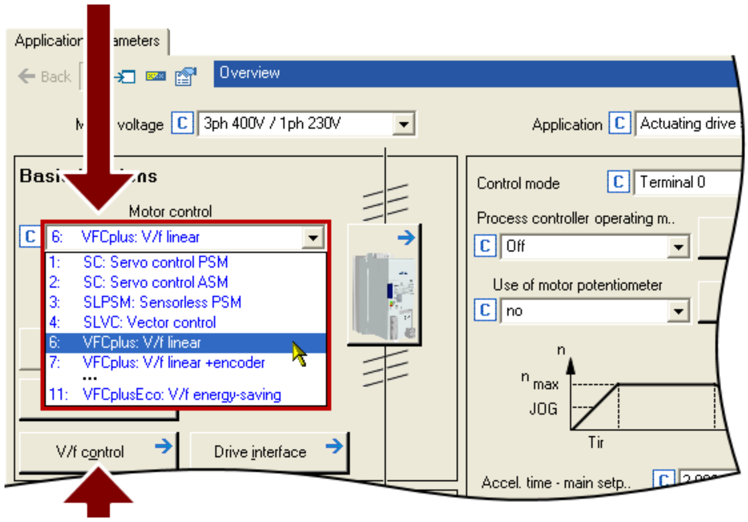
Remedies:

- For version 12.xx.xx and version 13.xx.xx the following applies:
  Limit the rated motor speed: C00087 = C00089 * 60 / (number of pole pairs + 0.5)
- From version 14.00.00:
  Set the motor type ("ASM" or "PSM") manually in C01001/1. The rated motor speed does not need to be limited since the number of pole pairs can be detected safely when the motor type has been set manually.

5.2 Selecting the control mode

The 8400 TopLine inverter supports various modes for motor control (open loop or closed loop).

- V/f characteristic control (VFCplus) with linear characteristic for asynchronous motors is preset.
- The control mode can be selected in the »Engineer« on the **Application parameter** tab via the **Motor control** (C00006) list field:



- A click on the **Motor control...** button leads you to the parameterisation dialog of the selected motor control. (The button is labelled according to the selected motor control.)

Tip!

In order to make the selection of the motor control easier, we provide a selection help with recommendations and alternatives for standard applications in the subchapter entitled "Selection help". (📖 163)

The following section briefly describe the control modes. A reference to more details can be found at the end of each section.

V/f characteristic control (VFCplus)

The V/f characteristic control (VFCplus) is a motor control mode for standard frequency inverter applications based on a simple and robust control process which is suitable for the operation of asynchronous motors with linear or square-law load torque characteristic (e.g. fans). Furthermore, this motor control mode is also suitable for group drives and special motors. Due to the low parameterisation effort, commissioning of such applications is fast and easy.

The V_{min} boost (C00016) and slip compensation (C00021) required for optimising the drive behaviour are dimensioned for asynchronous motors with power adaptations to the inverter in the Lenze setting.

▸ V/f characteristic control (VFCplus) (📖 167)

Energy-saving V/f characteristic control (VFCplusEco)

In contrast to the V/f characteristic control mode (VFCplus), this motor control mode uses a cosφ control in partial load operational range to automatically reduce the power loss in the asynchronous motor (energy optimisation).

The motor data required for the cosφ control and the V_{min} boost (C00016) and slip compensation (C00021) required for optimising the drive behaviour are dimensioned for asynchronous motors with power adaptations to the inverter in the Lenze setting.

The required motor data (motor rotor resistance, motor stator resistance, motor stator leakage inductance and mutual motor inductance) only affect the extent of energy optimisation but not the stability.

In case of applications with dynamically very high sudden load variations from the unloaded operation, this motor control mode should not be used since a motor stalling cannot be excluded.

Energy optimisation for dynamic applications is not possible with this motor control mode.

▸ V/f characteristic control - energy-saving (VFCplusEco) (📖 186)

V/f control (VFCplus + encoder)

The V/f control can be selected for operating asynchronous motors with speed feedback. With this motor control, a slip regulator can be additionally parameterised which adjusts the actual speed value dynamically to the speed setpoint.

▸ V/f control (VFCplus + encoder) (📖 196)

Sensorless vector control (SLVC)

Sensorless (field-oriented) vector control for asynchronous motors is based on a decoupled, separate control for the torque-producing and the field-producing current component. In addition, the actual speed is reconstructed by means of a motor model so that a speed sensor is not required.

In comparison to the V/f characteristic control without feedback, the following can be achieved by means of sensorless vector control SLVC:

- A higher maximum torque throughout the entire speed range
- A higher speed accuracy
- A higher concentricity factor
- A higher level of efficiency
- The implementation of torque-actuated operation with speed limitation
- The limitation of the maximum torque in motor and generator mode for speed-actuated operation

Tip!

If a high torque without feedback is to be provided at small speeds, we recommend the "Sensorless vector control" motor control mode.

▸ Sensorless vector control (SLVC) (📖 204)

Sensorless control for synchronous motors (SLPSM)

This sensorless control enables an encoderless control of synchronous motors. The process is based on field-oriented control within a higher speed range (e.g. > 10 % of the rated motor speed). The actual speed value and rotor position are reconstructed via a motor model.

Standard applications for this control type are pumps and fans, horizontal materials handling and simple positioning technology.

▸ Sensorless control for synchronous motors (SLPSM) (📖 221)

Servo control (SC)

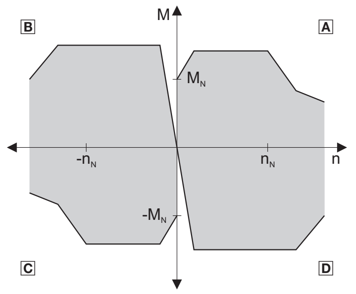
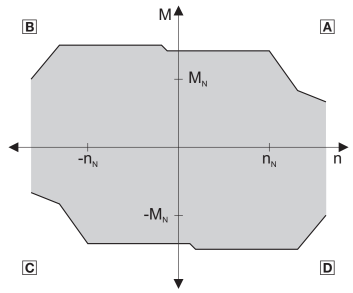
Field-oriented servo control (SC) is based on a decoupled, separate control of the torque-producing and the field-producing current component. The motor control is based on a field-oriented, cascaded controller structure with feedback function and enables dynamic and stable operation in all of the four quadrants.

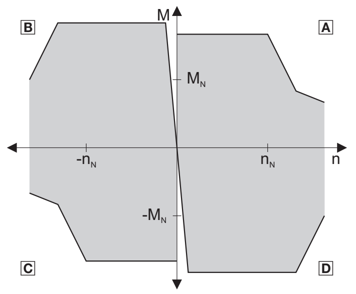
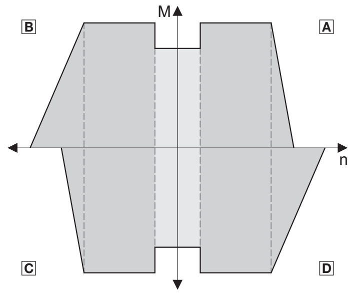
The servo control can be used for synchronous motor (PSM) and asynchronous motors (ASM) and basically offers the same advantages as the sensorless vector control (SLVC).

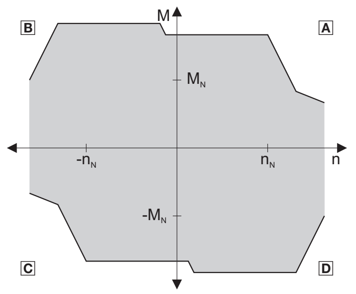
▸ Servo control (SC) (📖 239)

Speed feedback

As shown in the following graphics, the drive systems with feedback have, independently of the motor control, more advantages than systems without feedback.

V/f characteristic control (VFCplus)	V/f control (VFCplus + encoder)
 9300vec092	 9300vec093
A Operation in motor mode (CW rotation), B Operation in generator mode (CCW rotation), C Operation in motor mode (CCW rotation), D Operation in generator mode (CW rotation)	

Sensorless vector control (SLVC)	Sensorless control for synchronous motors (SLPSM)
 9300vec095	
A Operation in motor mode (CW rotation), B Operation in generator mode (CCW rotation), C Operation in motor mode (CCW rotation), D Operation in generator mode (CW rotation)	

Servo control (SC)	
 9300vec094	
A Operation in motor mode (CW rotation), B Operation in generator mode (CCW rotation), C Operation in motor mode (CCW rotation), D Operation in generator mode (CW rotation)	

5.2.1 Selection help

To ease the selection of the motor control mode, the two following tables contain recommendations and alternatives to standard applications.

Application	recommended	Alternatively
Single drives		
With constant load	VFCplus: V/f linear	SLVC or SLPSM
With extremely alternating loads	VFCplus: V/f linear	SLVC
With high starting duty	SLVC	VFCplus: V/f linear
Torque limitation	SLVC	SLPSM
With torque limitation (power control)	VFCplus: V/f linear	SLPSM
Three-phase reluctance motor	VFCplus: V/f linear	-
Three-phase sliding rotor motor	VFCplus: V/f linear	-
Three-phase AC motors with permanently assigned frequency/voltage characteristic	VFCplus: V/f linear	-
Pump and fan drives with quadratic load characteristic	VFCplusEco	SLVC or SLPSM
Simple hoists	VFCplus: V/f linear	-
Group drives *(several motors connected to inverter)*		
Identical motors and loads	VFCplus: V/f linear	-
Different motors and/or alternating loads	VFCplus: V/f linear	-

[5-1] Standard applications without speed feedback

Application	recommended	Alternatively
Single drives		
With constant load	SC	VFCplus / SLVC
With extremely alternating loads	SC	VFCplus / SLVC
With high starting duty	SC	VFCplus / SLVC
With speed control (speed feedback)	SC	VFCplus
With high dynamic performance e.g. for positioning and infeed drives	SC	-
Torque limitation	SC	SLVC
With torque limitation (power control)	-	-
Winder with dancer position control	SC	VFCplus
Unwinder with dancer position control	SC	VFCplus
Three-phase reluctance motor	-	-
Three-phase sliding rotor motor	-	-
Three-phase AC motors with permanently assigned frequency/voltage characteristic	-	-
Pump and fan drives with quadratic load characteristic	-	-
Simple hoists	VFCplus: V/f linear	-
Group drives *(several motors connected to inverter)*		
Identical motors and loads	VFCplus	-
Different motors and/or alternating loads	-	VFCplus

[5-2] Standard applications with speed feedback

5.3 Defining current and speed limits

Limitation of the speed setpoint

Parameterising the reference speed in C00011 means that the drive must rotate at the set speed if a speed setpoint of 100% is specified.

All speed setpoint selections are provided in % and always refer to the reference speed set in C00011.

Tip!

For reasons of achievable resolution and the accuracy involved, the reference speed should be geared to the speed range required for the respective application.

Lenze recommendation: Reference speed (C00011) = 1500 ... 3000 rpm

Irrespective of the selected motor control, there are more limitation options:

Parameters	Info	Lenze setting	
		Value	Unit
C00909/1	Max. positive speed	120	%
C00909/2	Max. negative speed	120	%
C00910/1	Max. positive output frequency	599	Hz
C00910/2	Max. negative output frequency	599	Hz

Note!

In the torque-controlled operation (*bTorquemodeOn* = TRUE), the limitation of the speed setpoint set in C00909/x has no impact!

For the torque-controlled operation, a permissible speed range can be defined via the speed limitation (inputs *nSpeedHighLimit_a* and *nSpeedLowLimit_a* at the SB LS_MotorInterface).

For a correct function of the speed limitation in both direction, the following applies: The speed limitation operates with a internal hysteresis of 50 min^{-1}. The upper or lower speed limit resulting from the settings has to be higher than this hysteresis!

Current limitation in motor and generator mode

In the various motor control modes, the inverter is provided with functions which determine the dynamic behaviour under load and counteract exceedance of the maximum current in motor or generator mode.

Parameters	Info	Lenze setting	
		Value	Unit
C00022	Imax in motor mode	47.00	A
C00023	Imax in generator mode • 100 % ≡ Imax in motor mode (C00022)	100	%

The current limits must be selected depending on

- the permissible maximum current of the motor → recommendation: $I(Mot)_N$ < 1.5 ... 2.0
- the permissible maximum current of the inverter
- the torque in motor/generator mode required for the application

Note!

Highly dynamic applications
(that have e.g. too short acceleration/deceleration times or excessively changing loads)

The overcurrent disconnection may respond (fault message oC1 or oC11) if the setting of the maximum current in motor mode in C00022 approximately corresponds to the maximum permissible value of the respective inverter.

Remedies:

- Increase of the acceleration and deceleration ramp times
- Reduction of the maximum current in motor mode (C00022)
- Reduction of the maximum current in generator mode (C00023)
- Adaptation of the indirect peak current limitation (procedure depends on the selected motor control mode, see below)
- Reduction of the reset time of the current limiting controller (C00074/1)

Influencing the torque in motor/generator mode

The torque in motor and generator mode can be limited via the *nTorqueMotLim* and *nTorqueGenLim* process signal inputs.

- If V/f characteristic control (VFCplus) is selected, limitation is indirectly performed via a so-called I_{max} controller.
- If sensorless vector control (SLVC), sensorless control for synchronous motors (SLPSM) or servo control (SC) is selected, limitation has a direct effect on the torque-producing current component.

If keypad control is selected, the *nTorqueMotLim* and *nTorqueGenLim* process signals can be parameterised via C00728/1...2.

From version 18.00.00 onwards

The positive and negative torque can be limited via the two process signal inputs *nTorqueMotLimit_a* and *nTorqueGenLimit_a*.

- C02864: Bit 15 = 1: *nTorqueMotLimt_a* acts as *nTorqueHighLimit_a* (positive torque limitation) and *nTorqueGenLimit_a* acts as n*TorqueLowLimit_a* (negative torque limitation).

How to adapt the peak current limitation:

V/f characteristic control (VFCplus):

- Reduce the slip compensation with C00021.

V/f control (VFCplus + encoder):

- Reduce the slip limitation to twice the rated motor slip with C00971.
- Reduce the V_{min} boost in C00016.

Sensorless vector control (SLVC):

- Reduce the slip compensation with C00021.
- Reduce the limitation of the torque in motor mode via *nTorqueMotLimit_a* (C00728/1) and the limitation of the torque in generator mode via *nTorqueGenLimit_a* (C00728/2).

Servo control (SC):

- Reduce the jerk limitation with C00274.
- Reduce the limitation of the torque in motor mode via *nTorqueMotLimit_a* (C00728/1) and the limitation of the torque in generator mode via *nTorqueGenLimit_a* (C00728/2).

5.4 V/f characteristic control (VFCplus)

In case of the V/f characteristic control (VFCplus), the motor voltage of the inverter is determined by means of a linear or quadratic characteristic depending on the field frequency or motor speed to be generated. The voltage follows a preselected characteristic.

Stop!

- The V/f characteristic control is only suitable for asynchronous motors.
- The following must be observed when operating drives with quadratic V/f characteristic:
  - Please always check whether the corresponding drive is suitable for operation with a quadratic V/f characteristic!
  - If you pump or fan drive is not suitable for operation with a square-law V/f characteristic, we recommend using the energy-saving V/f characteristic control (VFCplusEco). Alternatively, you can use the V/f characteristic control with linear V/f characteristic or the sensorless vector control (SLVC) or servo control (SC).
- For adjustment, observe the thermal performance of the connected asynchronous motor at low output frequencies.
  - Usually, standard asynchronous motors with insulation class B can be operated for a short time with their rated current in the frequency range 0 Hz ... 25 Hz.
  - Contact the motor manufacturer to get the exact setting values for the max. permissible motor current of self-ventilated motors in the lower speed range.
  - If you select square-law V/f characteristics, we recommend setting a lower V_{min} or using the energy-saving V/f characteristic control (VFCplusEco).
- The nameplate data of the motor (at least rated speed and rated frequency) must be entered if, instead of a standard motor, an asynchronous motor is used with the following values:
  - rated frequency $\neq$ 50 Hz (star) or
  - rated frequency $\neq$ 87 Hz (delta) or
  - number of pole pairs $\neq$ 2

Note!

When the auto DCB threshold (C00019) is set > 0 rpm, there is no torque at the motor shaft in the lower speed range!

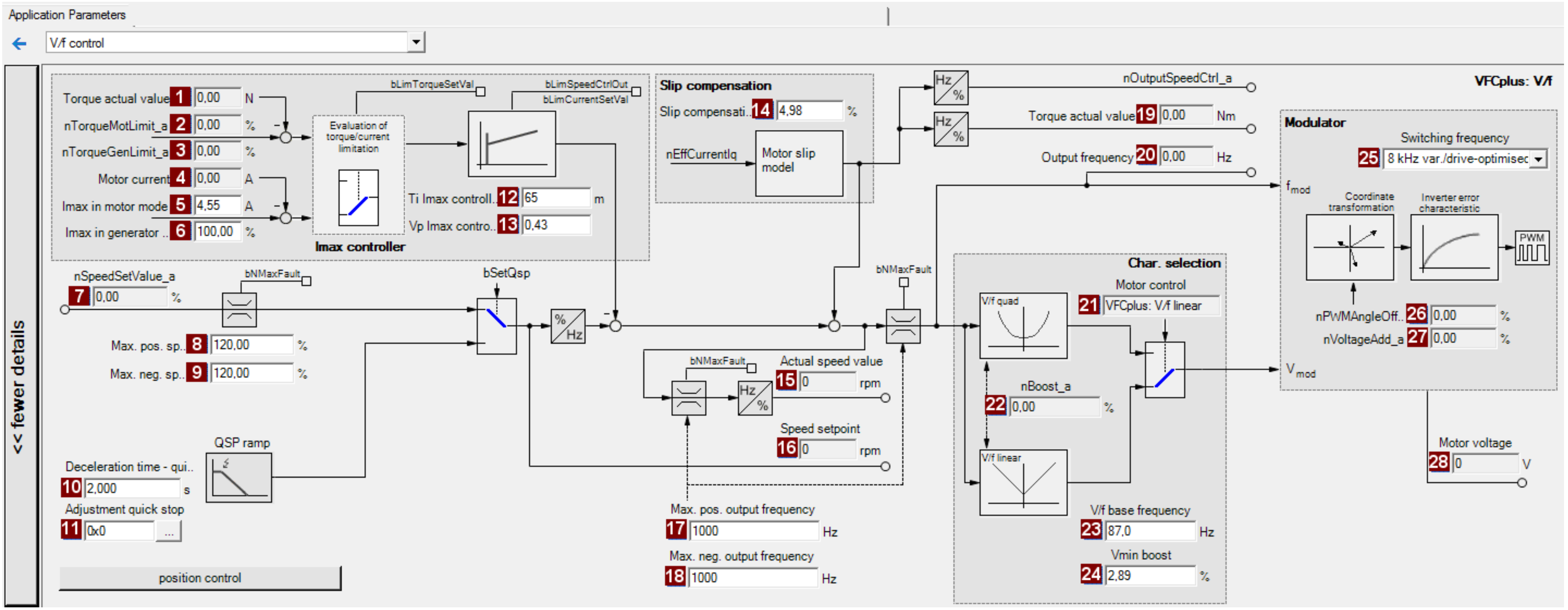
▶ Automatic DC-injection braking (Auto-DCB) (📖 284)

5.4.1 Parameterisation dialog/signal flow

Proceed as follows to open the dialog for parameterising the motor control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Select the motor control from the *Overview* dialog level in the **Motor control** list field:
   - "6: VFCplus: V/f linear" for linear characteristic or
   - "8: VFCplus: V/f quadr" for square-law characteristic

   More available V/f characteristic control modes:
   - "10: VFCplus: V/f definable".
     With this motor control, the V/f characteristic can be freely defined. ▸ Defining a user-defined V/f characteristic (📖 182)
   - "11: VFCplusEco: V/f energy-saving".
     With this motor control, the motor is always operated in an optimal efficiency range via a cosφ control and the resulting voltage reduction (reduced copper losses in the asynchronous motor). ▸ V/f characteristic control - energy-saving (VFCplusEco) (📖 186)
4. Click the **Motor control V/f** button to change to the *Overview* → *Motor control V/f* dialog box.
   - This dialog level only shows a simplified signal flow with the most important parameters.
   - When you click the **>>More details** button in the left-most position, a signal flow with more details/parameters is displayed.



Parameters		Info
1	C00056/2	Actual torque
2	C00830/29	Limitation of torque in motor mode
3	C00830/28	Limitation of torque in generator mode
4	C00054	Motor current
5	C00022	Imax in motor mode
6	C00023	Imax in generator mode
7	C00830/22	Speed setpoint
8	C00909/1	Max. pos. speed
9	C00909/2	Max. neg. speed
10	C00105	Decel. time - quick stop
11	C00104/1	Quick stop setting
12	C00074/1	Ti Imax controller
13	C00073/1	Vp Imax controller
14	C00021	Slip compensation
15	C00051	Actual speed value
16	C00050	Speed setpoint
17	C00910/1	Max. pos. output frequency
18	C00910/2	Max. neg. output frequency
19	C00056/2	Actual torque
20	C00058	Output frequency
21	C00006	Motor control
22	C00830/26	MCTRL: nBoost_a
23	C00015	V/f base frequency
24	C00016	Vmin boost
25	C00018	Switching frequency
26	C00830/32	MCTRL: nPWMAngleOffset_a
27	C00830/31	MCTRL: nVoltageAdd_a
28	C00052	Motor voltage

5.4.2 Basic settings

The "Initial commissioning steps" listed in the table below are sufficient for a simple characteristic control.

- Detailed information on the individual steps can be found in the following subchapters.

Initial commissioning steps	
1	Define V/f characteristic shape. (171)
2.	Defining current limits (Imax controller). (172)

Tip!

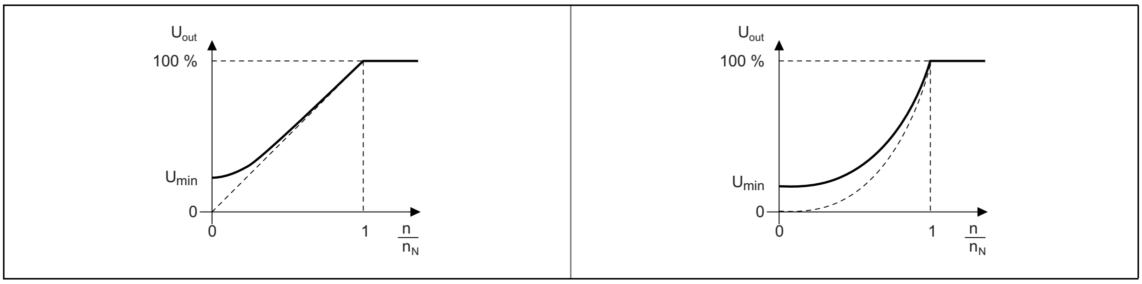
Information on the optimisation of the control mode and the adaptation to the real application is provided in chapter "Optimising the control mode". (173)

Parameterisable additional functions are described in the chapter entitled "Parameterisable additional functions". (272)

5.4.2.1 Define V/f characteristic shape

In principle, four different characteristic shapes can be stipulated:

1. **Linear V/f characteristic**:
   For drives for a constant, speed-independent load torque.
2. **Quadratic V/f characteristic**:
   For drives with a load torque curve which is quadratic or in relation to speed. Quadratic V/f characteristics are preferred in the case of centrifugal pumps and fan drives.
3. **Freely definable V/f characteristic**:
   For drives that require adaptation of the magnetising current by means of the output speed. The freely definable V/f characteristic can be used e.g. for operation in conjunction with special machines such as reluctance motors in order to suppress oscillations at the machine or to optimise energy consumption.



[5-3] Principle of a linear V/f characteristic (on the left) and a quadratic V/f characteristic (on the right)

4. **Linear V/f characteristic with voltage reduction**:
   For drives which often work in partial load operation, the energy-saving V/f characteristic control (VFCplusEco) offers the opportunity to reduce the voltage at low load in order to save energy. At higher loads, the voltage reduction is cancelled and a linear characteristic is caused.

The V/f characteristic shape is defined by selecting the corresponding motor control mode in C00006:

V/f characteristic shape	Motor control to be selected (C00006)
Linear V/f characteristic	6: VFCplus: V/f linear
Square-law V/f characteristic	8: VFCplus: V/f quadr
User-definable V/f characteristic	10: VFCplus: V/f definable
Linear V/f characteristic with voltage reduction	11: VFCplusEco: V/f energy-saving

Tip!

- You can find detailed information on freely definable V/f characteristics in the subchapter entitled "Defining a user-defined V/f characteristic". (📖 182)
- You can find detailed information on the linear V/f characteristic with voltage reduction in the chapter entitled "V/f characteristic control - energy-saving (VFCplusEco)". (📖 186)

5.4.2.2 Defining current limits (Imax controller)

The V/f characteristic control (VFCplus) and the V/f control (VFCplus + encoder) operating modes are provided with a current limitation control which is decisive for the dynamic behaviour under load and counteracts exceedance of the maximum current in motor or generator mode. This current limitation control is called I_{max} control.

- The efficiency (motor current) measured by the I_{max} control is compared with the current limit value for motor load set in C00022 and the current limit value for generator load set in C00023.
- If the current limit values are exceeded, the inverter changes its dynamic behaviour.

Motor overload during acceleration

The inverter prolongs the acceleration ramp to keep the current on or below the current limit.

Generator overload during deceleration

The inverter prolongs the acceleration ramp to keep the current on or below the current limit.

Increasing load with constant speed

- If the motor current limit value is reached:
  - The inverter reduces the effective speed setpoint until a stable working point is set or an effective speed setpoint of 0 rpm is reached.
  - If the load is reduced, the inverter increases the effective speed setpoint until the setpoint speed is reached or the load reaches the current limit value again.
- When the generator current limit value is reached:
  - The inverter increases the effective speed setpoint until a stable working point is set or the maximally permissible speed (C00909) or output frequency is reached (C000910).
  - If the load is reduced, the inverter reduces the effective speed setpoint until the setpoint speed is reached or the load reaches the current limit value again.
- If a sudden load is built up at the motor shaft (e.g. drive is blocked), the overcurrent disconnection may respond (fault message oC1 or oC11).

5.4.3 Optimising the control mode

The V/f characteristic control (VFCplus) is generally ready for operation. It can be adapted subsequently by adapting the characteristic and/or the drive behaviour.

Note!

Following successful motor parameter identification, the V/f base frequency (C00015) and the V_{min} boost (C00016) as well as the slip constant for slip compensation (C00021) are calculated automatically.

From version 12.00.00:

- Following successful motor parameter identification, the gain of the Imax controller (C00073/1) is calculated automatically.
  - If these parameters are not to be calculated, bit 4 of C02865/1 must be set to "1".
- Following successful motor parameter identification, other controller parameters (C00011, C00022, C00966) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "6".

Adapting characteristic

For the linear and quadratic characteristic, it is also possible to match its curve to different load profiles or motors by adapting the V/f base frequency (C00015) and the V_{min} boost (C00016).

▸ Adapting the V/f base frequency (📖 174)

▸ Adapting the Vmin boost (📖 176)

Freely defining the characteristic

The V/f characteristic can also be defined freely if the linear and quadratic characteristics are not suitable.

▸ Defining a user-defined V/f characteristic (📖 182)

Adapting drive behaviour

- Limitation of the maximum current by a current limitation controller (e.g. to prevent the motor from stalling or to limit to the maximally permissible motor current). ▸ Optimising the Imax controller (📖 178)
- Adaptation of the field frequency by a load-dependent slip compensation (improved speed accuracy for systems without feedback).
- Adaptation of the controller parameters of the slip regulator if V/f control (VFCplus + encoder) is selected. ▸ Parameterising the slip regulator (📖 200)

5.4.3.1 Adapting the V/f base frequency

The V/f base frequency (C00015) determines the slope of the V/f characteristic and has considerable influence on the current, torque, and power performance of the motor.

- The setting in C00015 applies to all permitted mains voltages.
- Mains fluctuations or fluctuations of the DC-bus voltage (operation in generator mode) do not need to be considered when the V/f base frequency is set. They are automatically compensated for by the internal mains voltage compensation of the device.
- Depending on the setting in C00015, it may be required to adapt the reference speed (C00011) to traverse the entire speed range of the motor.
- The V/f base frequency is automatically calculated from the stored motor nameplate data by the motor parameter identification:

$$C00015\,[Hz] = \frac{U_{FI}\,[V]}{U_{Ratedmot}\,[V]} \cdot f_{Rated}\,[Hz]$$

V_{FI}: Mains voltage 400 V or 230 V
$V_{ratedmot}$: Rated motor voltage depending on the connection method
f_{rated}: Rated motor frequency

[5-4] Calculation of the V/f base frequency

Typical values of the V/f base frequency

Inverter with 400-V mains connection			
Motor voltage [V]	**Motor frequency [Hz]**	**Motor connection**	**V/f base frequency (C00015)**
230 / 400	50	⅄	50 Hz
220 / 380	50	⅄	52.6 Hz
280 / 480	60	⅄	50 Hz
400 / 690 400	50 50	△	50 Hz
230 / 400 280 / 480 400	50 60 87	△	87 Hz
220 / 380	50	△	90.9 Hz

Inverter with 230-V mains connection			
Motor voltage [V]	**Motor frequency [Hz]**	**Motor connection**	**V/f base frequency (C00015)**
230	50	△	50 Hz
220 / 380	50	△	52.3 Hz

Note!

87-Hz operation

4-pole asynchronous motors which are designed for a rated frequency of f = 50 Hz in star connection can be operated in delta connection when being constantly excited up to f = 87 Hz.

- Advantages:
  - Higher speed-setting range
  - 73% higher power output in case of standard motors
- Motor current and motor power increase by the factor $\sqrt{3}$.
- The field weakening range starts above 87 Hz.
- Generally, this process can also be used with motors which have different numbers of pole pairs. In case of 2-pole asynchronous motors, the mechanical limit speed must be maintained.

5.4.3.2 Adapting the Vmin boost

The V_{min} boost (C00016) of the motor voltage serves to select a load independent magnetising current which is required for asynchronous motors. The torque behaviour of the motor can be optimised by adapting the setting in C00016.

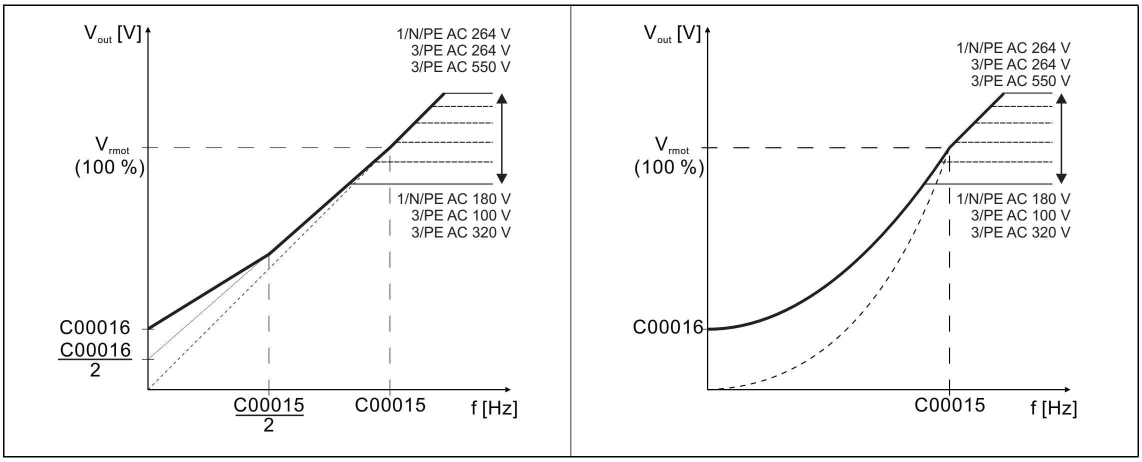
The *nBoost_a* process signal at the SB LS_MotorInterface serves to carry out a V_{min} boost as well:

Designator DIS code | data type	**Information/possible settings**
nBoost_a C00830/26 | INT	Process signal for the V_{min} boost • This signal is added to C00016 and has thus an increasing or decreasing effect. • This signal serves to implement a load-dependent V_{min} boost to improve the torque behaviour in different load states, as for instance in case of operation in generator or motor mode. • Scaling: 16384 ≡ 100 % rated device voltage V_{FU} (400 V or 230 V)
	STOP **Stop!** Values selected too high may cause the motor to heat up due to the resulting current!

Note!

The V_{min} boost has an effect on output frequencies below the V/f base frequency (C00015).

The general linear and quadratic V/f characteristics are shown in the illustrations below. The illustrations show the impacts of the parameters used to adapt the characteristic shape.



[5-5] Representation of the linear V/f characteristic (on the left) and quadratic V/f characteristic (on the right)

How to set the V_{min}boost:

1. Operate motor in idle state at approx. 6 % of the rated motor speed.
2. Increase V_{min} boost (C00016) until the following motor current is reached:

   Motor in short-time operation up to 0.5 n_{rated}
   - for self-ventilated motors: $I_{motor} \approx I_{rated\ motor}$
   - for forced ventilated motors: $I_{motor} \approx I_{rated\ motor}$

   Motor in continuous operation up to 0.5 n_{rated}
   - for self-ventilated motors: $I_{motor} \approx 0.8\ I_{rated\ motor}$
   - for forced ventilated motors: $I_{motor} \approx I_{rated\ motor}$

Note!

V_{min} boost is automatically calculated by the motor parameter identification using the data specified on the motor nameplate so that a no-load current of approx. 0.8 $I_{rated\ motor}$ results at the slip frequency of the machine.

V/f control (VFCplus + encoder)

- Occurring vibrations can be decreased by reducing the V_{min} boost C00016.
- In case of operation in motor mode, we recommend a considerably lower V_{min} boost: The V_{min} boost should be dimensioned in such a way that in case of slip frequency in idle state, approx. 50 % of the rated motor current flows.
- In case of operation in generator mode, a light increase of the speed may be caused in certain load ranges in the lower speed range. This speed increase can be reduced by a higher V_{min} boost.
- The *nBoost_a* process signal is added to C00016 and can be used to implement different V_{min} boosts for operation in motor and generator mode. As an alternative, the *nVoltageAdd_a* process signal can be used as well.

5.4.3.3 Optimising the Imax controller

Using the Lenze setting of the current limitation controller, the drive is stable:

Parameters	Info	Lenze setting	
		Value	Unit
C00073/1	VFC: Vp Imax controller	0.25	
C00074/1	VFC: Ti Imax controller	65	ms

Most applications do not require optimisation.

The setting of the current limitation controller must be adapted if

- power control including great moments of inertia is performed.
  - Recommendation: Increase of the reset time Ti (C00074/1) of the I_{max} controller.
- vibrations occur in the V/f control (VFCplus + encoder) mode during the intervention of the current limitation controller.
  - Recommendation: Increase of the reset time Ti (C00074/1) of the I_{max} controller.
- overcurrent errors (e.g. OC3) occur due to load impulses or too high acceleration ramps.
  - Recommendation: Reduction of the gain Vp (C00073/1) and reset time Ti (C00074/1) of the I_{max} controller.

5.4.3.4 Optimising the stalling behaviour

Motor stalling due to a torque overload in the field weakening range is prevented in all characteristic-based motor control types (VFCplus) by means of an inverter-internal stalling current monitoring. In the field weakening range, hence at frequencies above the base frequency, it reduces the maximum current to prevent the motor from stalling. The reduction depends on the current field frequency, the base frequency, the DC-bus voltage and the maximum current (C00022). Generally it applies that a higher field frequency causes a stronger limitation of the maximum current.

The behaviour in the field weakening range can be adapted via the override point of field weakening (C00080). This parameter serves to shift the frequency-dependent maximum current characteristic:

- C00080 > 0 Hz:
  - The maximum current characteristic is shifted by the entered frequency to higher field frequencies.
  - The maximally permissible current and the maximum torque increase in the field weakening range.
  - The risk of motor stalling increases.
- C00080 < 0 Hz:
  - The maximum current characteristic is shifted by the entered frequency to lower field frequencies.
  - The maximally permissible current and the maximum torque are reduced in the field weakening range.
  - The risk of motor stalling is reduced.

Note!

We recommend to keep the Lenze setting (0 Hz).

5.4.3.5 Torque limitation

The "Optimising the Imax controller" chapter describes how the drive can be protected from overload. During commissioning, these settings are carried out once and remain unchanged afterwards. However, it is often necessary to limit the torque to a lower value for plant or process reasons.

- To avoid overload in the drive train, the torque in motor mode can be limited via the *nTorqueMotLimit_a* process input signal, and the torque in generator mode can be limited via the *nTorqueGenLimit_a* process input signal:

Designator DIS code | data type	**Information/possible settings**
nTorqueMotLimit_a C00830/29 | INT	Torque limitation in motor mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: 0 ... +199.99 % • If keypad control is performed: Parameterisable via C00728/1. From version 18.00.00 onwards: C02864: Bit 15 = 1: positive torque limitation (nTorqueHighLimit_a)
nTorqueGenLimit_a C00830/28 | INT	Torque limitation in generator mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: -199.99 ... 0 % • If keypad control is performed: Parameterisable via C00728/2. From version 18.00.00 onwards: C02864: Bit 15 = 1: negative torque limitation (nTorqueLowLimit_a)

Note!

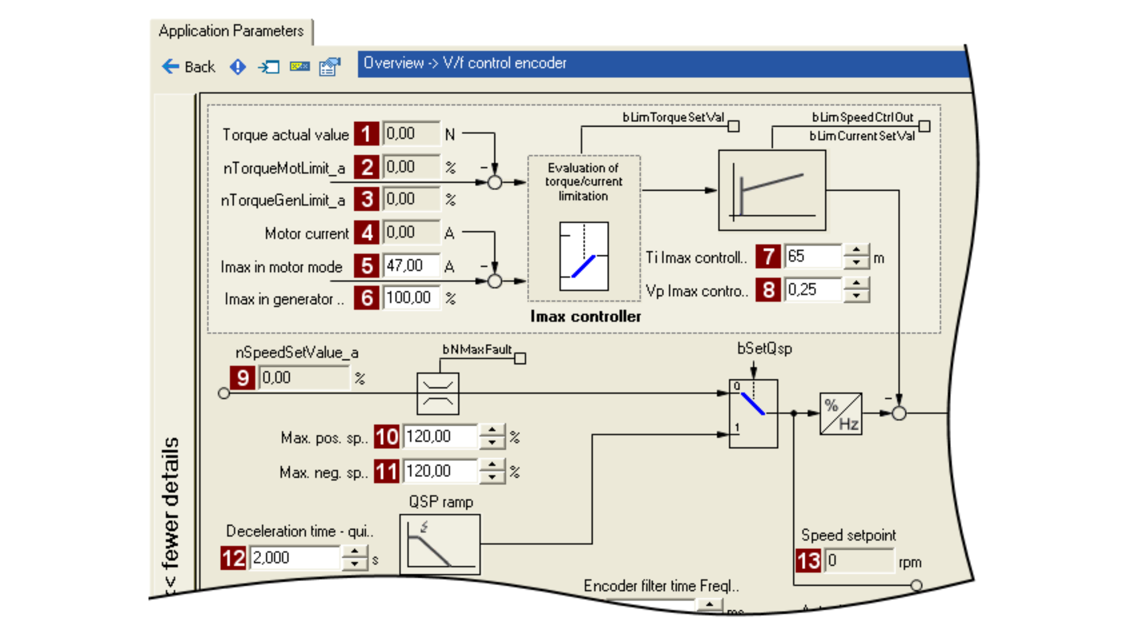
- The actual torque (C00056/2) is directly calculated from the current slip speed of the machine. This requires correct entry of the motor data. (📖 144) ▶ Motor selection/Motor data
- To avoid instabilities during operation with active slip compensation, the torque limit values are internally processed as absolute values.
- If slip compensation is deactivated (C00021 = 0), indirect torque limitation (differential signal between apparent motor current and *nTorqueMotLimit_a* or *nTorqueGenLimit_a*) occurs. Above the no-load current of the motor, the accuracy of the indirect torque limitation is limited.

V/f characteristic control (VFC)

The accuracy of the torque limitation is limited because the actual torque (C00056/2) is only calculated from the slip speed measured indirectly via the motor current.

V/f control (VFC + encoder)

The slip speed of the motor is available at the slip controller output. This leads to a high accuracy for the actual torque (C00056/2) and the torque limitation.



[5-6] Extract from the signal flow of the V/f control (VFC + encoder)

Parameters		Info	Parameters		Info
1	C00056/2	Actual torque	9	C00830/22	MCTRL: nSpeedSetValue_a
2	C00830/29	Limitation of torque in motor mode	10	C00909/1	Max. pos. speed
3	C00830/28	Limitation of torque in generator mode	11	C00909/2	Max. neg. speed
4	C00054	Motor current	12	C00105	Decel. time - quick stop
5	C00022	Imax in motor mode	13	C00050	Speed setpoint
6	C00023	Imax in generator mode			
7	C00074/1	Ti Imax controller			
8	C00073/1	Vp Imax controller			

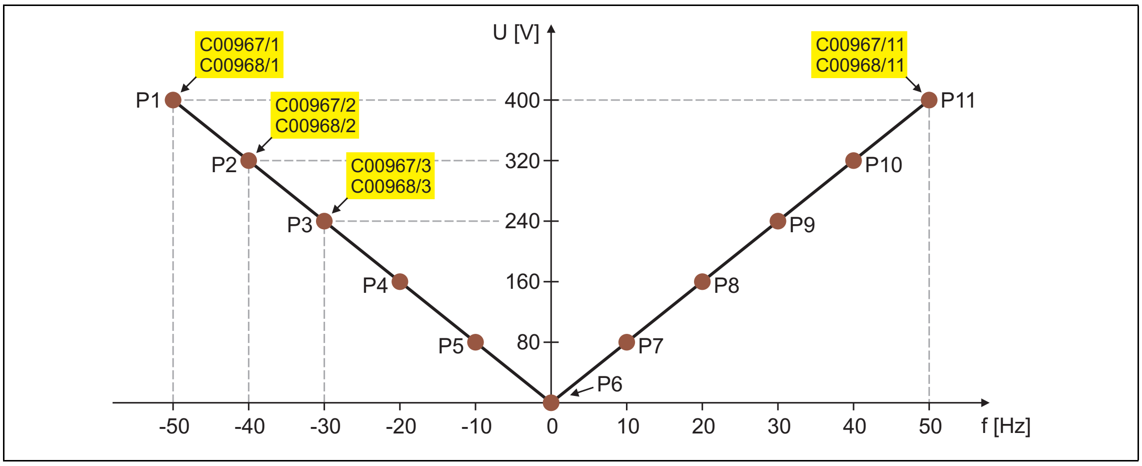
5.4.3.6 Defining a user-defined V/f characteristic

For individual adaptation of the motor magnetisation to the actual application, the motor control "10: VFCplus: V/f definable" with a freely definable characteristic can be selected in C00006 as an alternative if the linear and quadratic characteristics are not suitable.

Note!

The V/f base frequency (C00015) and the V_{min} boost (C00016) no longer exert an influence if this motor control is chosen.

- The 11 grid points (voltage/frequency values) of the characteristic are selected via the 11 subcodes of C00967 and C00968.
  - It is necessary to set all 11 grid points by means of corresponding subcodes.
  - If fewer grid points (voltage/frequency values) are needed, this can be achieved indirectly by ascribing the same voltage and frequency values to consecutive grid points.
    Example: C00967/**3** = C00967/**4** and C00968/**3** = C00968/**4**
  - The grid points can be specified in any sequence. Internally, they are automatically ordered from the minimum to the maximum frequency value.
  - Above the maximum and below the minimum frequency, the previous rise is continued until the maximum output voltage.
- In the Lenze setting, the 11 grid points represent a linear characteristic.
  - 3-phase devices: Output voltage 400 V at f = 50 Hz
  - 1-phase devices: Output voltage 230 V at f = 50 Hz



	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
V	400 V	320 V	240 V	160 V	80 V	0 V	80 V	160 V	240 V	320 V	400 V
f	-50 Hz	-40 Hz	-30 Hz	-20 Hz	-10 Hz	0 Hz	10 Hz	20 Hz	30 Hz	40 Hz	50 Hz

[5-7] Freely definable characteristic (Lenze setting for 3-phase devices)

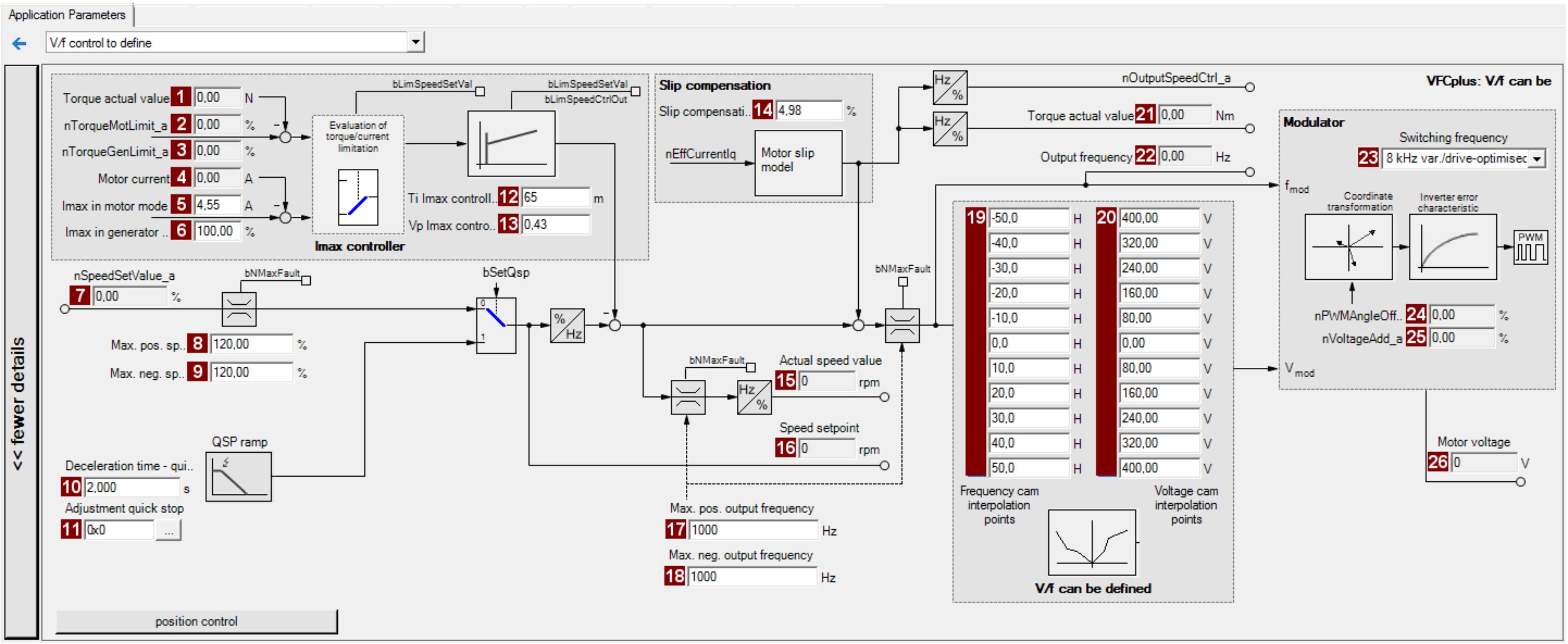
Tip!

Cases of application for this function:

- Operation of reluctance motors or synchronous motors during controlled acceleration (reduction of natural frequencies caused by wrong excitation).
- Adaptation of the voltage requirement for the motor, depending on specific load conditions.

Proceed as follows to open the dialog for parameterising the motor control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Select the motor control "10: VFCplus: V/f definable" from the *Overview* dialog box in the **Motor control** list field:
4. Click the **Motor control V/f definable** button to change to the *Overview* → *Motor control V/f* dialog box.
   - This dialog level only shows a simplified signal flow with the most important parameters.
   - When you click the **>>More details** button in the left-most position, a signal flow with more details/parameters is displayed.



	Parameters	Info		Parameters	Info		Parameters	Info
1	C00056/2	Actual torque	12	C00074/1	Ti Imax controller	19	C00967/x	Frequency interpol. points
2	C00830/29	Limitation of torque in motor mode	13	C00073/1	Vp Imax controller	20	C00968/x	Voltage interpol. points
3	C00830/28	Limitation of torque in generator mode	14	C00021	Slip compensation	21	C00056/2	Actual torque
4	C00054	Motor current	15	C00051	Actual speed value	22	C00058	Output frequency
5	C00022	Imax in motor mode	16	C00050	Speed setpoint	23	C00018	Switching frequency
6	C00023	Imax in generator mode	17	C00910/1	Max. pos. output frequency	24	C00830/32	MCTRL: nPWMAngleOffset_a
7	C00830/22	Speed setpoint	18	C00910/2	Max. neg. output frequency	25	C00830/31	MCTRL: nVoltageAdd_a
8	C00909/1	Max. pos. speed				26	C00052	Motor voltage
9	C00909/2	Max. neg. speed						
10	C00105	Decel. time - quick stop						
11	C00104/1	Quick stop setting						

5.4.4 Remedies for undesired drive behaviour

Drive behaviour	Remedy
Inadequate smooth running at low speeds, especially in the case of operation with a long motor cable	▸ Automatic motor data identification (📖 151)
Problems in case of high starting duty (great mass inertia)	▸ Adapting the Vmin boost (📖 176)
Drive does not follow the speed setpoint.	The current controller intervenes in the set field frequency to limit the controller output current to the maximum current (C0022, C0023). Therefore: • Prolong acceleration/deceleration times: C00012: Accel. time - main setpoint C00013: Decel. time - main setpoint • Consider a sufficient magnetising time of the motor. Depending on the motor power, the magnetising time amounts to 0.1 ... 0.2 s. • Increase the maximally permissible current: C00022: Imax in motor mode C00023: Imax in generator mode)
For operation without speed feedback (C00006 = 6): Insufficient speed constancy at high load (setpoint and motor speed are not proportional anymore)	• Increase slip compensation (C00021). Important: Unstable drive due to overcompensation! • With cyclic load impulses (e. g. centrifugal pump), a smooth motor characteristic is achieved by smaller values in C00021 (possibly negative values). Note: The slip compensation is only active for operation without speed feedback.
"Clamp operation active" error message (oC11): Inverter cannot follow dynamic processes, i.e. too short acceleration/deceleration times in terms of load ratios.	• Increase the gain of the I_{max} controller (C00073/1) • Reduce the reset time of the I_{max} controller (C00074/1) • Prolong the acceleration time (C00012) • Prolong the deceleration time (C00013)
Motor stalling in the field weakening range (adaptation especially required for small machines)	• Reduce the override point of field weakening (C00080) • If motor power < inverter power: Set C00022 to $I_{max} = 2\ I_{rated\ motor}$ • Reduce dynamic performance of setpoint generation

5.5 V/f characteristic control - energy-saving (VFCplusEco)

With the energy-saving V/f characteristic control mode (VFCplusEco), the motor voltage of the inverter is detected by means of a linear characteristic depending on the field frequency to be created or the motor speed. Moreover, a cosφ control and the resulting voltage reduction causes the motor to be always operated in the optimum efficiency range (reduction of copper losses in the asynchronous motor).

- Hence, these are the advantages of this motor control mode:
  - Good robustness
  - Easy parameter setting
  - High energy efficiency (lower heating of the motor in partial load operational range)
  - Same speed accuracy and maximum torques as with VFCplus
  - Less noise generation of the motor with active voltage reduction
- Predestinated application areas of this motor control mode are materials handling technology and pump and fan systems.
- This motor control mode serves to improve efficiency of standard asynchronous motors with efficiency class IE1 (standard IEC 60034-30 2008) in the range 0 ... $M_{efficiency\_max}$ between 0 ... 20 % (Ø 5 ... 10 %).
  - For asynchronous motors with energy efficiency class IE2 the potential for efficiency improvement is reduced to approx. 0 ... 15 %.
  - Description of $M_{efficiency\_max}$: Indicates the torque [%] of M_{rated_motor}, where the motor has the max. efficiency.)
- In case of asynchronous motors with a higher energy efficiency class (IE2 and IE3), the absolute energy saving of the motor control mode is lower due to improved efficiency of the machine. However, energy saving is still achieved in a higher load range.
- $M_{efficiency\_max}$ is performance-related and listed in the following table for some power values of the energy efficiency class IE1 and IE2:

	$M_{Efficiency\_max}$ (related to M_{rated_motor})	
Performance	**IE1**	**IE2**
0.25 kW	75 %	
0.75 kW	65 %	75 %
2.2 kW	55 %	85 %
7.5 kW	30 %	45 %
22 kW	23 %	
45 kW	21 %	

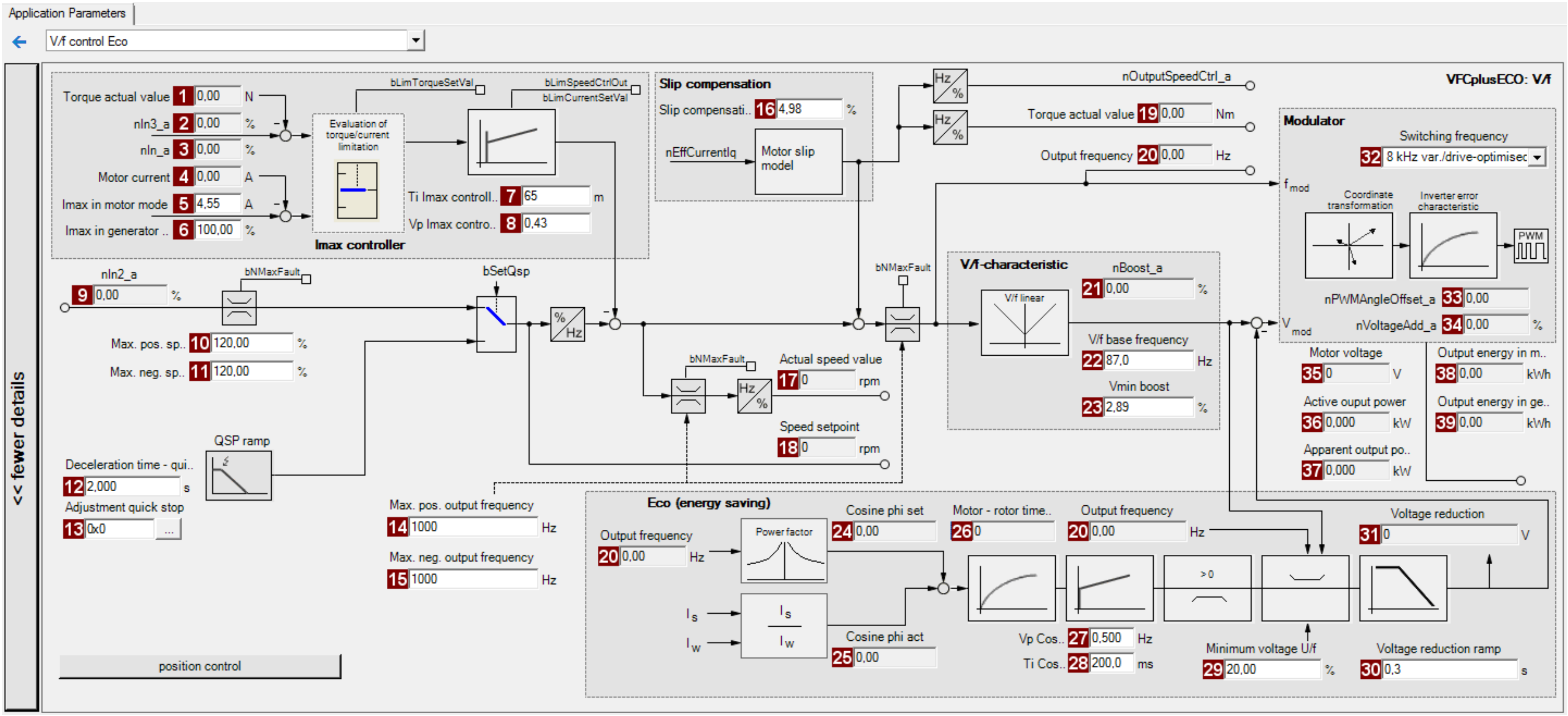
Stop!

- For adjustment, observe the thermal performance of the connected asynchronous motor at low output frequencies.
  - Usually, standard asynchronous motors with insulation class B can be operated for a short time with their rated current in the frequency range 0 Hz ... 25 Hz.
  - Contact the motor manufacturer to get the exact setting values for the max. permissible motor current of self-ventilated motors in the lower speed range.
- The nameplate data of the motor (at least rated speed and rated frequency) must be entered if, instead of a standard motor, an asynchronous motor is used with the following values:
  - rated frequency ≠ 50 Hz (star) or
  - rated frequency ≠ 87 Hz (delta) or
  - number of pole pairs ≠ 2

5.5.1 Parameterisation dialog/signal flow

Proceed as follows to open the dialog for parameterising the motor control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Select the motor control "11: VFCplusEco: V/f energy-saving" from the *Overview* dialog box in the **Motor control** list field:
4. Click the **Motor control V/f Eco** button to change to the *Overview* → *Motor control V/f* dialog box.
   - This dialog level only shows a simplified signal flow with the most important parameters.
   - When you click the **>>More details** button in the left-most position, a signal flow with more details/parameters is displayed.

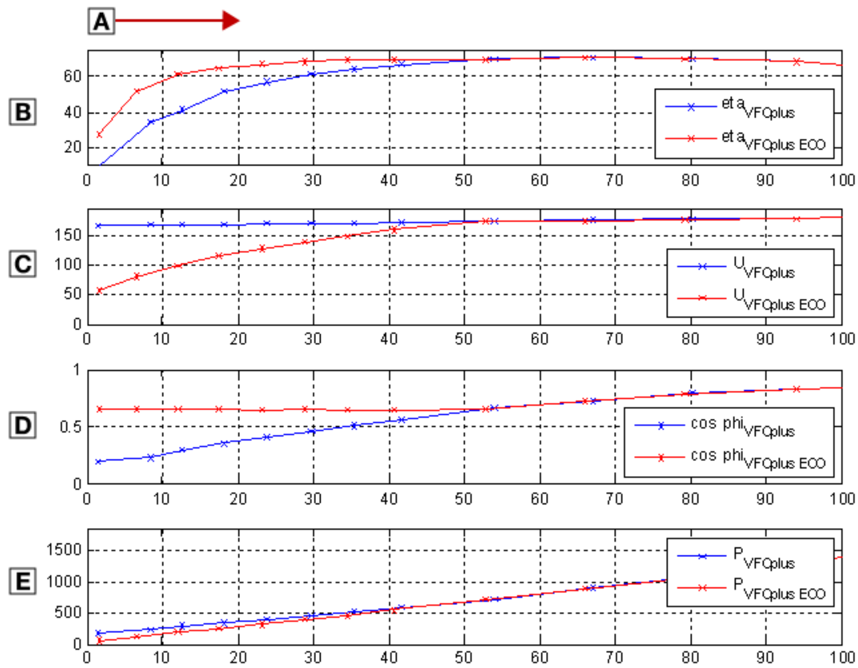


Parameters		Info	Parameters		Info	Parameters		Info
1	C00056/2	Actual torque	14	C00910/1	Max. pos. output frequency	27	C00975	VFC-ECO: Vp
2	C00830/4	Limitation of torque in motor mode	15	C00910/2	Max. neg. output frequency	28	C00976	VFC-ECO: Ti
3	C00830/5	Limitation of torque in generator mode	16	C00021	Slip compensation	29	C00977	VFC-ECO: Minimum voltage V/f
4	C00054	Motor current	17	C00051	Actual speed value	30	C00982	VFC-ECO: Voltage reduction ramp
5	C00022	Imax in motor mode	18	C00050	Speed setpoint	31	C00978	VFC-ECO: Voltage reduction
6	C00023	Imax in generator mode	19	C00056/2	Actual torque	32	C00018	Switching frequency
7	C00074/1	Ti Imax controller	20	C00058	Output frequency	33	C00830/32	MCTRL: nPWMAngleOffset_a
8	C00073/1	Vp Imax controller	21	C00830/26	MCTRL: nBoost_a	34	C00830/31	MCTRL: nVoltageAdd_a
9	C00830/3	Speed setpoint	22	C00015	V/f base frequency	35	C00052	Motor voltage
10	C00909/1	Max. pos. speed	23	C00016	Vmin boost	36	C00980/1	Active output power
11	C00909/2	Max. neg. speed	24	C00979/2	Cosine phi set	37	C00980/2	Apparent output power
12	C00105	Decel. time - quick stop	25	C00979/1	Cosine phi act	38	C00981/1	Output energy in motor mode
13	C00104/1	Quick stop setting	26	C00083	Motor rotor time constant	39	C00981/2	Output energy in generator mode

5.5.2 Comparison of VFCplusEco - VFCplus

The following characteristics show the impact of the energy-saving V/f characteristic control (VFCplusEco) compared to the standard V/f characteristic control (VFCplus).

- The characteristics were recorded with a standard asynchronous motor 2.2 kW with energy efficiency class IE1 at speed = 600 rpm.



A scaled torque in [%]
B efficiency in [%]
C motor voltage in [V]
D power factor
E electrical power in [W]

[5-8] Comparison of VFCplusEco - VFCplus

5.5.3 Basic settings

The "Initial commissioning steps" listed in the table below are sufficient for the V/f characteristic control - energy-saving (VFCplusECo).

- Detailed information on the individual steps can be found in the following subchapters.

<table>
<tr><th colspan="3">Initial commissioning steps</th></tr>
<tr><td>1</td><td colspan="2">Determine the motor control:
C00006 = "11: VFCplusEco: V/f energy-saving"</td></tr>
<tr><td rowspan="3">2.</td><td colspan="2">The required motor data are pre-initialised depending on the device and thus, they do not need to be entered directly.In order to achieve a high energy optimisation, these motor data can be entered (see the following section).</td></tr>
<tr><td colspan="2">Set the motor selection/motor data
• When selecting and parameterising the motor, the motor nameplate data and the equivalent circuit diagram data are relevant. Detailed information can be found in the chapter "Motor selection/Motor data". (📖 144)
Depending on the motor manufacturer, proceed as follows:</td></tr>
<tr><td>Lenze motor:
Selecting a motor from the motor catalogue in the »Engineer«
- or -
1. Set the motor nameplate data
2. Automatic motor data identification</td><td>Third party manufacturer's motor:
1. Set the motor nameplate data
2. Automatic motor data identification or set known equivalent circuit diagram manually:
C00084: Motor stator resistance
C00085: Motor stator leakage inductance
C00092: Motor magnetising inductance</td></tr>
<tr><td>3.</td><td colspan="2">Defining current limits (Imax controller). (📖 172)</td></tr>
</table>

Tip!

Information on the optimisation of the control mode and the adaptation to the real application is provided in chapter "Optimising the control mode". (📖 191)

Parameterisable additional functions are described in the chapter entitled "Parameterisable additional functions". (📖 272)

5.5.4 Optimising the control mode

The V/f characteristic control - energy-saving (VFCplus) is generally ready for operation. It can be adapted subsequently by adapting the characteristic and/or the drive behaviour.

Note!

Following successful motor parameter identification, the V/f base frequency (C00015) and the V_{min} boost (C00016) as well as the slip constant for slip compensation (C00021) are calculated automatically.

From version 12.00.00:

- Following successful motor parameter identification, the gain of the Imax controller (C00073/1) is calculated automatically.
  - If these parameters are not to be calculated, bit 4 of C02865/1 must be set to "1".
- Following successful motor parameter identification, other controller parameters (C00011, C00022, C00966, C00982) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "6".

Adapting characteristic

For the linear characteristic as part of the V/f characteristic control - energy-saving (VFCplusEco), it is also possible (like in case of the standard V/f characteristic control) to match its curve to different load profiles or motors by adapting the V/f base frequency (C00015) and the V_{min} boost (C00016).

▶ Adapting the V/f base frequency (📖 174)

▶ Adapting the Vmin boost (📖 176)

Adapting drive behaviour

- Limitation of the maximum current by a current limitation controller (e.g. to prevent the motor from stalling or to limit to the maximally permissible motor current). ▶ Optimising the Imax controller (📖 178)
- Adaptation of the field frequency by a load-dependent slip compensation (improved speed accuracy for systems without feedback).
- Improving the behaviour at high dynamic load changes. (📖 192)
- Adapting the slope limitation for lowering the Eco function. (📖 193)
- Optimising the cos/phi controller. (📖 193)

Torque limitation

Limit the torque to a lower value. ▶ Torque limitation (📖 180)

5.5.4.1 Improving the behaviour at high dynamic load changes

Due to the voltage reduction executed via the cosφ control, the motor may stall in the Lenze setting at high dynamic load torque changes. This is caused by the flux reduction and the connected reduction of the stalling torque of the motor current:

$$M_{Max(t)} = M_{Stalling} \cdot \frac{U^2_{Motor(t)}}{(U_{Motor(t)} - U_{Reduction})^2} \qquad \text{with } M_{Stalling} = 1.6 \ldots 2.5 \cdot M_{Rated\_motor}$$

V_{Motor} = display in C00052
$V_{Reduction}$ = display in C00978

It generally applies that when the output voltage is divided in halves, the maximum torque is approx. reduced by the factor 4. A reduction by the factor 3 reduces the torque to approx. 15 %.

The minimum voltage and thus the maximum influence access of the Eco function on the output voltage can be defined in C00977. With full influence of the Eco function, the following stalling torque can be ensured depending on the setting in C00977:

Minimum voltage V/f (C00977)	Maximum torque
100 %	160 % ... 250 % M_{rated}
70 %	80 % ... 130 % M_{rated}
50 %	40 % ... 70 % M_{rated}
20 %	15 % ... 50 % M_{rated}

An adaptation of the minimum voltage V/f (C00977) improves the stability in case of load impulses.

- In the Lenze setting, the minimum voltage V/f is set to 20 % for the highest energy optimisation. This setting serves to respond to load torques if these amount to approx. 25 % of the rated torque or occur with low dynamics.
- An increase of the minimum voltage V/f to 70 % permits to apply a dynamic load impulse from 0 to 100 % rated motor torque without the motor stalling. This reduces the energy optimisation to be achieved by approx. 75 %.
- A further increase of the stability at still higher dynamic load impulses can be achieved by a further increase of the minimum voltage V/f, but means a further loss in energy optimisation.

Note!

In case of applications with very high dynamic sudden load variations from the unloaded operation, this motor control mode should not be used or the energy optimisation should be switched off, since a motor stalling cannot be excluded.

- The energy optimisation can be switched off by setting the minimum voltage V/f (C00977) to 100 %. Then, the behaviour corresponds to the V/f characteristic control (VFCplus) with linear characteristic.
- From version 13.00.00, the energy optimisation can be switched off via the *bVfcEcoDisable* process signal if a dynamic load change is known to take place.

5.5.4.2 Adapting the slope limitation for lowering the Eco function

The ramp set in C00982 for voltage reduction serves as slope limitation in order to prevent that voltage is suddenly applied to the motor when the Eco function is deactivated. Otherwise, the overvoltage limitation (Imax, Clamp) would be activated.

- This ramp is, depending on the device, pre-initialised to approx. the triple rotor time constant. An adaptation of this parameter is not required.

When the Eco function is switched off, a quick reaction (high dynamic performance) is required, but with a low current overshoot and a small torque jump. Thus, the Lenze setting of C00982 is a compromise regarding the switch-off of the Eco function (voltage reduction = 0).

- To increase the dynamics when switching off the Eco function:
  Reduce ➞ setting in C00982.
  (Current compensation actions increase when the Eco function is switched off.)
- In order to reduce current compensation actions when switching off the Eco function:
  Increase ➞ setting in C00982.
  (The dynamics when switching off the eco function is reduced)

5.5.4.3 Optimising the cos/phi controller

With the Lenze setting, the cosφ controller is set such that usually no adaptation is required for all power ratings and application cases.

Behaviour	Remedy/recommendation
The cosφ actual value (C00979/1) varies greatly.	Reduce gain Vp (C00975) and reset time Ti (C00976).
The cosφ actual value (C00979/1) is permanently lower than the cosφ setpoint (C00979/2).	Increase gain Vp (C00975) and reset time Ti (C00976).

5.5.5 Remedies for undesired drive behaviour

Drive behaviour	Remedy
Inadequate smooth running at low speeds, especially in the case of operation with a long motor cable	▸ Automatic motor data identification (📖 151) Reduce the influence of the Eco function by increasing the minimum voltage V/f (C00977) if necessary.
Problems in case of high starting duty (great mass inertia)	1. Set motor control VFCplus with linear characteristic (C00006 = 6). 2. Adapting the Vmin boost. (📖 176) 3. Again set motor control VFCplusEco (C00006 = 11).

Drive does not follow the speed setpoint	The current controller intervenes in the set field frequency to limit the controller output current to the maximum current (C0022, C0023). Therefore:
	• Prolong acceleration/deceleration times:
	C00012: Accel. time - main setpoint
	C00013: Decel. time - main setpoint
	• Consider a sufficient magnetising time of the motor. Depending on the motor power, the magnetising time amounts to 0.1 ... 0.2 s.
	• Increase the maximally permissible current:
	C00022: Imax in motor mode
	C00023: Imax in generator mode
	• Make adaptations for the Eco function:
	  • Improving the behaviour at high dynamic load changes. (📖 192)
	  • Adapting the slope limitation for lowering the Eco function. (📖 193)
	  • Optimising the cos/phi controller. (📖 193)

Insufficient speed constancy at high load (setpoint and motor speed are not proportional anymore)	• Increase slip compensation (C00021). Important: Unstable drive due to overcompensation! • With cyclic load impulses (e. g. centrifugal pump), a smooth motor characteristic is achieved by smaller values in C00021 (possibly negative values). Note: The slip compensation is only active for operation without speed feedback.

"Clamp operation active" error message (oC11): Inverter cannot follow dynamic processes, i.e. too short acceleration/deceleration times in terms of load ratios.	• Increase the gain of the I_{max} controller (C00073)
	• Reduce the reset time of the I_{max} controller (C00074)
	• Prolong the acceleration time (C00012)
	• Prolong the deceleration time (C00013)
	• Make adaptations for the Eco function:
	  • Improving the behaviour at high dynamic load changes. (📖 192)
	  • Adapting the slope limitation for lowering the Eco function. (📖 193)

Motor stalling in the field weakening range (adaptation especially required for small machines)	• If motor power < inverter power:
	Set C00022 to $I_{max} = 2\ I_{rated\ motor}$
	• Reduce dynamic performance of setpoint generation
	• Make adaptations for the Eco function:
	  • Improving the behaviour at high dynamic load changes. (📖 192)
	  • Adapting the slope limitation for lowering the Eco function. (📖 193)

Speed variations in no-load operation for speeds > 1/3 rated speed.	Minimise speed oscillations with oscillation damping (C00234).

Drive behaviour	Remedy
Speed variations in no-load operation and with load for speeds > rated speed.	Minimise speed oscillations with increasing the oscillation damping field weakening (C00236). Caution: If C00236 is increased, the maximum output voltage of the device is reduced!
Output voltage is too low. There is a too low maximum torque in the high field weakening range.	Reduction of the oscillation damping field weakening (C00236). Caution: When C00236 = 0, oscillation damping field weakening is inactive. Thus, a maximum output voltage is available but the tendency to speed oscillations in the field weakening range at no-load operation and with load increases.

5.6 V/f control (VFCplus + encoder)

The V/f characteristic control (VFCplus) described above can be operated with a speed feedback. This has the following advantages:

- Steady-state accuracy of the speed
- Less parameterisation effort compared to the sensorless vector control (SLVC)
- Improved dynamics compared to V/f characteristic control without feedback or to sensorless vector control (SLVC).
- Suitability for group drives

Stop!

The V/f control requires a speed feedback!

- The speed sensor used has to be set in C00495. This setting is not made automatically with the selection of the motor from the »Engineer« motor catalogue!
- If no speed sensor is set in C00495 and the controller is enabled, an impermissibly high motor current occurs which may destroy the motor thermally!
- From version 14.00.00, the error response set in C00571/2 (Lenze setting: "Fault") takes place if in case of controller enable it is detected that a motor control type with feedback is set in C00006 but no speed encoder is set in C00495.

Stop!

V/f emergency operation

From version 15.00.00 , it is internally switched to the encoderless V/f characteristic control in case of an encoder open circuit in order to avoid impermissible motor movements.

- In order that this "V/f emergency operation" works properly, the parameters relevant for the V/f characteristic control (base frequency, Vmin boost, slip compensation, etc.) have to be set correctly. As an alternative, a motor parameter identification can be executed as well.
- The change-over to "V/f emergency operation" is reported via bit 4 in C01000 and via the *bWirebreakUfLinearActive* status signal at the SB LS_DeviceMonitor.
- The change-over to "V/f emergency operation" can be suppressed by setting bit 8 to "1" in C02864/1.
- When selecting the speed sensor and position encoder for the encoder signal *FreqIn12* or *FreqIn67* (C00495 = 1 / 2 and/or C00490 = 1 / 2)
  - "Latching of open circuit" must be performed: C02864, Bit11 = 1.
    This causes the system to switch to the encoderless V/f characteristiic in case of wire breakage.
  - The output speed will drop briefly and then start up again with an encoderless V/f characteristic control. V/f emergency operation will only be exited after CINH.

Note!

As the slip is calculated in the feedback V/f operation and injected through the slip regulator, the slip compensation (C00021) is deactivated with V/f control.

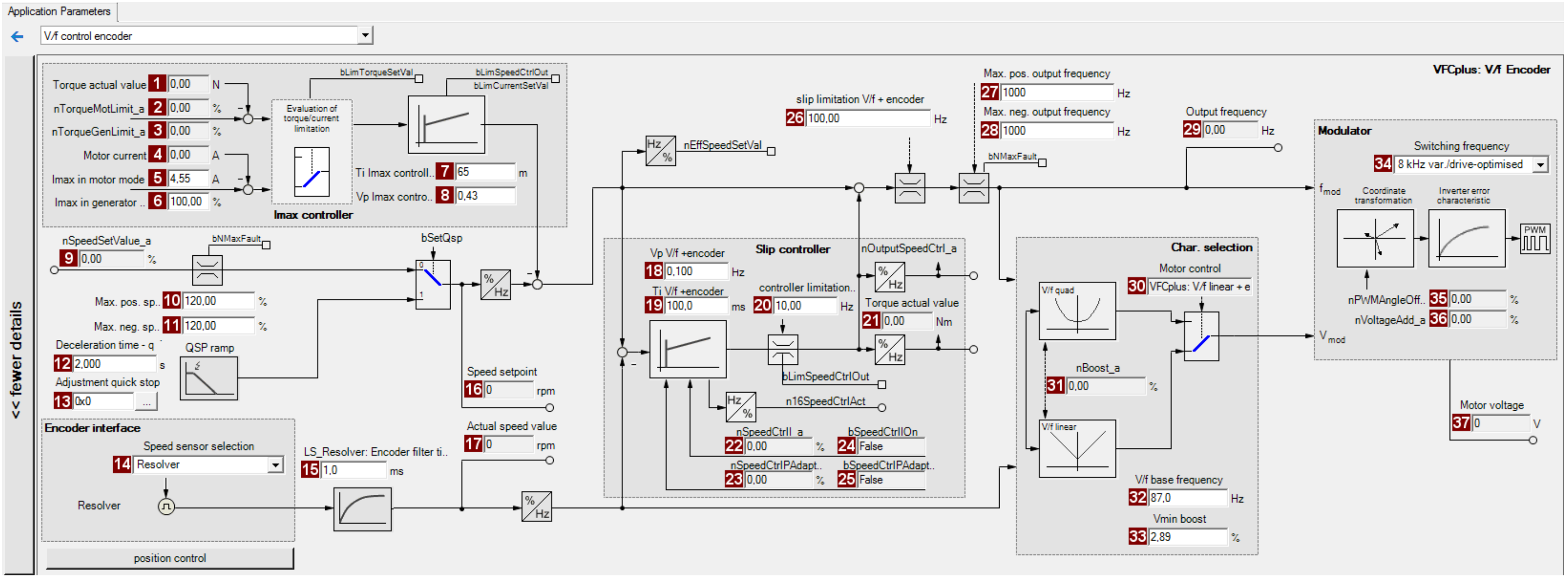
The descriptions in chapter "V/f characteristic control (VFCplus)" also apply to the V/f control. (📖 167)

Detailed information on the speed feedback can be found in the chapter "Encoder/feedback system". (📖 330)

5.6.1 Parameterisation dialog/signal flow

Proceed as follows to open the dialog for parameterising the motor control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Select the motor control from the *Overview* dialog level in the **Motor control** (C00006) list field:
   - "7: VFCplus: V/f linear +encoder" for linear characteristic or
   - "9: VFCplus: V/f quadr +encoder" for quadratic characteristic
4. Click the **Motor control V/f encoder** button to change to the *Overview* → *Motor control V/f* dialog box.
   - This dialog level only shows a simplified signal flow with the most important parameters.
   - When you click the **>>More details** button in the left-most position, a signal flow with more details/parameters is displayed, as shown in the following subchapter.



No.	Parameters	Info
1	C00056/2	Actual torque
2	C00830/29	Limitation of torque in motor mode
3	C00830/28	Limitation of torque in generator mode
4	C00054	Motor current
5	C00022	Imax in motor mode
6	C00023	Imax in generator mode
7	C00074/1	Ti Imax controller
8	C00073/1	Vp Imax controller
9	C00830/22	Speed setpoint
10	C00909/1	Max. pos. speed
11	C00909/2	Max. neg. speed
12	C00105	Decel. time - quick stop
13	C00104/1	Quick stop setting
14	C00495	Speed sensor selection
15	C00497/4	Encoder filter time FreqIn12
16	C00050	Speed setpoint
17	C00051	Actual speed value
18	C00972	Vp Vf+encoder
19	C00973	Ti Vf+encoder
20	C00971/1	Controller limitation Vf+encoder
21	C00056/2	Actual torque
22	C00830/24	MCTRL: nSpeedCtrlI_a
23	C00830/25	MCTRL: nSpeedCtrlPAdapt_a
24	C00833/31	MCTRL: bSpeedCtrlIOn
25	C00833/69	MCTRL: bSpeedCtrlPAdaptOn
26	C00971/2	Slip limitation Vf+encoder
27	C00910/1	Max. pos. output frequency
28	C00910/2	Max. neg. output frequency
29	C00058	Output frequency
30	C00006	Motor control
31	C00830/26	MCTRL: nBoost_a
32	C00015	V/f base frequency
33	C00016	Vmin boost
34	C00018	Switching frequency
35	C00830/32	MCTRL: nPWMAngleOffset_a
36	C00830/31	MCTRL: nVoltageAdd_a
37	C00052	Motor voltage

5.6.2 Basic settings

In order to protect the drive system, carry out the commissioning of the V/f control and the slip regulator in several steps.

- Detailed information on the single steps can be found in the following subchapters or in the corresponding subchapters for V/f characteristic control.

Initial commissioning steps	
1	Define the V/f characteristic: • C00006 = 7: Linear characteristic • C00006 = 9: Quadratic characteristic
2.	Defining current limits (Imax controller). (🕮 172)
3.	Parameterise encoder/feedback system. ▸ Encoder/feedback system (🕮 330)
4.	If special motors with a rated frequency other than 50 Hz or with a number of pole pairs ≠ 2 are used, set the motor parameters according to the motor nameplate. ▸ Motor selection/Motor data (🕮 144)
5th	Define speed setpoint (e.g. 20 % of the rated speed) and enable inverter.
6.	Check whether the actual speed value (C00051) ≈ speed setpoint (C00050) and then inhibit the inverter again. • In case of a sign reversal between actual value and setpoint, check the connection or setting of the encoder. • In case the actual value differs considerably from the setpoint (factor 2), set the motor parameters according to motor nameplate. Then repeat step 5.
6.	To protect the drive, reduce the slip regulator limitation in C00971/1. • e.g. reduction to half the slip frequency (≈ 2 Hz)
8	Define speed setpoint (e.g. 20 % of the rated speed) and enable inverter.
9	In case of a semi-stable operational performance, reduce the reset time (C00972) or the proportional gain (C00973) of the slip regulator until a stable operation has been achieved. ▸ Parameterising the slip regulator (🕮 200)
10	In a final step, increase the slip regulator limitation again in C00971/1. • e.g. increase to twice the slip frequency

Tip!

Information on the optimisation of the control mode and the adaptation to the real application is provided in chapter "Optimising the control mode". (🕮 173)

Parameterisable additional functions are described in the chapter entitled "Parameterisable additional functions". (🕮 272)

5.6.2.1 Parameterising the slip regulator

The slip regulator is designed as a PI controller. In order to improve the response to setpoint changes, the setpoint speed or setpoint frequency is added to the output (correcting variable) of the slip regulator as feedforward control value.

- Unlike traditional speed controllers, the slip regulator only controls the slip.
- In the Lenze setting, the configuration of the slip regulator provides robustness and moderate dynamics.

Note!

By increasing the smoothing time of the actual speed measurement in C00497, you can reduce occurring vibrations. This measure may be particularly useful when using low-pulse HTL encoders.

Parameters	Info	Lenze setting	
		Value	Unit
C00971/1	VFC: Controller limitation V/f +encoder	10.00	Hz
C00971/2	VFC: Slip limitation V/f +encoder	100.00	Hz
C00972	VFC: Vp V/f +encoder	0.100	Hz/Hz
C00973	VFC: Ti V/f +encoder	100.0	ms

Slip regulator gain Vp

The setting range of the slip regulator gain Vp (C00972) which leads to a stable operational performance, mainly depends on the resolution of the speed sensor. There is a direct relationship between encoder resolution and gain:

- The higher the encoder resolution, the higher the gain can be set.

The following table provides maximum and recommended slip regulator gains for encoder with standard encoder increments:

Encoder increment [Increments/revolution]	**Slip regulator gain Vp**	
	maximum	recommended
8	0.09	0.06
64	0.52	0,31
100	0.79	0.47
120	0.94	0.57
128	1.00	0.60
256	1.29	0.77
386	1.63	0.98
512	1.97	1.18
640	2.31	1.38
768	2.65	1.59
896	2.99	1.79
1014	3.33	2.00
1536	4.69	2.81
2048	6.05	3.63
3072	8.77	5.26
4096	11.49	6.90

[5-1] Slip regulator gain Vp based on the encoder increment

How to adapt the slip regulator gain to the operating conditions:

1. Adapt the slip regulator gain (C00972) to the encoder increment according to table [5-1].
2. Set controller limitation (C00971/1) to half the slip frequency (≈ 2 Hz).
3. Select speed setpoint (e.g. 20 % of the rated speed).
4. Enable inverter.
5. Increase the slip regulator gain (C00972) until the drive is semi-stable.
   - This can be recognised by motor noises or "humming" of the motor or by a noise on the actual speed signal.
6. Reduce slip regulator gain (C00972) until the drive runs stable again (no motor "humming").
7. Reduce slip regulator gain (C00972) to approx. half the value.
   - With low encoder resolutions, another reduction of the slip regulator gain for low speeds may be necessary (speed setpoint ≈ 0).
   - We recommend to finally check the behaviour at setpoint speed = 0 and to further reduce the slip regulator gain if irregular running occurs.
8. Increase controller limitation (C00971/1) again (e.g. to twice the slip frequency).

Slip regulator time constant Ti

How to set the slip regulator time constant:

1. Set controller limitation (C00971/1) to half the slip frequency (≈ 2 Hz).
2. Select speed setpoint (e.g. 20 % of the rated speed).
3. Enable inverter.
4. Reduce the slip regulator time constant (C00973) until the drive is semi-stable.
   - This can be recognised by motor noise, "motor vibrations" or resonance on the actual speed value signal.
5. Increase slip regulator time constant (C00973) until the drive runs stable again (no motor "oscillation").
6. Increase the slip regulator time constant (C00973) to approx. twice the value.
7. Increase controller limitation (C00971/1) again (e.g. to twice the slip frequency).

Controller limitation

Max. intervention of the controller is limited by the controller limitation (C00971/1).

- The controller can be limited depending on the application.
- We recommend to limit the max. intervention to twice the rated slip of the motor.
- The rated slip is calculated as follows:

$$f_{Slip_{Rated}}[Hz] = f_{Rated}[Hz] - \left(\frac{n_{Motor_{Rated}}[rpm]}{60} \cdot p_{\text{Number of pole pairs}}\right)$$

[5-9] Calculation of the rated slip

Note!

A setting of C00971/1 = 0 Hz deactivates the slip regulator. In this case, the structure of the V/f control corresponds to the structure of a V/f characteristic control without feedback.

Slip limitation

In addition to limiting the slip regulator, the field frequency to be injected can also be limited by another limiting element, the slip limitation (C00971/2).

- If the slip is e.g. limited to twice the rated slip of the motor, a stalling of the motor during very dynamic processes can be avoided.
- Motor stalling is caused by:
  - a high overcurrent at very steep speed ramps
  - very fast speed changes due to load, e.g. abrupt stopping of the drive due to an encounter with a stop or a load that is not moving.

5.7 Sensorless vector control (SLVC)

Sensorless vector control (SLVC) is based on a better motor current control according to a field-oriented control mode by Lenze.

Stop!

- The sensorless vector control (SLVC) is only suitable for asynchronous motors.
- The connected motor may be maximally two power classes lower than the motor assigned to the inverter.
- Operation of the sensorless vector control (SLVC) is only permissible for one single drive!
- Operation of the sensorless vector control (SLVC) is not permissible for hoists!
- The Lenze setting permits the operation of a power-adapted motor. Optimal operation is only possible if either:
  - the motor is selected via the »Engineer« motor catalogue,
  - the motor nameplate data are entered and motor parameter identification is carried out afterwards
    - or -
  - the nameplate data and equivalent circuit data of the motor (motor leakage inductance and mutual motor inductance, slip compensation and motor stator resistance) are entered manually.
- When you enter the motor nameplate data, take into account the phase connection implemented for the motor (star or delta connection). Only enter the data applying to the selected connection type.
  - In this context, also observe the instructions in chapter "Adapting the V/f base frequency" relating to V/f characteristic control. (🕮 174)

Note!

Optimal operation of the sensorless vector control (SLVC) can be achieved from a minimum speed of approx. 0.5-fold slip speed. At lower speed values below the 0.5-fold slip speed, the maximum torque is reduced.

The maximum field frequency with this motor control mode is 650 Hz.

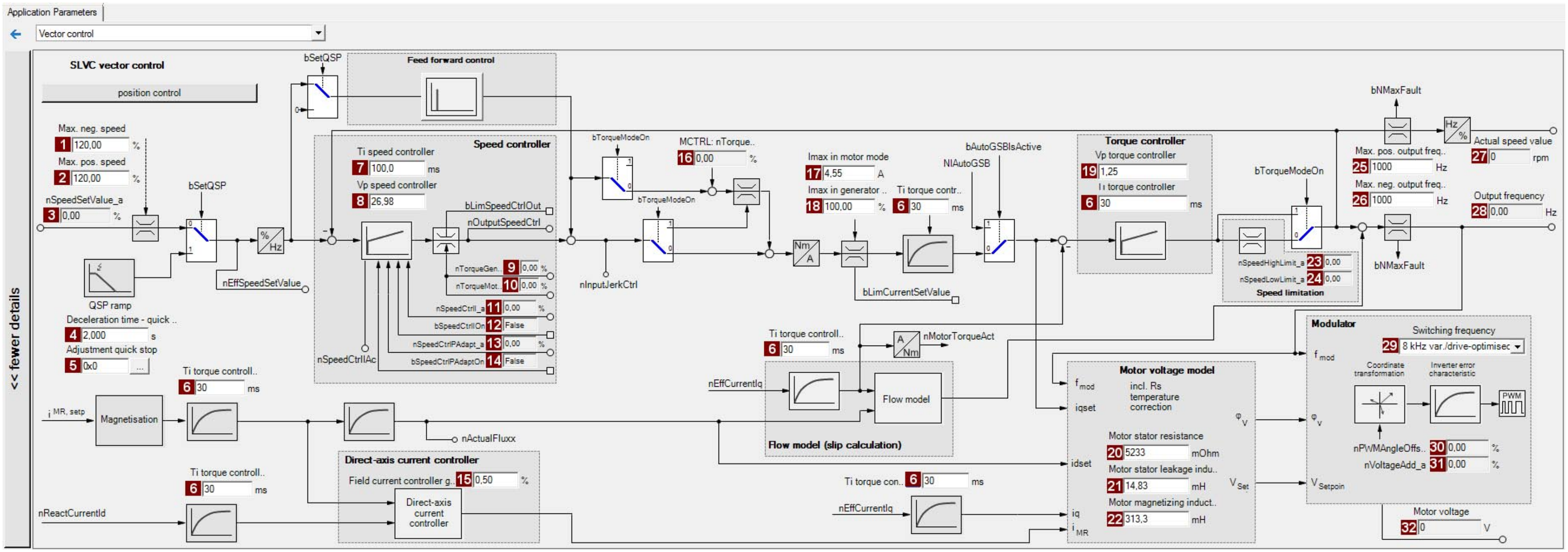
In comparison to the V/f characteristic control without feedback, the following can be achieved by means of sensorless vector control SLVC:

- A higher maximum torque throughout the entire speed range
- A higher speed accuracy
- A higher concentricity factor
- A higher level of efficiency
- The implementation of torque-controlled operation with speed limitation
- The limitation of the maximum torque in motor and generator mode for speed-controlled operation

5.7.1 Parameterisation dialog/signal flow

Proceed as follows to open the dialog for parameterising the motor control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Select the motor control "4: SLVC: Vector control" from the *Overview* dialog level in the **Motor control** list field (C00006):
4. Click the **Motor control vector** button to change to the *Overview* ➞ *Motor control vector* dialog box.
   - This dialog level only shows a simplified signal flow with the most important parameters.
   - When you click the **>>More details** button in the left-most position, a signal flow with more details/parameters is displayed, as shown in the following subchapter.



	Parameters	Info		Parameters	Info		Parameters	Info
1	C00909/2	Max. neg. speed	16	C00830/27	MCTRL: nTorqueSetValue_a	19	C00073/2	SLVC: Vp torque controller
2	C00909/1	Max. pos. speed	17	C00022	Imax in motor mode	20	C00084	Motor stator resistance
3	C00830/22	Speed setpoint	18	C00023	Imax in generator mode	21	C00085	Motor stator leakage inductance
4	C00105	Decel. time - quick stop				22	C00092	Motor magnetising inductance
5	C00104/1	Quick stop setting				23	C00830/88	MCTRL: nSpeedHighLimit_a
6	C00074/2	SLVC: Ti torque controller				24	C00830/23	MCTRL: nSpeedLowLimit_a
7	C00071/1	SLVC: Ti speed controller				25	C00910/1	Max. pos. output frequency
8	C00070/1	SLVC: Vp speed controller				26	C00910/2	Max. neg. output frequency
9	C00830/28	Limitation of torque in generator mode				27	C00051	Actual speed value
10	C00830/29	Limitation of torque in motor mode				28	C00058	Output frequency
11	C00830/24	MCTRL: nSpeedCtrlI_a				29	C00018	Switching frequency
12	C00833/31	MCTRL: bSpeedCtrlIOn				30	C00830/32	MCTRL: nPWMAngleOffset_a
13	C00830/25	MCTRL: nSpeedCtrlPAdapt_a				31	C00830/31	MCTRL: nVoltageAdd_a
14	C00833/69	MCTRL: bSpeedCtrlPAdaptOn				32	C00052	Motor voltage
15	C00985	SLVC: Gain of field current controller						

5.7.2 Types of control

The sensorless vector control can be operated in two different modes:

- Speed control with torque limitation (*bTorquemodeOn* = FALSE)
- Torque control with speed limitation (*bTorquemodeOn* = TRUE)

5.7.2.1 Speed control with torque limitation

When *bTorquemodeOn* = FALSE, the drive system is operated with a selected speed setpoint in a speed-controlled manner.

Note!

From version 13.00.00, the torque setpoint *nSpeedSetValue_a* is set to 0 by quick stop (QSP) device-internally and the torque limit values *nTorqueMotLimit_a* and *nTorqueGenLimit_a* are set to 100 % to stop the drive quickly and safely anytime. The previous behaviour can be set in C2865/1 via bit 12 and bit 13.

The operational performance can be adapted in the following ways:

A. Overload limitation in the drive train
B. Motor current limitation
C. Slip compensation

Overload limitation in the drive train

The torque is limited via the torque setpoint.

- The torque setpoint is identical to the value at the output of the speed controller, *nOutputSpeedCtrl*.
- To avoid overload in the drive train, the torque in motor mode can be limited via the *nTorqueMotLimit_a* process input signal, and the torque in generator mode can be limited via the *nTorqueGenLimit_a* process input signal:

Designator DIS code | data type	Information/possible settings
nTorqueMotLimit_a C00830/29 | INT	Torque limitation in motor mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: 0 ... +199.99 % • If keypad control is performed: Parameterisable via C00728/1. From version 18.00.00 onwards: C02864: Bit 15 = 1: positive torque limitation (nTorqueHighLimit_a)
nTorqueGenLimit_a C00830/28 | INT	Torque limitation in generator mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: -199.99 ... 0 % • If keypad control is performed: Parameterisable via C00728/2. From version 18.00.00 onwards: C02864: Bit 15 = 1: negative torque limitation (nTorqueLowLimit_a)

Note!

To avoid instabilities during operation, the torque limit values are internally processed as absolute values.

Motor current limitation

A cross current setpoint is calculated from the torque setpoint which is limited depending on the magnetising current, the max. current in motor mode (C00022) and the max. current in generator mode (C00023). The total current injected into the motor does not exceed the max. currents in motor mode and in generator mode.

Slip compensation

The slip of the machine is reconstructed using the slip model. An influencing parameter is the slip constant (C00021). ▶ Slip compensation (📖 287)

5.7.2.2 Torque control with speed limitation

When *bTorquemodeOn* = TRUE, a torque-controlled operation is activated. The setpoint torque directly follows the default value *nTorqueSetValue_a*.

Due to its speed limitation, the torque-controlled drive can only rotate within a speed range whose positive speed is limited by *nSpeedHighLimit_a* and whose negative speed is limited by *nSpeedLowLimit_a*.

Note!

- Absolute speed limitation to speed 0 rpm (*nSpeedLowLimit_a* or *nSpeedHighLimit_a* = 0) is only possible from version 12.00.00.
- Quick stop (QSP) is used to switch over to Speed control with torque limitation.
  - From version 13.00.00, the torque setpoint *nSpeedSetValue_a* is set to 0 by quick stop (QSP) device-internally and the two torque limit values *nTorqueMotLimit_a* and *nTorqueGenLimit_a* are set to 100 % to stop the drive quickly and safely anytime. The previous behaviour can be set in C2865/1 via bit 12 and bit 13.

- From version 13.00.00, the *bLimSpeedTorquemodeOn* status signal is used to show that the speed limitation is active.
- The speed is defined by the process.
- The torque setpoint is calculated directly from *nTorqueSetValue_a*.
  - From version 12.00.00 onwards, the torque limitation is active via *nTorqueMotLimit_a* and *nTorqueGenLimit_a* in this control mode, too, for the torque setpoint to be limited. The torque limitation can be deactivated in C2865/1 via bit 0 to obtain the previous function.

Designator DIS code | data type	**Information/possible settings**
nTorqueSetValue_a C00830/27 | INT	Torque setpoint / additive torque • Scaling: 16384 ≡ 100 % M_{max} (C00057)
nSpeedHighLimit_a C00830/88 | INT	Upper speed limit for the speed limitation • During torque-controlled operation only (*bTorquemodeOn* = TRUE) • Scaling: 16384 ≡ 100 % rated speed (C00011)
nSpeedLowLimit_a C00830/23 | INT	Lower speed limit for speed limitation • During torque-controlled operation only (*bTorquemodeOn* = TRUE) • Scaling: 16384 ≡ 100 % rated speed (C00011)

Designator DIS code | data type	**Information/possible settings**
nTorqueMotLimit_a C00830/29 | INT	Torque limitation in motor mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: 0 ... +199.99 % • If keypad control is performed: Parameterisable via C00728/1. From version 18.00.00 onwards: C02864: Bit 15 = 1: positive torque limitation (nTorqueHighLimit_a)
nTorqueGenLimit_a C00830/28 | INT	Torque limitation in generator mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: -199.99 ... 0 % • If keypad control is performed: Parameterisable via C00728/2. From version 18.00.00 onwards: C02864: Bit 15 = 1: negative torque limitation (nTorqueLowLimit_a)

5.7.3 Basic settings

The following "Initial commissioning steps" must be performed to commission the sensorless vector control:

<table>
<tr><th colspan="3">Initial commissioning steps</th></tr>
<tr><td>1</td><td colspan="2">Determine the motor control:
C00006 = "4: SLVC: Vector control"</td></tr>
<tr><td>2.</td><td colspan="2">Set the motor selection/motor data
• When selecting and parameterising the motor, the motor nameplate data and the equivalent circuit diagram data are relevant. Detailed information can be found in the chapter "Motor selection/Motor data". (📖 144)
Depending on the motor manufacturer, proceed as follows:</td></tr>
<tr><td></td><td>Lenze motor:
Selecting a motor from the motor catalogue in the »Engineer«
- or -
1. Set the motor nameplate data
2. Automatic motor data identification</td><td>Third party manufacturer's motor:
1. Set the motor nameplate data
2. Automatic motor data identification or set known equivalent circuit diagram data manually:
C00082: Motor rotor resistance
C00084: Motor stator resistance
C00085: Motor stator leakage inductance
C00092: Motor magnetising inductance
C00095: Motor magnetising current</td></tr>
<tr><td>3.</td><td colspan="2">Define the type of control:
bTorquemodeOn = FALSE: Speed control with torque limitation
bTorquemodeOn = TRUE: Torque control with speed limitation</td></tr>
<tr><td>4.</td><td colspan="2">Set the slip compensation (C00021). ▸ Slip compensation (📖 287)</td></tr>
</table>

Tip!

Information on the optimisation of the control mode and the adaptation to the real application is provided in chapter "Optimising the control mode". (📖 211)

We recommend to use the flying restart function for connecting/synchronising the inverter to an already rotating drive system. ▸ Flying restart function (📖 280)

Parameterisable additional functions are described correspondingly in the chapter "Parameterisable additional functions". (📖 272)

5.7.4 Optimising the control mode

Note!

From version 12.00.00:

- Following successful motor parameter identification, the current controller parameters (C00075, C00076) are calculated automatically.
  - If these parameters are not to be calculated, bit 4 of C02865/1 must be set to "1".
- Following successful motor parameter identification, the speed controller parameters (C00070/1, C00071/1) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "5".
- Following successful motor parameter identification, other controller parameters (C00011, C00022) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "6".

5.7.4.1 Optimising the starting performance after a controller enable

After the inverter has been enabled, the starting action of the motor is delayed due to the magnetisation of the motor. Under consideration of the motor rotor time constant (C00083), the time delay is calculated as follows:

Magnetisation = 1.5 * motor rotor time constant

If this delay cannot be tolerated for specific applications, the motor must always be operated in an energised condition. For this, select one of the following options:

Procedure without setting a controller inhibit

1. Deactivate the auto DCB function with C00019 = 0.
2. Do not activate the controller inhibit. Instead, stop the drive by selecting a setpoint of 0 or by activating the quick stop function.

Procedure with setting a controller inhibit due to application requirements

1. Deactivate the auto DCB function with C00019 = 0.
2. Enter a greater value for the motor rotor resistance (max. factor 2!) to reduce the magnetisation time in C00082.

Note!

During the starting action, a jerk may occur in the machine due to the temporarily increased motor current!

5.7.4.2 Optimise speed controller

The speed controller is designed as a PI controller.

- In the Lenze setting, the configuration of the speed controller provides robustness and moderate dynamics.

Parameters	Info	Lenze setting	
		Value	Unit
C00070/1	SLVC: Vp speed controller	15.00	
C00071/1	SLVC: Ti speed controller	100.0	ms

Speed controller gain Vp

The gain Vp (C00070/1) of the speed controller is defined in a scaled representation which enables a comparable parameterisation almost independent of the power of the motor or inverter. Here, the speed input difference of the controller is scaled to the rated motor speed whereas the output torque refers to the rated motor torque. A gain of 10 means that a speed difference of 1 % is gained through the P component with 10 % torque.

If the rated data of the motor and the mass inertia of the drive system are known, we recommend the following setting:

$$V_P \approx 1.5 \ldots 3 \cdot \frac{T_M[s]}{0.01[s]}$$

$$T_M[s] = \frac{2 \cdot \pi \cdot n_N[rpm]}{M_N[Nm] \cdot 60} \cdot J_{Drive,\,total}[kgm^2]$$

$$M_N[Nm] = \frac{P_N[W] \cdot 60}{2 \cdot \pi \cdot n_N[rpm]}$$

V_P = Gain of the speed controller (C00070/1)
T_M = Time constant for the acceleration of the motor
M_N = Rated motor torque
n_N = Rated motor speed
$J_{drive,\,total}$ = Total moment of inertia of the drive

[5-10] Recommendation for the setting of the gain of the speed controller

Tip!

Values recommended by Lenze for the setting of the (proportional) gain:

- For drive systems without feedback: Vp = 6 ... 25
- For drive systems with a good disturbance behaviour: Vp > 15
  In this case, we recommend the optimisation of the dynamic performance of the torque controller.

Speed controller reset time Ti

Apart from setting the P component, C00071/1 provides the possibility to take influence on the I component of the PI controller.

Tip!

Value range recommended by Lenze for the setting of the reset time:

Ti = 20 ms ... 150 ms

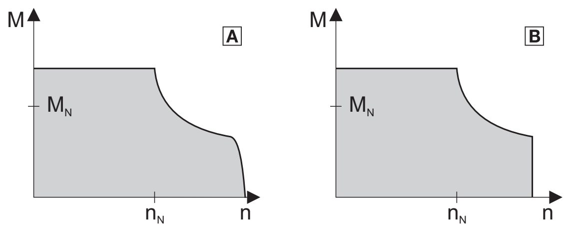
5.7.4.3 Optimising dynamic performance and field weakening behaviour

In the Lenze setting, the torque controller has been preset in such a way that robust and stable operation with a moderate dynamic response is enabled over the entire speed range. Retrospective optimisation of the controller parameters is not necessary.

Parameters	Info	Lenze setting	
		Value	Unit
C00073/2	SLVC: Vp torque controller	1.25	
C00074/2	SLVC: Ti torque controller	30	ms

A greater dynamic performance of the sensorless vector control can be achieved by reducing time constant Ti of the speed controller (C00074/2).

A greater dynamic performance of the field weakening function can be achieved by setting a time constant ≤ 15 ms. This means for actual speeds above rated speed a better torque-speed-characteristic in the field weakening range:



A Torque controller time constant Ti > 15 ms
B Torque controller time constant Ti ≤ 15 ms

[5-11] Speed / torque characteristic diagram in the field weakening range

- For Ti > 15 ms (see A), the actual speed value slightly drops in the field weakening range if the load torque increases in the motor mode.
- For Ti ≤ 15 ms (see B), the speed remains stable in the field weakening range if the torque is within the M/n characteristic field highlighted in grey.

Tip!

For applications with high dynamic performance and speed/torque accuracy requirements in the field weakening range, we recommend a time constant Ti ≤ 15 ms.

In this case, the maximum torque should be limited via the *nTorqueMotLimit_a* and *nTorqueGenLimit_a* process input signals to 1.5 x M_N to ensure stable operation in the field weakening range.

5.7.4.4 Optimising the stalling behaviour

Motor stalling due to a torque overload in the field weakening range is prevented in sensorless vector control by means of an inverter-internal stalling current monitoring. In the field weakening range, hence at frequencies above the base frequency, it reduces the maximum current to prevent the motor from stalling. The reduction depends on the current field frequency, the base frequency, the DC-bus voltage and the maximum current (C00022). Generally it applies that a higher field frequency causes a stronger limitation of the maximum current.

The field weakening behaviour of the sensorless vector control depends on the setting of the reset time Ti of the torque controller (C00074/2).

The following applies to the reset time Ti (C00074/2) > 15 ms:

The behaviour in the field weakening range can be adapted via the override point of field weakening (C00080). This parameter serves to shift the frequency-dependent maximum current characteristic:

- C00080 > 0 Hz:
  - The maximum current characteristic is shifted by the entered frequency to higher field frequencies.
  - The maximally permissible current and the maximum torque increase in the field weakening range.
  - The risk of motor stalling increases.
- C00080 < 0 Hz:
  - The maximum current characteristic is shifted by the entered frequency to lower field frequencies.
  - The maximally permissible current and the maximum torque are reduced in the field weakening range.
  - The risk of motor stalling is reduced.

Note!

We recommend to keep the Lenze setting (0 Hz).

The following applies to the reset time Ti (C00074/2) <= 15 ms:

The reduction of the magnetising current in the field weakening range can be adapted via the override point of field weakening (C00080):

- C00080 > 0 Hz:
  The reduction of the magnetising current is shifted to higher field frequencies. Here, there is a risk of the motor being magnetised too much and having too little voltage reserve for the torque-creating current.
- C00080 < 0 Hz:
  The reduction of the magnetising current is shifted to lower field frequencies.

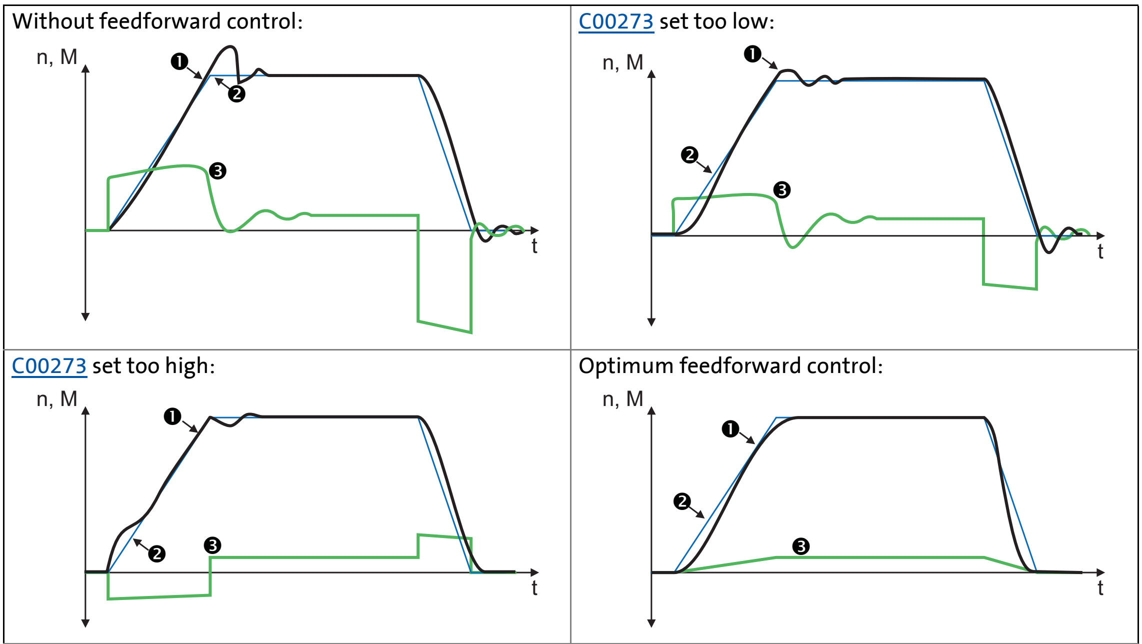
Note!

A function for enabling a stable operation can only be implemented to a limited extent with a reset time Ti <= 15 ms For applications with speeds above the 2-fold rated speed, we recommend a reset time Ti (C00074/2) > 15 ms.

5.7.4.5 Optimise response to setpoint changes and determine mass inertia

Optimisation at constant mass inertia

Setting the total moment of inertia under C00273 provides the optimum torque feedforward control. Depending on the application, an adjustment of the setting under C00273 may be necessary to optimise the response to position/speed setpoint changes by means of the torque feedforward control.



❶ Actual speed value (signal *nMotorSpeedAct_a*)
❷ Speed setpoint (*nSpeedSetValue_a* signal)
❸ Speed controller output (signal *nOutputSpeedCtrl_a*)

[5-12] Typical signal characteristics for different settings of the load moment of inertia

How to optimise the torque feedforward control:

1. Run a typical speed profile and record the inputs and outputs of the speed controller with the data logger.
   - Motor control variables to be recorded:
     nSpeedSetValue_a (speed setpoint)
     nMotorSpeedAct_a (actual speed value)
     nOutputSpeedCtrl_a (speed controller output)
2. Estimate the moment of inertia and set it in C00273 in relation to the motor end (i.e. with account being taken of the gearbox factors).
3. Repeat the data logger recording (see step 1).

   Now the data logger should show that part of the required torque is generated by the feedforward control and the speed controller output signal (*nOutputSpeedCtrl_a*) is correspondingly smaller. The resulting following error decreases.
4. Change the setting in C00273 and repeat the data logger recording until the intended response to setpoint changes is reached.
   - The optimisation could aim at the speed controller being completely relieved (see signal characteristics in Fig. [5-12]).
5. Save the parameter set (device command: C00002/11).

Optimisation at variable mass inertia

From version V12.00.00, mass inertia that changes during the process (e.g. a reel) can be taken into account when optimising the response to setpoint changes.

How to proceed:

1. In C00273 the known constant total moment of inertia (motor, gearbox, shaft, etc.) must be set or determined according to previous instructions ("How to optimise ...").
   - The determination requires travelling the typical speed profile without variable mass inertia (e.g. reels).
2. At the LS_MotorInterface SB, the *nInertiaAdapt_a* process signal must be interconnected in a way which ensures that a value of "100 %" is applied at this input.
3. In C00919/1 set the known maximum value of the variable moment of inertia or determine the value according to previous instructions ("How to optimise ...").
   - The determination requires travelling the typical speed profile including variable mass inertia (e.g. maximum reels).
4. The *nInertiaAdapt_a* process signal can be used during the process to dynamically control the percentage of the variable moment of inertia set in C00919/1 which is to be considered for setpoint feedforward control.

   Example:
   - If there is no variable moment of inertia (e.g. no reel), the *nInertiaAdapt_a* process signal must be set to "0 %".
   - If the maximally variable moment of inertia is available (e.g. maximum reels), the *nInertiaAdapt_a* process signal must be set to "100 %".

Tip!

Via the *nTorqueSetValue_a* process signal at the LS_MotorInterface SB, any differential signal can be defined for torque feedforward control. First the speed controller, then this additive torque is connected which is hence not derived from the differential change in speed setpoint.

Other functions for differential setpoint feedforward control

From version V12.00.00 onwards, the following additional functions are available for differential setpoint feedforward control (torque feedforward control):

- In C00653/1, the sensitivity of setpoint feedforward control can be adapted.
- In C00654/1, alternatively to the *nSpeedSetValue_a* process signal, the new *nSpeedSetValueInertia_a* process signal for the setpoint feedforward control can be selected at the LS_MotorInterface SB. Via the *nSpeedSetValueInertia_a* process signal, an optional input value (e.g. setpoint of the position or process controller) for the torque feedforward control can be specified.
- For *bTorqueModeOn* = TRUE, the setpoint feedforward control is added to the torque setpoint *nTorqueSetValue_a*. In this way, feedforward control of torque is also possible for torque-controlled operation (e.g. for winder applications).

From version V18.00.00 onwards, the following additional functions are available for differential setpoint feedforward control (torque feedforward control):

- In C00654/1, an exact torque feedforward control can be set for the nSpeedSetValue_a speed setpoint. In case of a very dynamic application, set C00654/1 = 2.
- From version V15.00.00 onwards, the torque feedforward control behaves differently than in version V14.00.00 or older. In order to achieve the same behaviour as in version V14.00.00 or older, set C00654/1 = 3.

Stop!

In the Lenze setting of C00654/1, the speed setpoint of the speed controller (*nSpeedSetValue_a*) is used for the torque feedforward control which is why it is also called "differential setpoint feedforward control". A very sharp change of the speed setpoint at the speed controller thus causes a strong torque impulse at the machine!

Causes for a very sharp change of the speed setpoint of the speed controller:

- A control creates the setpoint ramp itself and the speed setpoint is only written every 20 ms to the drive. (In this case, the speed setpoint is changed every 20 ms.)
  - Recommendation: Deactivate the torque feedforward control for the speed setpoint in C00654/1 if the speed setpoint changes very sharply or the mass inertia is unknown (e.g. in case of hoists)!
- The reference speed C0011 is much higher than the rated motor speed set in C0087.
  - Recommendation: Deactivate the torque feedforward control for the speed setpoint in C00654/1 if the reference speed C0011 is 5 times higher than the rated motor speed C0087.

5.7.4.6 Slip calculation from motor equivalent circuit diagram data

This function extension is available from version 02.00.00!

In order to achieve a better speed stability and torque accuracy, the slip calculation can be either derived from the motor nameplate data (e.g. rated motor speed) or the motor equivalent circuit diagram data (stator resistance, rotor resistance etc.).

The data to be used for sensorless vector control is selected via bit 0 in C02879/1:

setting		Info
Bit 0	SLVC	In case of sensorless vector control: • "0" ≡ Slip calculation from motor nameplate data (Lenze setting) • "1" ≡ Slip calculation from motor equivalent circuit diagram data
Bit 1	SC_ASM	In case of servo control for asynchronous motors: • "0" ≡ Slip calculation from motor nameplate data • "1" ≡ Slip calculation from motor equivalent circuit diagram data (Lenze setting)
Bit 2 ... 7	Reserved	

Note!

In order that the slip can be calculated from the motor equivalent circuit diagram data, the equivalent circuit data (stator resistance, rotor resistance etc.) must be known as exactly as possible.

- Selecting a motor in the »Engineer« motor catalogue loads the exact motor equivalent circuit diagram data.
- When the motor nameplate data is entered manually and the motor equivalent circuit diagram data is then detected via the motor parameter identification, the "extended identification" (C02867/1 = 2) must be used. ▶ Automatic motor data identification (📖 151)

In the slip calculation from the motor equivalent circuit diagram data, the slip compensation (C00021) has no influence anymore.

5.7.4.7 Optimising field feedforward control and torque feedforward control

At the start of an acceleration process and at the end of a deceleration process, the field current (*nReaktCurrentId_a*) may oscillate.

- Especially when accelerating via small acceleration ramps, these oscillations become obvious by the speed being unable to follow the setpoint or speed drops during acceleration.
- Oscillating can be reduced by increasing the gain of the field current controller in C00985. A small increase of the cross current controller gain in C00986 can further improve the starting performance for small acceleration ramps.

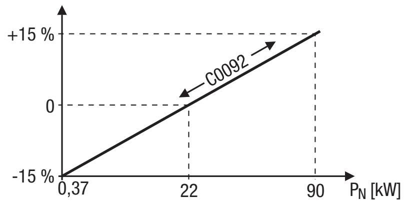
In the field weakening range, the cross current (*nEffCurrentIq_a*) may oscillate when an acceleration process starts or a deceleration process ends.

- In case of some motors, this cross current oscillation (*nEffCurrentIq_a*) can also occur in the motor setting range/field weakening range transitions which may also cause an overcurrent interruption.
- These oscillations can be reduced by again increasing the cross current controller gain slightly in C00986.

Note!

The setting of C00985 and C00986 reduces the torque setting range.

5.7.5 Remedies for undesired drive behaviour

Drive behaviour	Remedy
Deviation between no-load current and magnetising current or bad speed or torque accuracy.	Adapt the motor magnetising inductance (C00092) for no-load operation. • If the no-load current is greater than the magnetising current (C00095) at 0.5-fold rated motor speed, the magnetising inductance must be reduced until the no-load current and the magnetising current have the same values. • Otherwise, the magnetising inductance must be increased. Tendency of the correction of C00092:  PN: Rated motor power
Insufficient speed constancy at high load: Setpoint and motor speed are not proportional anymore. **Caution:** Overcompensation of the settings mentioned under " Remedy" may result in unstable behaviour!	Via the slip compensation (C00021), the speed stability under high loads can be affected: • If $n_{act} > n_{slip}$, reduce the value in C00021 • If $n_{act} < n_{slip}$, increase the value in C00021
Unstable control with higher speeds.	• Check the setting of the magnetising inductance (C00092) by comparing the current consumption in no-load operation with the rated magnetising current (C00095). • Optimise oscillation damping (C00234).
"Short circuit" (oC1) or "Clamp operation active" (oC11) error messages at short acceleration time (C00012) in proportion to the load (inverter cannot follow the dynamic processes).	• Increase the gain of the torque controller (C00073/2). • Reduce the reset time of the torque controller (C00074/2). • Increase the acceleration (C00012)/deceleration (C00013) time.
Mechanical resonance at certain speeds.	The L_NSet_1 function block masks out those speed ranges that include resonance.
Speed variations in no-load operation for speeds > 1/3 rated speed.	Minimise speed oscillations with oscillation damping (C00234).
Drive runs unstable.	Check set motor data (nameplate data and equivalent circuit diagram data). ▶ Motor selection/Motor data (📖 144)
Setpoint speed and actual speed differ strongly.	
The torque required is not generated at standstill.	Increase motor magnetising current (C00095).
Current overshoots occur when heavy loads are accelerated from standstill (OC1 or OC11 error).	
The machine runs uneven.	

5.8 Sensorless control for synchronous motors (SLPSM)

The sensorless control for synchronous motors is based on a decoupled and separated control of the torque-creating and field-creating current share of synchronous motors. In contrast to the servo control, the actual speed value and the rotor position are reconstructed via a motor model.

Stop!

- When the maximum output frequency is limited to ± 599 Hz, the devices are not subject to the export restrictions of the "EC-Dual-Use Regulation" - EC 428/2009. This applies to the devices supplied from the middle of the year 2015. The maximum possible output frequency is given on the nameplate.
  - For certain applications, the devices can be supplied with the previous maximum output frequency of ± 1000 Hz. If required, get in touch with your Lenze contact person.
- The sensorless control for synchronous motors is only possible up to a maximum output frequency of 650 Hz (from version 14.00.00: 1000 Hz)!
  - Depending on the number of motor pole pairs, the reference speed (C00011) may only be selected that high that the output frequency displayed in C00059 is less than or equal to 650 Hz (from version 14.00.00: 1000 Hz).
- We recommend to select a power-adapted combination of inverter and motor.
- The Lenze setting permits the operation of a power-adapted motor. Optimal operation is only possible if either:
  - the motor is selected via the »Engineer« motor catalogue,
  - the motor nameplate data are entered and motor parameter identification is carried out afterwards
    - or -
  - the nameplate data and equivalent circuit data of the motor (motor leakage inductance and motor stator resistance) are entered manually.
- When you enter the motor nameplate data, take into account the phase connection implemented for the motor (star or delta connection). Only enter the data applying to the selected connection type.
- In order to protect the motor (e.g. from demagnetisation) we recommend setting the ultimate motor current in C00939. This ensures motor protection even with an unstable operation. ▶ Maximum current monitoring (📖 320)
- Controller enable is only possible if the motor is at standstill.
  - When the controller is enabled, a jerk may occur due to an angle jump since the rotor displacement angle is not known after controller enable. For some applications, this jerk in the machine is not acceptable.
  - From version 02.00.00, the rotor displacement angle is identified with every controller enable in the Lenze setting, and thus a jerk in the machine after controller enable can be avoided. ▶ Pole position identification without motion (📖 385)
  - A flying restart circuit for synchronising to rotating motors is in preparation.
- The injection of a constant current may cause an unwanted heating of the motor at controlled operation.
  - We recommend the use of a temperature feedback via PT1000 or KTY, PTC or thermal contact.
    ▶ Motor temperature monitoring (PT1000 or KTY) (📖 395)
    ▶ Motor temperature monitoring (PTC) (📖 313)

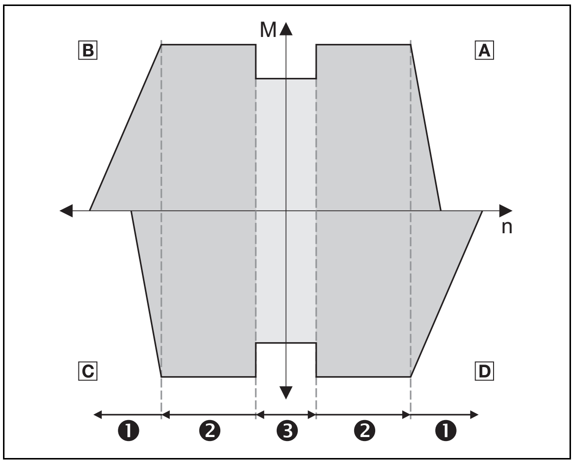
Note!

Currently, the sensorless control does not contain a flying restart function that enables a synchronisation of the inverter to a rotating machine.

- Thus, we recommend taking measures for preventing overvoltages at operation in generator mode (e.g. brake resistor).
- By any means, the delay time for the "DC-bus overvoltage" error trigger in C00601/1 must be set to 0 s.

The motor model-based speed monitoring requires a rotating machine. Thus, the operational performance of the sensorless control for synchronous motors is divided into two categories:

1. Open-loop controlled operation ($|n_{setpoint}| < n_{C00996}$)
   - In the range of low speeds, the speed of a synchronous motor is not possible. Thus, only an adjustable and constant current is injected that enables an acceleration.
2. Closed-loop controlled operation ($|n_{setpoint}| > n_{C00996}$)
   - In this range, the rotor flux position and the speed are reconstructed via an observer. The control is carried out field-oriented. Only the current is injected that is needed for the required torque.



A Operation in motor mode (CW rotation)
B Operation in generator mode (CCW rotation)
C Operation in motor mode (CCW rotation)
D Operation in generator mode (CW rotation)

❶ Field weakening (in preparation)
❷ Closed-loop controlled operation
❸ Open-loop controlled operation

[5-13] Operating ranges of the sensorless control for synchronous motors

The sensorless control for synchronous motors has similar advantages for the closed-loop controlled operating range and the servo control (SC) for synchronous motors. Compared to asynchronous motors, there are the following advantages:

- Higher power density of the motor
- Higher efficiency
- Limitation of the maximum torque in motor mode and generator mode in closed-loop operating range
- Implementation of simple positioning

Note!

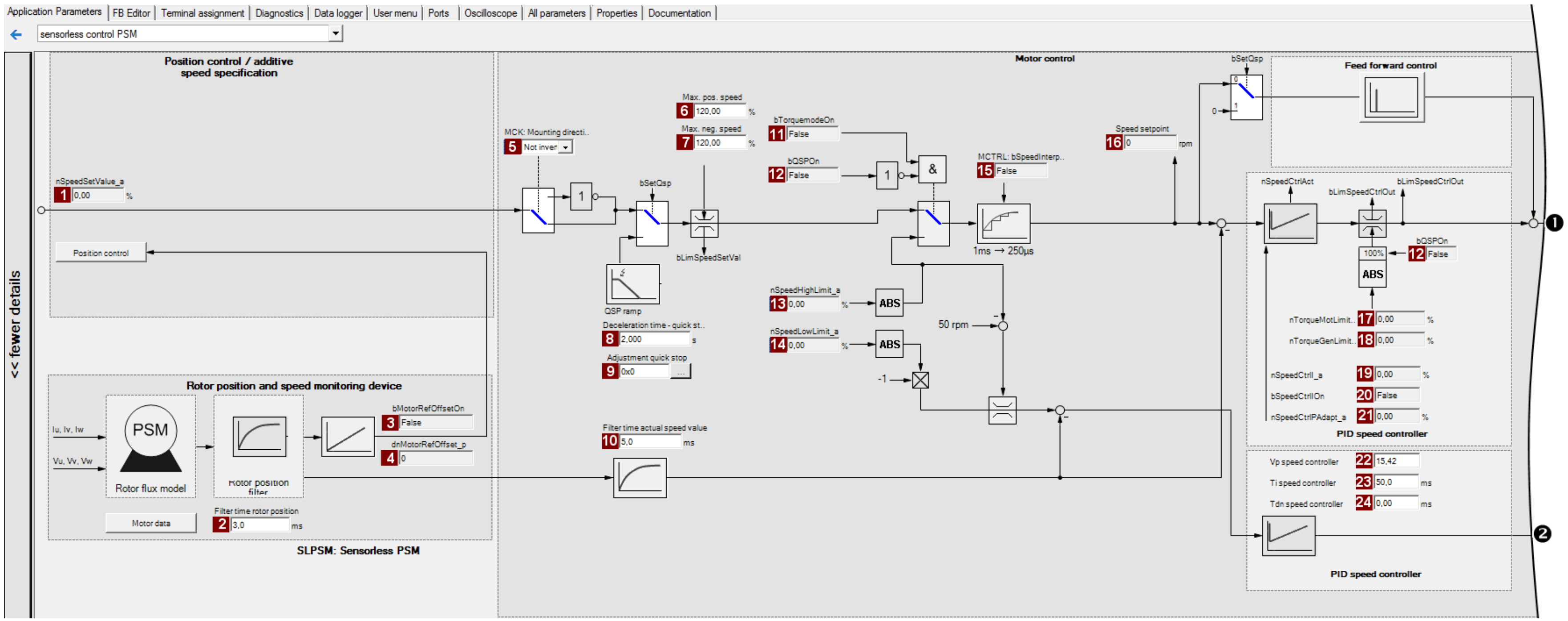
In the open-loop controlled operation for synchronous motors, usually less maximum torque is created than in the closed-loop controlled operation. Thus, the dynamics of this control is limited and a high acceleration of the drive is not possible in this control mode.

From version V17.00.00onwards, the *bSlpsmSpeedopenLoopControl*process output is available at SB LS_DeviceMonitor(open-loop controlled operation of the SLPM is active). This signal can be connected to the additional ramp *L_Nset.bTix*to realise an own ramp in the open-loop controlled operation. This serves to set a flat ramp for the open-loop controlled operation and a steep ramp for the closed-loop controlled operation and thus achieve a considerably higher acceleration of the entire drive. Moreover, less vibrations are caused.

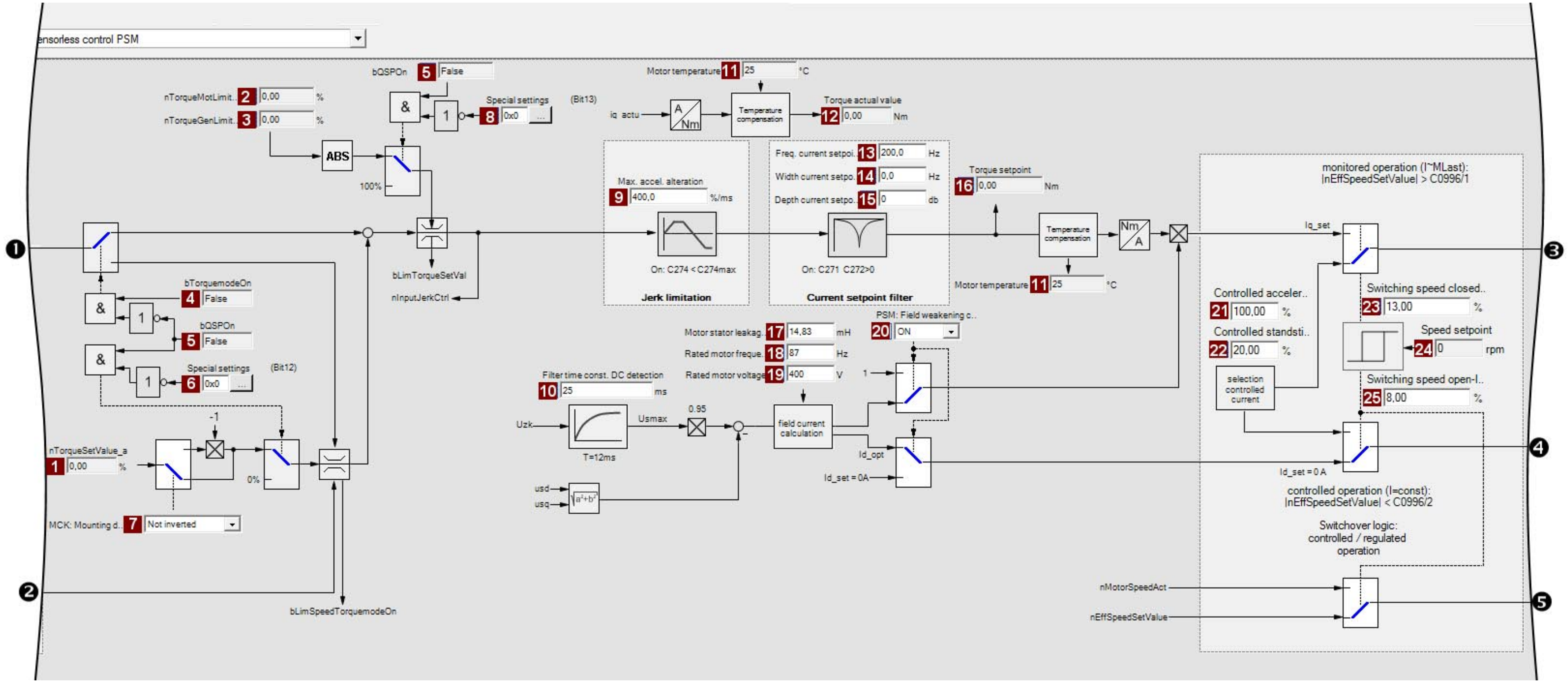
5.8.1 Parameterisation dialog/signal flow

Proceed as follows to open the dialog for parameterising the motor control:

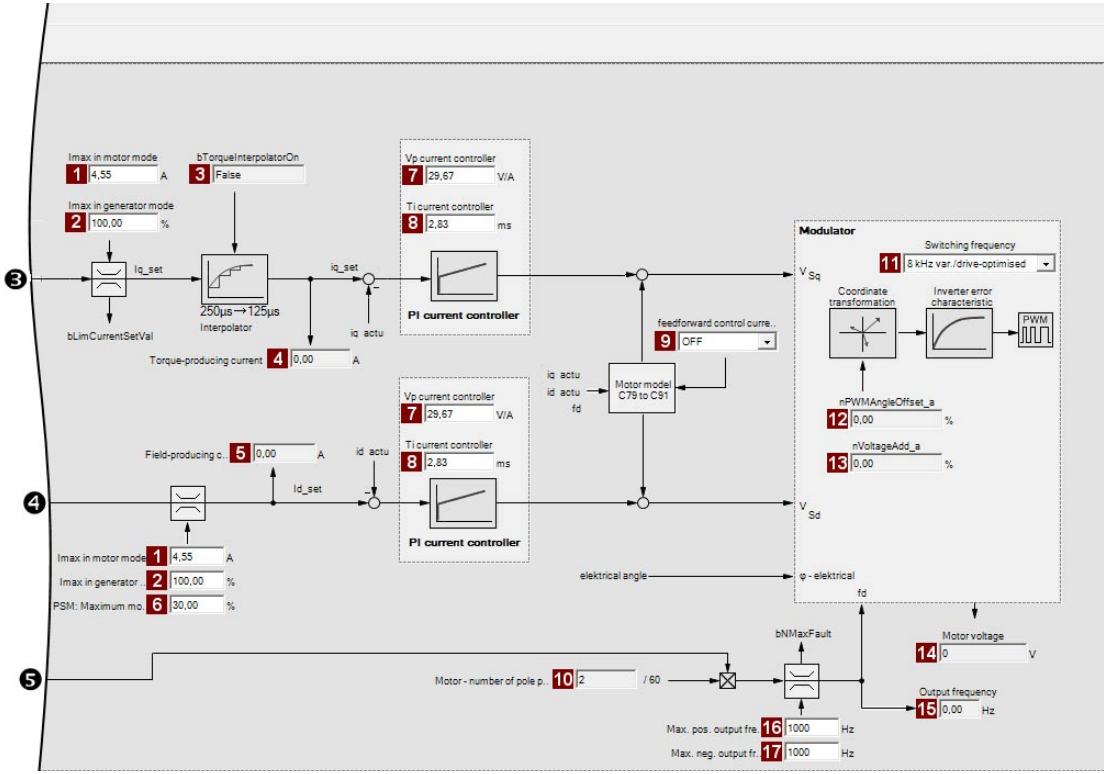
1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Select the motor control "3: SLPSM: Sensorless PSM" from the *Overview* dialog level in the **Motor control** list field:
4. Click the **Motor control servo SLPSM** button to change to the *Overview* → *Motor control vector* dialog box.
   - This dialog level only shows a simplified signal flow with the most important parameters.
   - When you click the **>>More details** button in the left-most position, a signal flow with more details/parameters is displayed.



	Parameters	Info		Parameters	Info		Parameters	Info
1	C00830/22	Speed setpoint	6	C00909/1	Max. pos. speed	13	C00830/88	MCTRL: nSpeedHighLimit_a
2	C00998/1	SLPSM: Filter time rotor position	7	C00909/2	Max. neg. speed	14	C00830/23	MCTRL: nSpeedLowLimit_a
3	C00833/68	MCTRL: bMotorRefOffsetOn	8	C00105	Decel. time - quick stop	15	C00833/28	MCTRL: bSpeedInterp
4	C00834/6	MCTRL: dnMotorRefOffset_p	9	C00104/1	Quick stop setting	16	C00050	Speed setpoint
5	C01206/1	MCK: Mounting direction: Motor	10	C00998/2	SLPSM: Filter time actual speed value	17	C00830/29	Limitation of torque in motor mode
			11	C00833/30	Actual speed value	18	C00830/28	Limitation of torque in generator mode
			12	C00833/33	bQSPOn	19	C00830/24	MCTRL: nSpeedCtrlI_a
						20	C00833/31	MCTRL: bSpeedCtrlIOn
						21	C00830/25	MCTRL: nSpeedCtrlPAdapt_a
						22	C00070/2	SLPSM: Vp speed controller
						23	C00071/2	SLPSM: Ti speed controller
						24	C00072	SC: Tdn speed controller



Parameters		Info	Parameters		Info	Parameters		Info
1	C00830/27	MCTRL: nTorqueSetValue_a	11	C00063/1	Motor temperature	21	C00995/1	SLPSM: Open-loop controlled accelerating current
2	C00830/29	Limitation of torque in motor mode	12	C00056/2	Actual torque	22	C00995/2	SLPSM: Open-loop controlled standstill current
3	C00830/28	Limitation of torque in generator mode	13	C00270	SC: Freq. current setpoint filter	23	C00996/1	SLPSM: Closed-loop controlled switching speed
4	C00833/30	bTorquemodeOn	14	C00271	SC: Current setpoint filter width	24	C00050	Speed setpoint
5	C00833/33	bQSPOn	15	C00272	SC: Current setpoint filter depth	25	C00996/2	SLPSM: Open-loop controlled switching speed
6	C02865/1	Special settings (bit 12)	16	C00056/1	Torque setpoint			
7	C01206/1	MCK: Mounting direction	17	C00085	Motor stator leakage inductance			
8	C02865/1	Sondereinstellungen (Bit 13)	18	C00089	Rated motor frequency			
9	C00274	SC: Max. change in acceleration	19	C00090	Rated motor voltage			
10	C00280	SC: Filter time const. DC detection	20	C00079/4	Field weakening			



	Parameters	Info
1	C00022	Imax in motor mode
2	C00023	Imax in generator mode
3	C00833/29	MCTRL: bTorqueInterpolatorOn
4	C00937/2	Torque-producing current
5	C00937/1	Maximally effective field-producing motor current
6	C00938	Limitation of the field-producing motor current
7	C00075	Vp current controller
8	C00076	Ti current controller
9	C00079/1	SC: Current controller - feedforward control
10	C00969/1	Motor - number of pole pairs
11	C00018	Switching frequency
12	C00830/32	MCTRL: nPWMAngleOffset_a
13	C00830/31	MCTRL: nVoltageAdd_a
14	C00052	Motor voltage
15	C00058	Output frequency
16	C00910/1	Max. pos. output frequency
17	C00910/2	Max. neg. output frequency

5.8.2 Types of control

Sensorless control for synchronous motors can only be executed in the "Speed control with torque limitation" (*bTorquemodeOn* = FALSE) mode.

Speed control with torque limitation

A speed setpoint is selected and the drive system is operated in a speed-controlled manner. For adapting the operational performance, the overload in the drive train can be limited:

- The torque is limited via the torque setpoint.
- The torque setpoint is identical to the value at the output of the speed controller, *nOutputSpeedCtrl*.
- To avoid overload in the drive train, the torque in motor mode can be limited via the *nTorqueMotLimit_a* process input signal, and the torque in generator mode can be limited via the *nTorqueGenLimit_a* process input signal:

Designator DIS code | data type	**Information/possible settings**
nTorqueMotLimit_a C00830/29 | INT	Torque limitation in motor mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: 0 ... +199.99 % • If keypad control is performed: Parameterisable via C00728/1. From version 18.00.00 onwards: C02864: Bit 15 = 1: positive torque limitation (nTorqueHighLimit_a)
nTorqueGenLimit_a C00830/28 | INT	Torque limitation in generator mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: -199.99 ... 0 % • If keypad control is performed: Parameterisable via C00728/2. From version 18.00.00 onwards: C02864: Bit 15 = 1: negative torque limitation (nTorqueLowLimit_a)

STOP

Stop!

Torque limitation is only active in the closed-loop controlled operation ($|n_{Setpoint}| > n_{C00996}$)!

- It must be prevented that the actual speed value is braked into the non-observable area due to the torque limitation!

Note!

To avoid instabilities during operation, the torque limit values are internally processed as absolute values.

5.8.3 Basic settings

The following "Initial commissioning steps" must be performed to commission the sensorless control for synchronous motors:

<table>
<tr><th colspan="3">Initial commissioning steps</th></tr>
<tr><td>1</td><td colspan="2">Select motor control:
C00006 = "3: SLPSM: Sensorless PSM"</td></tr>
<tr><td rowspan="2">2.</td><td colspan="2">Set the motor selection/motor data
• When selecting and parameterising the motor, the motor nameplate data and the equivalent circuit diagram data are relevant. Detailed information can be found in the "Motor selection/Motor data" chapter. (📖 144)
Depending on the motor manufacturer, proceed as follows:</td></tr>
<tr><td>Lenze motor:
Selecting a motor from the motor catalogue in the »Engineer«
- or -
1. Set the motor nameplate data
2. Automatic motor data identification</td><td>Third party manufacturer's motor:
1. Set the motor nameplate data
2. Automatic motor data identification or set known equivalent circuit diagram manually:
C00084: Motor stator resistance
C00085: Motor stator leakage inductance</td></tr>
<tr><td>3.</td><td colspan="2">Set speed switching thresholds between open-loop and closed-loop controlled operation:
• Set transition speed from closed-loop to open-loop operation in C00996/1 in [%] with regard to the rated motor speed (C00087).
• Set transition speed from closed-loop to open-loop operation in C00996/2 in [%] with regard to the rated motor speed (C00087).
Tip!
• With voltage-adjusted motors, a speed switching threshold of 10 % is recommended.
• As a rule of thumb, the speed switching threshold should be selected as follows:
$$C00996/1...2\,[\%] = \frac{U_{Rated,\,motor}[V]}{U_{Rated,\,FI}[V]} \cdot 10$$</td></tr>
<tr><td>4.</td><td colspan="2">Set open-loop accelerating current in C00995/1 in [%] with regard to the rated motor current (C00088).
• This value defines the height of the current that is injected during the acceleration process.
• The accelerating current must be dimensioned so that the required torque in the lower speed range can always be reached (acceleration torque + load torque):
$$C00995/1\,[\%] = \frac{M_{Meax}[Nm]}{M_{Rated}[Nm]} \cdot I_{Rated,\,motor}[A] \cdot 1.3$$</td></tr>
<tr><td>5th</td><td colspan="2">Set open-loop steady-state current in C00995/2 in [%] with regard to the rated motor current (C00088).
• This value defines the height of the current for processes without acceleration (e.g. standstill or constant setpoint speed).</td></tr>
<tr><td>6.</td><td colspan="2">For improving the operating characteristics:
If required, adapt the filter time for reconstructing the rotor position and the actual speed value through the motor model in C00998/1 and C00998/2.
• We recommend using the Lenze setting:
Filter time rotor position (C00998/1) = 3 ms
Filter time actual speed value (C00998/2) = 5 ms
• Deviant from this, the following value range can be used:
Filter time rotor position (C00998/1) = 2 ... 5 ms
Filter time actual speed value (C00998/2) = 3 ... 8 ms</td></tr>
<tr><td>6.</td><td colspan="2">For protecting the motor from demagnetisation:
Set the ultimate current in C00939.</td></tr>
</table>

Note!

The Lenze settings of the current controller are predefined for a power-adapted motor. For an optimal drive behaviour of a synchronous motor, we recommend to adapt the controller settings.

Tip!

Information on the optimisation of the control mode and the adaptation to the real application is provided in the "Optimising the control mode" chapter.

Parameterisable additional functions are described correspondingly in the chapter "Parameterisable additional functions". (272)

5.8.3.1 Reduction of speed overshoot

During the transition from the controlled to the speed-monitored range, the I component of the speed controller will be preloaded with the maximum possible torque of the controlled range. The determining variable for this maximum torque is the controlled accelerating current (C00995/1). The Lenze setting (C00995/1 = 100 %) corresponds to the maximum torque.

In the event that less torque is actually required from the motor, a brief speed overshoot occurs during the transition from the controlled range to the speed-monitored range.

This speed overshoot is strongly noticeable in synchronous machines with very low power (e.g. type **MCS06C41** with P_N = 250 W).

For versions < 21.00.00, the speed overshoot can be reduced by decreasing the controlled accelerating current (C00995/1). Disadvantage of this measure: the maximum possible motor torque is reduced.

This function expansion is available from version 21.00.00.

If necessary, an adjustment for loading the I component of the speed controller during transition from the controlled range to the speed-monitored range can be performed to reduce this speed overshoot. The maximum possible torque is still available in this case because the controlled accelerating current continues to have an effect.

The adjustment can be performed with

C00936/1 = 0 ... 200 % (SLPSM: load value of the speed controller)

Recommendations

- Synchronous machines with low power
  Setting for the load value of the speed controller for synchronous machines with low power (e.g. type **MCS06C41** with P_N = 250 W):
  C00936/1 = 50%.
- Drives with high starting torque:
  Setting for the load value of the speed controller when a very high starting torque is required from the drive:
  C00936/1 = 101% ... 200%
  The load value of the speed controller is 100%.

5.8.4 Optimising the control mode

Note!

From version 12.00.00:

- Following successful motor parameter identification, the current controller parameters (C00075, C00076) are calculated automatically.
  - If these parameters are not to be calculated, bit 4 of C02865/1 must be set to "1".
- Following successful motor parameter identification, the speed controller parameters (C00070/3, C00071/3) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "5".
- Following successful motor parameter identification, other controller parameters (C00011, C00022) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "6".

The "optimisation steps" given in the table below serve to further optimise the control behaviour of the sensorless control for synchronous motors and adjust it to the concrete application.

- Detailed information on the individual steps can be found in the following subchapters.

Generally, the following optimisation steps are recommended:

Optimisation steps	
1	Optimise current controller. (📖 232) • The current controller should always be optimised if a motor of a third-party manufacturer with unknown motor data is used!
2.	Optimise speed controller. (📖 233) • The setting of the speed controller must be adapted depending on the mechanical path.
3.	Optimise response to setpoint changes and determine mass inertia. (📖 236) • For an optimal reference behaviour, the total moment of inertia can be used to make a feedforward control of the speed setpoint.

Note!

Current setpoint filter (band-stop filter) / jerk limitation

The use of the functions is only recommended in exceptional cases.

The functions are described in chapter "Servo control (SC)":

- Setting the current setpoint filter (band-stop filter) (📖 259)
- Adapting the max. acceleration change (jerk limitation) (📖 260)

5.8.4.1 Optimise current controller

Note!

An optimisation of the current controller should generally be carried out unless a power-adapted standard motor is used or the motor has been selected from the motor catalogue of the »Engineer«!

An optimisation of the current controller is sensible since the two control parameters gain (C00075) and reset time (C00076) depend on the required maximum current and the set switching frequency.

Parameters	Info	Lenze setting	
		Value	Unit
C00075	Vp current controller	7.00	V/A
C00076	Ti current controller	10.61	ms

- Gain and reset time can be calculated as per the following formulae:

$$V_p = \frac{L_{ss}[H]}{T_E[s]}$$

$$T_i = \frac{L_{ss}[H]}{R_s[\Omega]}$$

V_p = Current controller gain (C00075)
T_i = Current controller reset time (C00076)
L_{ss} = Motor stator leakage inductance (C00085)
R_s = Motor stator resistance (C00084)
T_E = Equivalent time constant (= 500 µs)

5.8.4.2 Optimise speed controller

The speed controller is in the form of a PID controller with an additional differential speed-setpoint gain. For optimum behaviour, the PID speed controller has to be optimised and the overall mass inertia of the drive train has to be determined.

- In the Lenze setting, the configuration of the speed controller provides robustness and moderate dynamics.

Parameters	Info	Lenze setting	
		Value	Unit
C00070/3	SLPSM: Vp speed controller	3.00	
C00071/3	SLPSM: Ti speed controller	100.0	ms
C00072	SC: Tdn speed controller	0.00	ms

Speed controller gain Vp

The gain Vp (C00070/3) of the speed controller is defined in a scaled representation which enables a comparable parameterisation almost independent of the power of the motor or inverter. Here, the speed input difference of the controller is scaled to the rated motor speed whereas the output torque refers to the rated motor torque. A gain of 10 means that a speed difference of 1 % is gained through the P component with 10 % torque.

If the rated data of the motor and the mass inertia of the drive system are known, we recommend the following setting:

$$V_P \approx 0.2 \ldots 0.5 \cdot \frac{T_M[s]}{0.01[s]}$$

$$T_M[s] = \frac{2 \cdot \pi \cdot n_N[rpm]}{M_N[Nm] \cdot 60} \cdot J_{Drive,\ total}[kgm^2]$$

$$M_N[Nm] = \frac{P_N[W] \cdot 60}{2 \cdot \pi \cdot n_N[rpm]}$$

V_P = Gain of the speed controller (C00070/3)
T_M = Time constant for the acceleration of the motor
M_N = Rated motor torque
n_N = Rated motor speed
$J_{drive,\ total}$ = Total moment of inertia of the drive

[5-14] Recommendation for the setting of the gain of the speed controller

If the mass inertia of the drive is unknown, the optimisation can be achieved as follows:

1. Specify speed setpoint.
   - A small speed just above the switching threshold is recommended in the closed-loop controlled operation.
2. Increase Vp (C00070/3) until the drive starts to oscillate (observe engine noise).
3. Reduce Vp (C00070/3) until the drive runs stable again.
4. Reduce Vp (C00070/3) to approx. half the value.
5. Afterwards check results of the optimisation in the entire speed range (one-time passing through of the speed range).

Tip!

Values recommended by Lenze for the setting of the (proportional) gain:
- For drive systems without feedback: Vp = 2 ... 8
- For drive systems with a good disturbance behaviour: Vp > 6

Speed controller reset time Ti

Apart from setting the P component, C00071/3 provides the possibility to take influence on the I component of the PI controller.

If the mass inertia of the drive is unknown, the optimisation can be achieved as follows:

1. Specify speed setpoint.
2. Reduce Ti (C00071/3) until the drive starts to oscillate (observe engine noise).
3. Increase Ti (C00071/3) until the drive runs stable again.
4. Increase Ti (C00071/3) to approx. twice the value.

Tip!

Value range recommended by Lenze for the setting of the reset time:

Ti = 20 ms ... 150 ms

Using the ramp response for setting the speed controller

If the mechanical components cannot be operated at the stability limit, the ramp response can also be used for setting the speed controller.

Stop!

If the controller parameters are preset unfavourably, the control can tend to heavy overshoots up to instability!
- Following and speed errors can adopt very high values.
- If the mechanics are sensitive, the corresponding monitoring functions are to be activated.

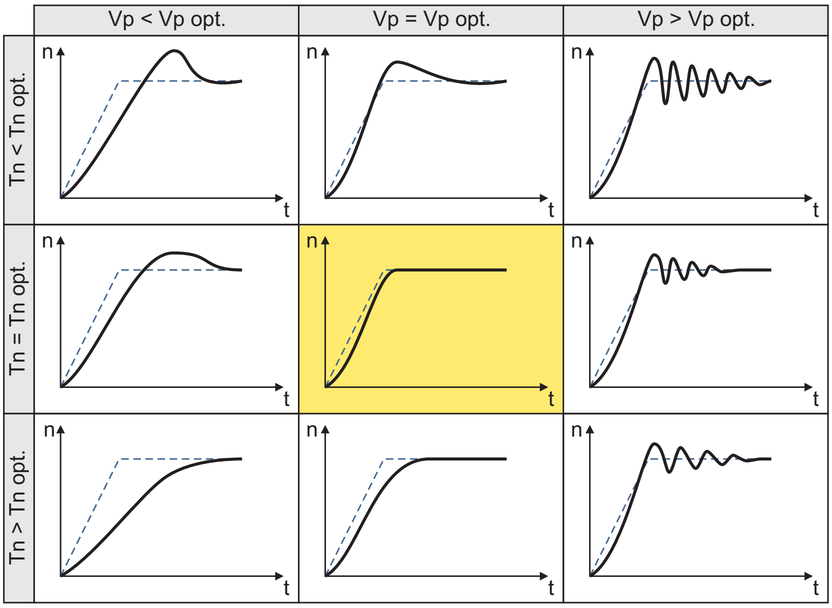
Note!

For an optimal setting, we recommend to determine the mass inertia (optimal response to setpoint changes) first.

▶ Optimise response to setpoint changes and determine mass inertia (📖 236)

How to optimise the speed controller setting by means of the ramp response:

1. Run a typical speed profile and record the ramp response of the speed using the data logger.
   - Motor control variables to be recorded:
     nSpeedSetValue_a (speed setpoint)
     nMotorSpeedAct_a (actual speed value)
2. Evaluate the ramp response:

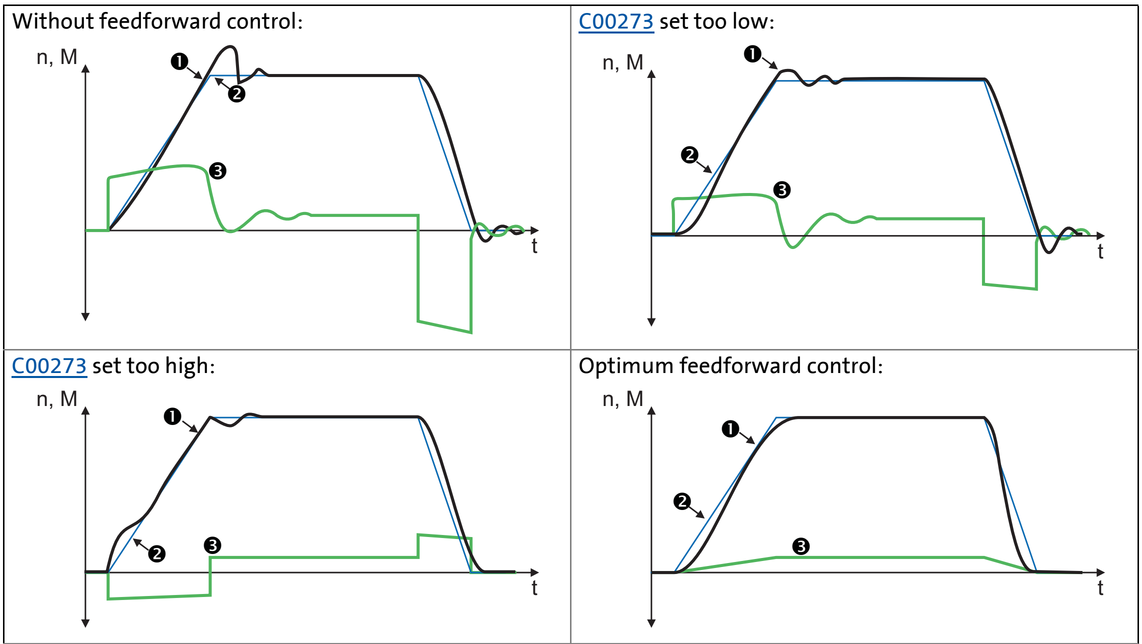


   - Solid line = ramp response (actual speed value)
   - Dash line = speed setpoint
3. Change gain Vp in C00070/3 and reset time Tn in C00071/3.
4. Repeat steps 1 ... 3 until the optimum ramp response is reached.

5.8.4.3 Optimise response to setpoint changes and determine mass inertia

Optimisation at constant mass inertia

Setting the total moment of inertia under C00273 provides the optimum torque feedforward control. Depending on the application, an adjustment of the setting under C00273 may be necessary to optimise the response to position/speed setpoint changes by means of the torque feedforward control.



❶ Actual speed value (signal *nMotorSpeedAct_a*)
❷ Speed setpoint (*nSpeedSetValue_a* signal)
❸ Speed controller output (signal *nOutputSpeedCtrl_a*)

[5-15] Typical signal characteristics for different settings of the load moment of inertia

How to optimise the torque feedforward control:

1. Run a typical speed profile and record the inputs and outputs of the speed controller with the data logger.
   - Motor control variables to be recorded:
     nSpeedSetValue_a (speed setpoint)
     nMotorSpeedAct_a (actual speed value)
     nOutputSpeedCtrl_a (speed controller output)
2. Estimate the moment of inertia and set it in C00273 in relation to the motor end (i.e. with account being taken of the gearbox factors).
3. Repeat the data logger recording (see step 1).

   Now the data logger should show that part of the required torque is generated by the feedforward control and the speed controller output signal (*nOutputSpeedCtrl_a*) is correspondingly smaller. The resulting following error decreases.
4. Change the setting in C00273 and repeat the data logger recording until the intended response to setpoint changes is reached.
   - The optimisation could aim at the speed controller being completely relieved (see signal characteristics in Fig. [5-15]).
5. Save the parameter set (device command: C00002/11).

Optimisation at variable mass inertia

From version V12.00.00, mass inertia that changes during the process (e.g. a reel) can be taken into account when optimising the response to setpoint changes.

How to proceed:

1. In C00273 the known constant total moment of inertia (motor, gearbox, shaft, etc.) must be set or determined according to previous instructions ("How to optimise ...").
   - The determination requires travelling the typical speed profile without variable mass inertia (e.g. reels).
2. At the LS_MotorInterface SB, the *nInertiaAdapt_a* process signal must be interconnected in a way which ensures that a value of "100 %" is applied at this input.
3. In C00919/1 set the known maximum value of the variable moment of inertia or determine the value according to previous instructions ("How to optimise ...").
   - The determination requires travelling the typical speed profile including variable mass inertia (e.g. maximum reels).
4. The *nInertiaAdapt_a* process signal can be used during the process to dynamically control the percentage of the variable moment of inertia set in C00919/1 which is to be considered for setpoint feedforward control.
   Example:
   - If there is no variable moment of inertia (e.g. no reel), the *nInertiaAdapt_a* process signal must be set to "0 %".
   - If the maximally variable moment of inertia is available (e.g. maximum reels), the *nInertiaAdapt_a* process signal must be set to "100 %".

Tip!

Via the *nTorqueSetValue_a* process signal at the LS_MotorInterface SB, any differential signal can be defined for torque feedforward control. First the speed controller, then this additive torque is connected which is hence not derived from the differential change in speed setpoint.

Other functions for differential setpoint feedforward control

From version V12.00.00 onwards, the following additional functions are available for differential setpoint feedforward control (torque feedforward control):

- In C00653/1, the sensitivity of setpoint feedforward control can be adapted.
- In C00654/1, alternatively to the *nSpeedSetValue_a* process signal, the new *nSpeedSetValueInertia_a* process signal for the setpoint feedforward control can be selected at the LS_MotorInterface SB. Via the *nSpeedSetValueInertia_a* process signal, an optional input value (e.g. setpoint of the position or process controller) for the torque feedforward control can be specified.
- For *bTorqueModeOn* = TRUE, the setpoint feedforward control is added to the torque setpoint *nTorqueSetValue_a*. In this way, feedforward control of torque is also possible for torque-controlled operation (e.g. for winder applications).

From version V18.00.00 onwards, the following additional functions are available for differential setpoint feedforward control (torque feedforward control):

- In C00654/1, an exact torque feedforward control can be set for the nSpeedSetValue_a speed setpoint. In case of a very dynamic application, set C00654/1 = 2.
- From version V15.00.00 onwards, the torque feedforward control behaves differently than in version V14.00.00 or older. In order to achieve the same behaviour as in version V14.00.00 or older, set C00654/1 = 3.

STOP

Stop!

In the Lenze setting of C00654/1, the speed setpoint of the speed controller (*nSpeedSetValue_a*) is used for the torque feedforward control which is why it is also called "differential setpoint feedforward control". A very sharp change of the speed setpoint at the speed controller thus causes a strong torque impulse at the machine!

Causes for a very sharp change of the speed setpoint of the speed controller:

- A control creates the setpoint ramp itself and the speed setpoint is only written every 20 ms to the drive. (In this case, the speed setpoint is changed every 20 ms.)
  - Recommendation: Deactivate the torque feedforward control for the speed setpoint in C00654/1 if the speed setpoint changes very sharply or the mass inertia is unknown (e.g. in case of hoists)!
- The reference speed C0011 is much higher than the rated motor speed set in C0087.
  - Recommendation: Deactivate the torque feedforward control for the speed setpoint in C00654/1 if the reference speed C0011 is 5 times higher than the rated motor speed C0087.

5.9 Servo control (SC)

Field-oriented servo control (SC) is based on a decoupled, separate control of the torque-producing and the field-producing current component. The motor control is based on a field-oriented, cascaded controller structure with feedback function and enables dynamic and stable operation in all of the four quadrants. It can be used for synchronous motors (PSM) and asynchronous motors (ASM).

Stop!

- Servo control requires a speed feedback!
  - The speed sensor used has to be set in C00495. This setting is not made automatically with the selection of the motor from the »Engineer« motor catalogue!
  - If no speed sensor is set in C00495 and the controller is enabled, an impermissibly high motor current occurs which may destroy the motor thermally!
  - From version 14.00.00, the error response set in C00571/2 (Lenze setting: "Fault") takes place if in case of controller enable it is detected that a motor control type with feedback is set in C00006 but no speed encoder is set in C00495.
- We recommend to select a power-adapted combination of inverter and motor.
- The Lenze setting permits the operation of a power-adapted motor. Optimal operation is only possible if either:
  - the motor is selected via the »Engineer« motor catalogue,
  - the motor nameplate data are entered and motor parameter identification is carried out afterwards
    - or -
  - the nameplate data and equivalent circuit data of the motor are entered manually.
- When you enter the motor nameplate data, take into account the phase connection implemented for the motor (star or delta connection). Only enter the data applying to the selected connection type.

Stop!

V/f emergency operation

From version 15.00.00 , it is internally switched to the encoderless V/f characteristic control in case of an encoder open circuit in order to avoid impermissible motor movements.

- In order that this "V/f emergency operation" works properly, the parameters relevant for the V/f characteristic control (base frequency, Vmin boost, slip compensation, etc.) have to be set correctly. As an alternative, a motor parameter identification can be executed as well.
- The change-over to "V/f emergency operation" is reported via bit 4 in C01000 and via the *bWirebreakUfLinearActive* status signal at the SB LS_DeviceMonitor.
- The change-over to "V/f emergency operation" can be suppressed by setting bit 8 to "1" in C02864/1.
- In case of servo control for synchronous motors (PSM), no change-over to "F/f emergency operation" takes place.

Note!

- For closed-loop control of a synchronous motor (PSM), the pole position of the motor must be known! ▶ Pole position identification (PPI) (📖 379)
- For closed-loop control of an asynchronous motor (ASM), the maximum current (C00022) should be higher than the magnetising current as otherwise the motor does not generate a torque.

Detailed information on the speed feedback can be found in the chapter "Encoder/feedback system". (📖 330)

Generally, the servo control offers the same advantages as the sensorless vector control (SLVC), i.e. compared to the V/f characteristic control, the servo control (SC) can be used to achieve

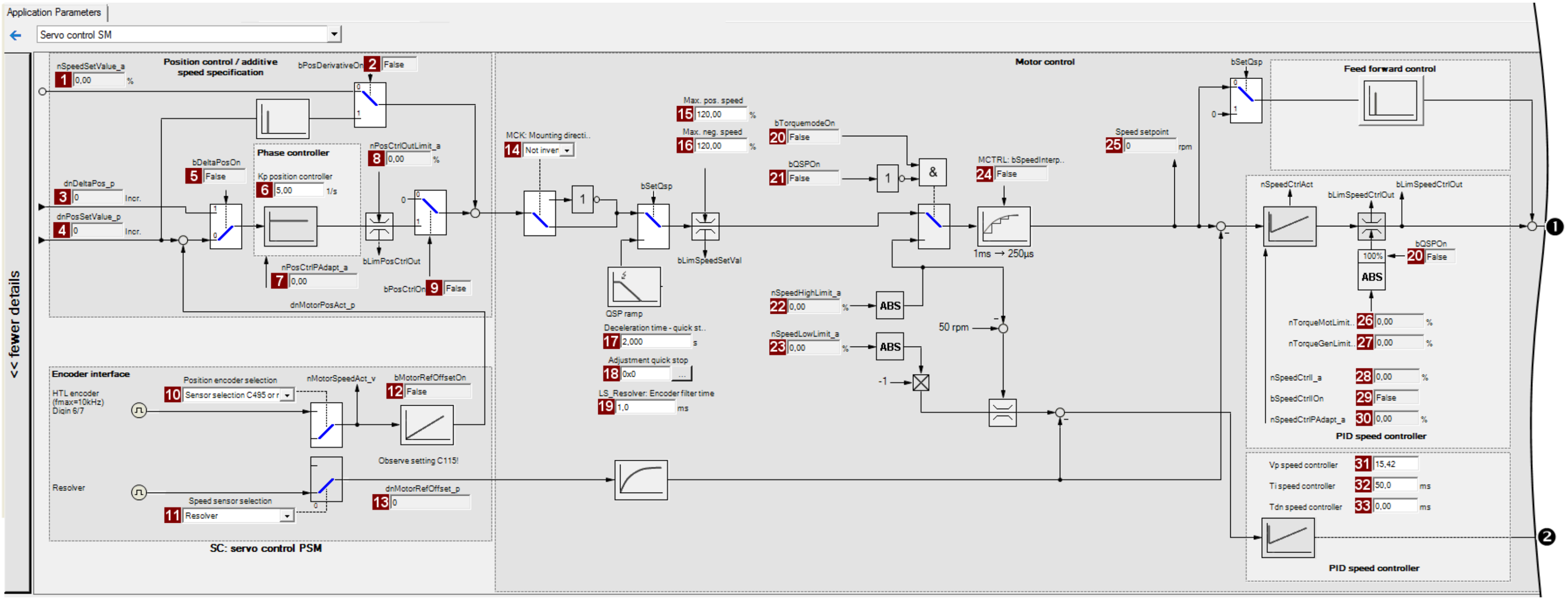
- A higher maximum torque throughout the entire speed range
- A higher speed accuracy
- A higher concentricity factor
- A higher level of efficiency
- The implementation of torque-controlled operation with speed limitation
- The limitation of the maximum torque in motor and generator mode for speed-controlled operation

5.9.1 Parameterisation dialog/signal flow

Proceed as follows to open the dialog for parameterising the motor control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Select the motor control from the *Overview* dialog level in the **Motor control** (C00006) list field:
   - "1: SC: Servo control PSM" for synchronous motor
     - or -
   - "2: SC: Servo control ASM" for asynchronous motor
4. Click the **Motor control servo** button to change to the *Overview* ➜ *Motor control vector* dialog box.
   - This dialog level only shows a simplified signal flow with the most important parameters.
   - When you click the **>>More details** button in the left-most position, a signal flow with more details/parameters is displayed, as shown in the following subchapter.

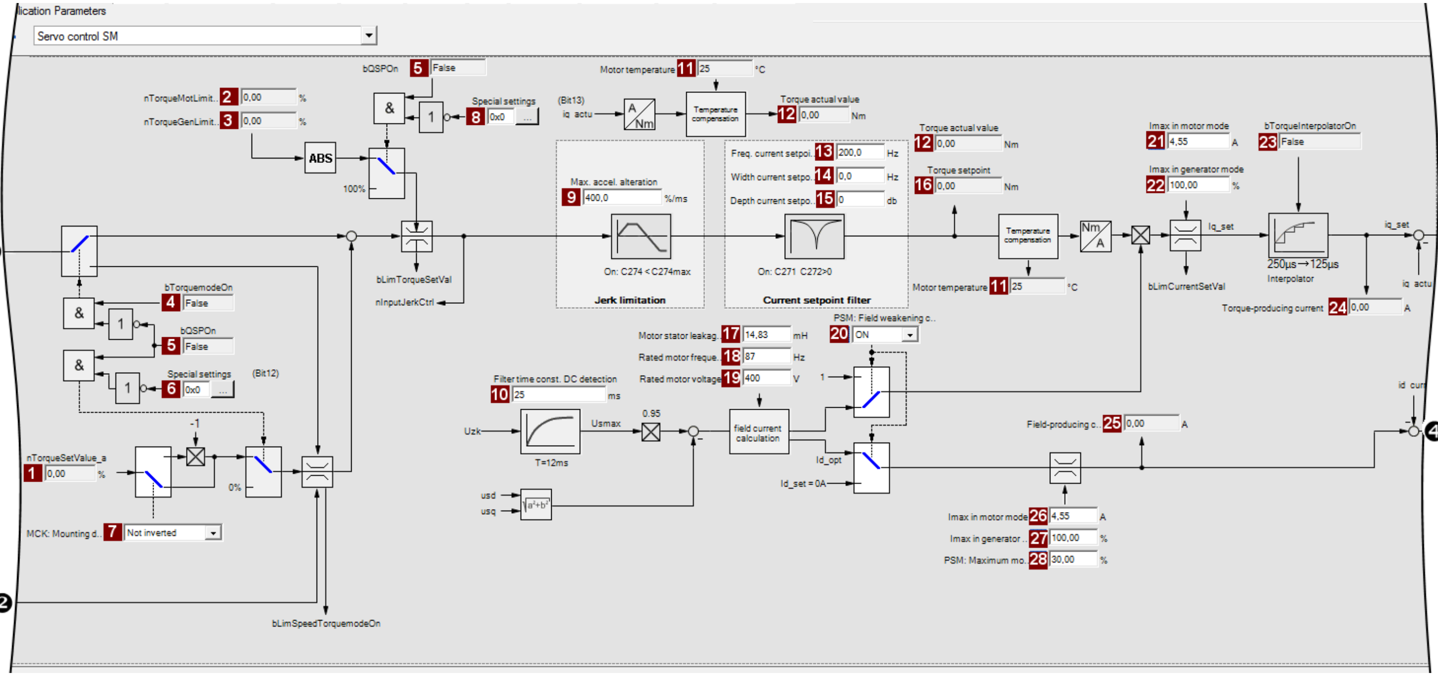
Signal flow for servo control for synchronous motor (PSM):



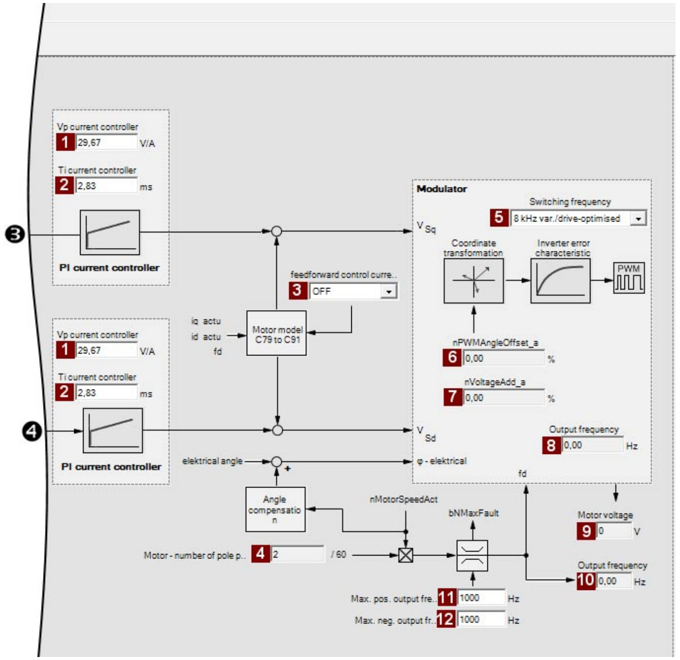
Parameters		Info
1	C00830/22	Speed setpoint
2	C00833/67	MCTRL: bPosDerivativeOn
3	C00834/4	MCTRL: dnDeltaPos_p
4	C00834/5	MCTRL: dnPosSetValue_p
5	C00833/35	MCTRL: bDeltaPosOn
6	C00254	Kp position controller
7	C00830/20	MCTRL: nPosCtrlPAdapt_a
8	C00830/21	MCTRL: nPosCtrlOutLimit_a
9	C00833/27	MCTRL: bPosCtrlOn
10	C00490	Position encoder selection
11	C00495	Speed sensor selection
12	C00833/68	MCTRL: bMotorRefOffsetOn
13	C00834/6	MCTRL: dnMotorRefOffset_p

Parameters		Info
14	C01206/1	MCK: Mounting direction: Motor
15	C00909/1	Max. pos. speed
16	C00909/2	Max. neg. speed
17	C00105	Decel. time - quick stop
18	C00104/1	Quick stop setting
19	C00497/4	LS_Resolver: Encoder filter time FreqIn67
20	C00833/30	bTorquemodeOn
21	C00833/33	bQSPOn
22	C00830/88	MCTRL: nSpeedHighLimit_a
23	C00830/23	MCTRL: nSpeedLowLimit_a
24	C00833/28	MCTRL: bSpeedInterp.
25	C00050	Speed setpoint

Parameters		Info
26	C00830/29	Limitation of torque in motor mode
27	C00830/28	Limitation of torque in generator mode
28	C00830/24	MCTRL: nSpeedCtrlI_a
29	C00833/31	MCTRL: bSpeedCtrlIOn
30	C00830/25	MCTRL: nSpeedCtrlPAdapt_a
31	C00070/2	SC: Vp speed controller
32	C00071/2	SC: Ti speed controller
33	C00072	SC: Tdn speed controller

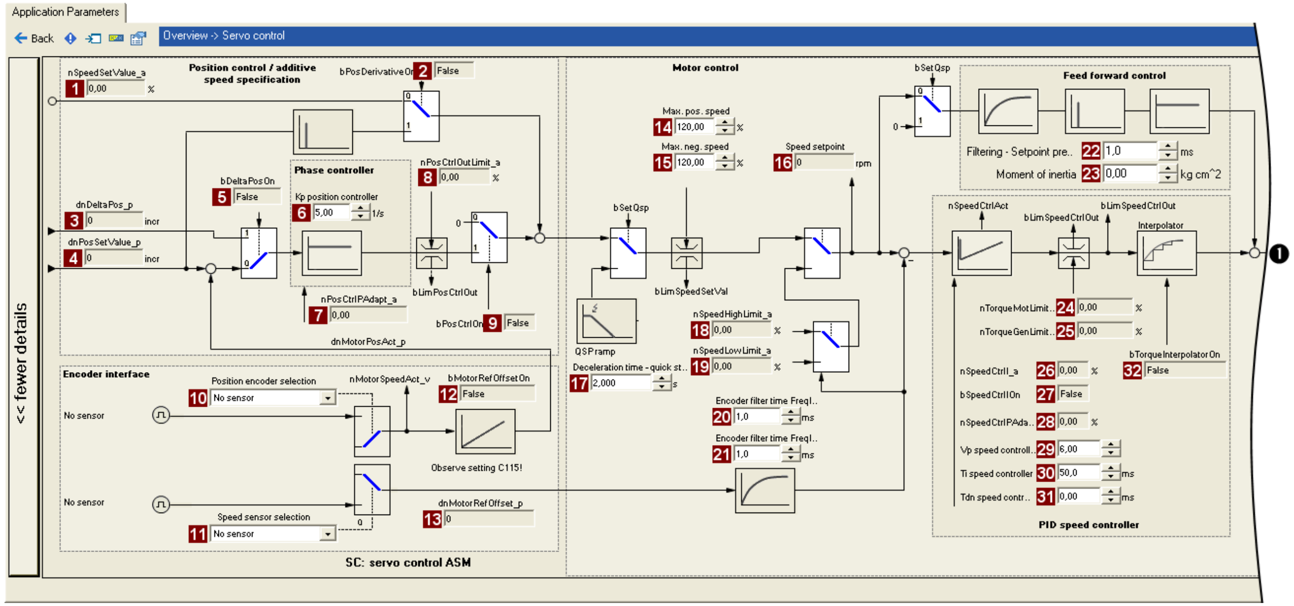


	Parameters	Info		Parameters	Info		Parameters	Info
1	C00830/27	MCTRL: nTorqueSetValue_a	11	C00063/1	Motor temperature	21	C00022	Imax in motor mode
2	C00830/29	Limitation of torque in motor mode	12	C00056/2	Actual torque	22	C00023	Imax in generator mode
3	C00830/28	Limitation of torque in generator mode	13	C00270	SC: Freq. current setpoint filter	23	C00833/29	MCTRL: bTorqueInterpolatorOn
4	C00833/30	bTorquemodeOn	14	C00271	SC: Current setpoint filter width	24	C00937/2	Torque-producing current
5	C00833/33	bQSPOn	15	C00272	SC: Current setpoint filter depth	25	C00937/1	Maximally effective field-producing motor current
6	C02865/1	Special settings (bit 12)	16	C00056/1	Torque setpoint	26	C00022	Imax in motor mode
7	C01206/1	MCK: Mounting direction: Motor	17	C00085	Motor stator leakage inductance	27	C00023	Imax in generator mode
8	C02865/1	Sondereinstellungen (Bit 13)	18	C00089	Rated motor frequency	28	C00938	Limitation of the field-producing motor current
9	C00274	SC: Max. change in acceleration	19	C00090	Rated motor voltage			
10	C00280	SC: Filter time const. DC detection	20	C00079/4	Field weakening			

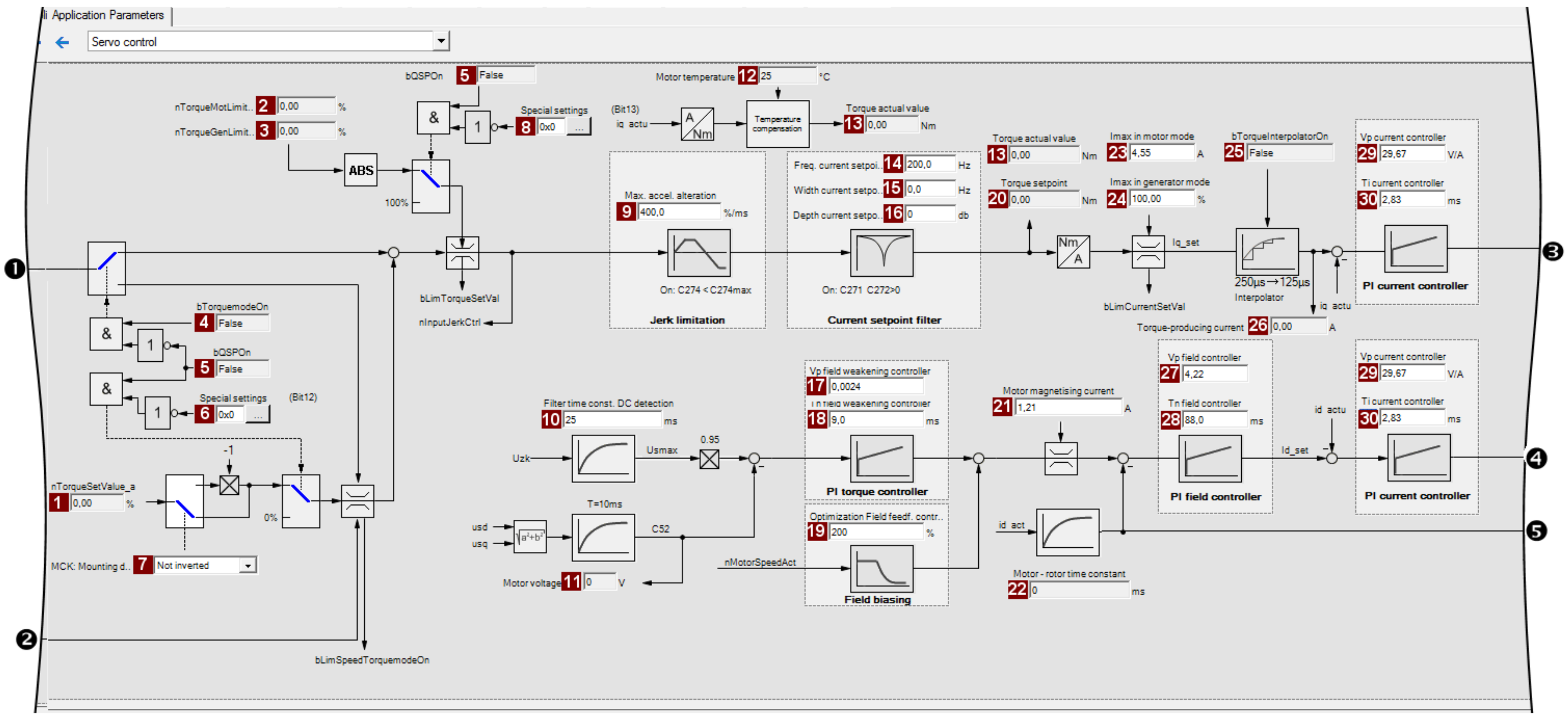


Parameters		Info
1	C00075	Vp current controller
2	C00076	Ti current controller
3	C00079/1	SC: Current controller - feedforward control
4	C00969/1	Motor - number of pole pairs
5	C00018	Switching frequency
6	C00830/32	MCTRL: nPWMAngleOffset_a
7	C00830/31	MCTRL: nVoltageAdd_a
8	C00058	Output frequency
9	C00052	Motor voltage
10	C00058	Output frequency
11	C00910/1	Max. pos. output frequency
12	C00910/2	Max. neg. output frequency

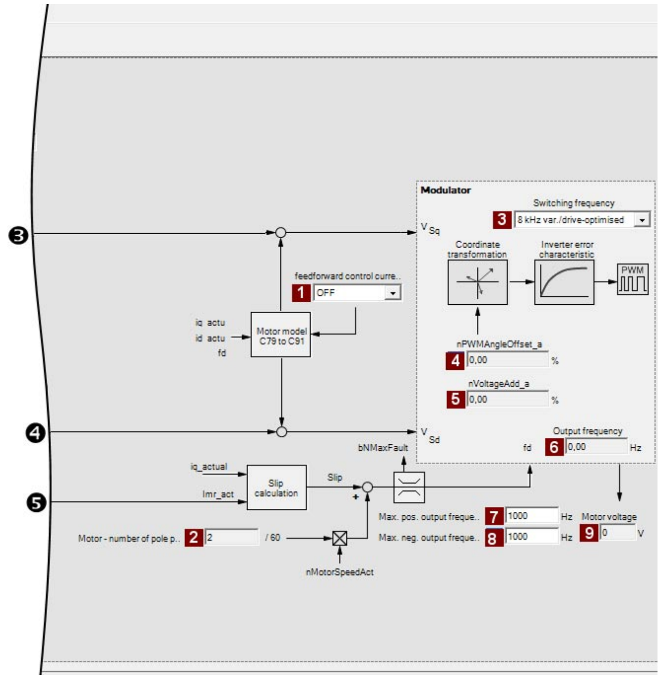
Signal flow for servo control for asynchronous motor (ASM):



Parameters		Info	Parameters		Info	Parameters		Info
1	C00830/22	Speed setpoint	14	C01206/1	MCK: Mounting direction: Motor	26	C00830/29	Limitation of torque in motor mode
2	C00833/67	MCTRL: bPosDerivativeOn	15	C00909/1	Max. pos. speed	27	C00830/28	Limitation of torque in generator mode
3	C00834/4	MCTRL: dnDeltaPos_p	16	C00909/2	Max. neg. speed	28	C00830/24	MCTRL: nSpeedCtrlI_a
4	C00834/5	MCTRL: dnPosSetValue_p	17	C00105	Decel. time - quick stop	29	C00833/31	MCTRL: bSpeedCtrlIOn
5	C00833/35	MCTRL: bDeltaPosOn	18	C00104/1	Quick stop setting	30	C00830/25	MCTRL: nSpeedCtrlPAdapt_a
6	C00254	Kp position controller	19	C00497/4	LS_Resolver: Encoder filter time FreqIn67	31	C00070/3	SC: Vp speed controller
7	C00830/20	MCTRL: nPosCtrlPAdapt_a	20	C00833/30	bTorquemodeOn	32	C00071/3	SC: Ti speed controller
8	C00830/21	MCTRL: nPosCtrlOutLimit_a	21	C00833/33	bQSPOn	33	C00072	SC: Tdn speed controller
9	C00833/27	MCTRL: bPosCtrlOn	22	C00830/88	MCTRL: nSpeedHighLimit_a			
10	C00490	Position encoder selection	23	C00830/23	MCTRL: nSpeedLowLimit_a			
11	C00495	Speed sensor selection	24	C00833/28	MCTRL: bSpeedInterp.			
12	C00833/68	MCTRL: bMotorRefOffsetOn	25	C00050	Speed setpoint			
13	C00834/6	MCTRL: dnMotorRefOffset_p						



Parameters		Info
1	C00830/27	MCTRL: nTorqueSetValue_a
2	C00830/29	Limitation of torque in motor mode
3	C00830/28	Limitation of torque in generator mode
4	C00833/30	MCTRL: bTorqueModeOn
5	C00833/33	bQSPOn
6	C02865/1	Special settings (bit 12)
7	C01206/1	MCK: Mounting direction: Motor
8	C02865/1	Sondereinstellungen (Bit 13)
9	C00274	SC: Max. change in acceleration
10	C00280	SC: Filter time const. DC detection
11	C00052	Motor voltage
12	C00063/1	Motor temperature
13	C00056/2	Actual torque
14	C00270	SC: Freq. current setpoint filter
15	C00271	SC: Current setpoint filter width
16	C00272	SC: Current setpoint filter depth
17	C00577	SC: Vp field weakening controller
18	C00578	SC: Tn field weakening controller
19	C00576	SC: Field feedforward control
20	C00056/1	Torque setpoint
21	C00095	Motor magnetising current
22	C00083	Motor rotor time constant
23	C00022	Imax in motor mode
24	C00023	Imax in generator mode
25	C00833/29	MCTRL: bTorqueInterpolatorOn
26	C00937/2	Torque-producing current
27	C00077	SC: Vp field controller
28	C00078	SC: Tn field controller
29	C00075	Vp current controller
30	C00076	Ti current controller



Parameters		Info	
1	C00079/1	SC: Current controller - feedforward control	
2	C00969/1	Motor - number of pole pairs	
3	C00018	Switching frequency	
4	C00830/32	MCTRL: nPWMAngleOffset_a	
5	C00830/31	MCTRL: nVoltageAdd_a	
6	C00058	Output frequency	
7	C00910/1	Max. pos. output frequency	
8	C00910/2	Max. neg. output frequency	
9	C00052	Motor voltage	

5.9.2 Types of control

The servo control can be operated in two different modes:

- Speed control with torque limitation (*bTorquemodeOn* = FALSE)
- Torque control with speed limitation (*bTorquemodeOn* = TRUE)

Tip!

A position-controlled application requires a speed control with torque limitation (*bTorquemodeOn* = FALSE).

5.9.2.1 Speed control with torque limitation

When *bTorquemodeOn* = FALSE, the drive system is operated with a selected speed setpoint in a speed-controlled manner.

Note!

From version 13.00.00, the torque setpoint *nSpeedSetValue_a* is set to 0 by quick stop (QSP) device-internally and the torque limit values *nTorqueMotLimit_a* and *nTorqueGenLimit_a* are set to 100 % to stop the drive quickly and safely anytime. The previous behaviour can be set in C2865/1 via bit 12 and bit 13.

The operational performance can be adapted in the following ways:

A. Overload limitation in the drive train

B. Motor current limitation

Overload limitation in the drive train

The torque is limited via the torque setpoint.

- The torque setpoint is identical to the value at the output of the speed controller, *nOutputSpeedCtrl*.
- To avoid overload in the drive train, the torque in motor mode can be limited via the *nTorqueMotLimit_a* process input signal, and the torque in generator mode can be limited via the *nTorqueGenLimit_a* process input signal:

Designator DIS code | data type	**Information/possible settings**
nTorqueMotLimit_a C00830/29 | INT	Torque limitation in motor mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: 0 ... +199.99 % • If keypad control is performed: Parameterisable via C00728/1. From version 18.00.00 onwards: C02864: Bit 15 = 1: positive torque limitation (nTorqueHighLimit_a)
nTorqueGenLimit_a C00830/28 | INT	Torque limitation in generator mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: -199.99 ... 0 % • If keypad control is performed: Parameterisable via C00728/2. From version 18.00.00 onwards: C02864: Bit 15 = 1: negative torque limitation (nTorqueLowLimit_a)

Note!

To avoid instabilities during operation, the torque limit values are internally processed as absolute values.

Motor current limitation

A cross current setpoint is calculated from the torque setpoint which is limited depending on the magnetising current, the max. current in motor mode (C00022) and the max. current in generator mode (C00023). The total current injected into the motor does not exceed the max. currents in motor mode and in generator mode.

Note!

For synchronous motors, the Lenze setting of the torque limits *nTorqueMotLimit_a* and *nTorqueGenLimit_a* to 100 % can result in the activation of the torque limitation for motor temperature < max. motor temperature before the set current limit values (C00022, C00023) are reached.

- With a motor temperature of approx. 20°C and maximum load, the maximum current will then be set to approx. 15 % below the set current limit values.
- Remedy: If the torque limits *nTorqueMotLimit_a* and *nTorqueGenLimit_a* are increased to 115 %, the set current limit values can also be reached for a motor temperature of 20°C and maximum load. Alternatively, the motor temperature compensation can be switched off via C02878. In this case, however, the torque accuracy decreases.

5.9.2.2 Torque control with speed limitation

When *bTorquemodeOn* = TRUE, a torque-controlled operation is activated. The setpoint torque directly follows the default value *nTorqueSetValue_a*.

Due to its speed limitation, the torque-controlled drive can only rotate within a speed range whose positive speed is limited by *nSpeedHighLimit_a* and whose negative speed is limited by *nSpeedLowLimit_a*.

Note!

- Absolute speed limitation to speed 0 rpm (*nSpeedLowLimit_a* or *nSpeedHighLimit_a* = 0) is only possible from version 12.00.00.
- Quick stop (QSP) is used to switch over to Speed control with torque limitation.
    - From version 13.00.00, the torque setpoint *nSpeedSetValue_a* is set to 0 by quick stop (QSP) device-internally and the two torque limit values *nTorqueMotLimit_a* and *nTorqueGenLimit_a* are set to 100 % to stop the drive quickly and safely anytime. The previous behaviour can be set in C2865/1 via bit 12 and bit 13.

- From version 13.00.00, the *bLimSpeedTorquemodeOn* status signal is used to show that the speed limitation is active.
- The speed is defined by the process.
- The torque setpoint is calculated directly from *nTorqueSetValue_a*.
    - From version 12.00.00 onwards, the torque limitation is active via *nTorqueMotLimit_a* and *nTorqueGenLimit_a* in this control mode, too, for the torque setpoint to be limited. The torque limitation can be deactivated in C2865/1 via bit 0 to obtain the previous function.

Designator DIS code | data type	**Information/possible settings**
nTorqueSetValue_a C00830/27 | INT	Torque setpoint / additive torque • Scaling: 16384 ≡ 100 % M_{max} (C00057)
nSpeedHighLimit_a C00830/88 | INT	Upper speed limit for the speed limitation • During torque-controlled operation only (*bTorquemodeOn* = TRUE) • Scaling: 16384 ≡ 100 % rated speed (C00011)
nSpeedLowLimit_a C00830/23 | INT	Lower speed limit for speed limitation • During torque-controlled operation only (*bTorquemodeOn* = TRUE) • Scaling: 16384 ≡ 100 % rated speed (C00011)
nTorqueMotLimit_a C00830/29 | INT	Torque limitation in motor mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: 0 ... +199.99 % • If keypad control is performed: Parameterisable via C00728/1. From version 18.00.00 onwards: C02864: Bit 15 = 1: positive torque limitation (nTorqueHighLimit_a)
nTorqueGenLimit_a C00830/28 | INT	Torque limitation in generator mode • Scaling: 16384 ≡ 100 % M_{max} (C00057) • Setting range: -199.99 ... 0 % • If keypad control is performed: Parameterisable via C00728/2. From version 18.00.00 onwards: C02864: Bit 15 = 1: negative torque limitation (nTorqueLowLimit_a)

5.9.3 Basic settings

The following "Initial commissioning steps" must be performed to commission the servo control:

<table>
<tr><th colspan="3">Initial commissioning steps</th></tr>
<tr><td>1</td><td colspan="2">Determine the motor control:
• For the closed-loop control of a synchronous motor (PSM):
C00006 = "1: SC: Servo control PSM"
• For the closed-loop control of an asynchronous motor (ASM):
C00006 = "2: SC: Servo control ASM"</td></tr>
<tr><td rowspan="2">2.</td><td colspan="2">Set the motor selection/motor data
• When selecting and parameterising the motor, the motor nameplate data and the equivalent circuit diagram data are relevant. Detailed information can be found in the "Motor selection/Motor data" chapter. (📖 144)
Depending on the motor manufacturer, proceed as follows:</td></tr>
<tr><td>Lenze motor:
Selecting a motor from the motor catalogue in the »Engineer«
- or -
1. Set the motor nameplate data
2. Automatic motor data identification</td><td>Third party manufacturer's motor:
1. Set the motor nameplate data
2. Automatic motor data identification or set known equivalent circuit diagram data manually:
C00082: Motor rotor resistance*
C00084: Motor stator resistance
C00085: Motor stator leakage inductance
C00092: Motor magnetising inductance*
C00095: Motor magnetising current*
* Setting only required for asynchronous motors.</td></tr>
<tr><td>3.</td><td colspan="2">Define the type of control:
bTorquemodeOn = FALSE: Speed control with torque limitation
bTorquemodeOn = TRUE: Torque control with speed limitation</td></tr>
<tr><td>4.</td><td colspan="2">Parameterise the encoder/feedback system. ▸ Encoder/feedback system (📖 330)</td></tr>
<tr><td>5th</td><td colspan="2">Only with servo control for synchronous motors (PSM):
Detect pole position of the motor. ▸ Pole position identification (PPI) (📖 379)</td></tr>
</table>

Note!

- The Lenze settings of the inverter are predefined for a power-adapted standard asynchronous motor. For an optimal drive behaviour, we recommend to adapt the controller settings.
- When controlling an asynchronous motor (ASM), magnetisation is carried out prior to enabling the setpoint after a controller enable in the Lenze setting. The (delayed) setpoint enabling ensures direct availability of the maximum torque at the motor.
  - The delay of the setpoint enabling associated with the magnetisation can be reduced by reducing the threshold set in C00918. However, this also reduces the maximum torque which is directly available after the setpoint enabling.
  - If C00918 = "0 %", magnetisation is not carried out.

Tip!

Information on the optimisation of the control mode and the adaptation to the real application is provided in chapter "Optimising the control mode". (📖 251)

Parameterisable additional functions are described correspondingly in the chapter "Parameterisable additional functions". (📖 272)

5.9.4 Optimising the control mode

Note!

When the inverter has been enabled, starting up is delayed by the magnetization process of the motor. The speed setpoint is only enabled for motor control if 87 % of the motor magnetising current are available. If this delay is not tolerable for certain applications, the preset percentage threshold can be reduced in C00918.

From version 12.00.00:

- Following successful motor parameter identification, the current controller parameters (C00075, C00076) and field controller parameters (C00077, C00078) are calculated automatically.
  - If these parameters are not to be calculated, bit 4 of C02865/1 must be set to "1".
- Following successful motor parameter identification, the speed controller parameters (C00070/2, C00071/2, C00072) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "5".
- Following successful motor parameter identification, other controller parameters (C00011, C00022, C00497) can be calculated automatically.
  - If these parameters are to be calculated, bit 6 of C02865/1 must be set to "6".

The "optimisation steps" given in the table below serve to further optimise the control behaviour of the servo control and adjust it to the concrete application.

- Detailed information on the individual steps can be found in the following subchapters.

Generally, the following optimisation steps are recommended:

Optimisation steps	
1	Optimise current controller. (📖 252) • The current controller should always be optimised if a motor of a third-party manufacturer with unknown motor data is used!
2.	Optimise speed controller. (📖 253) • The setting of the speed controller must be adapted depending on the mechanical path.
3.	Optimise response to setpoint changes and determine mass inertia. (📖 256) • For an optimal reference behaviour, the total moment of inertia can be used to make a feedforward control of the speed setpoint.

Special cases may require further optimisation steps:

Optimisation steps	
1	Setting the current setpoint filter (band-stop filter). (📖 259) • In order to suppress or damp (mechanical) resonant frequencies, a current setpoint filter is integrated in the speed control loop which is switched off in the default setting but can be parameterised accordingly, if required. Then readjust the speed controller: Optimise speed controller. (📖 253)
2.	Adapting the max. acceleration change (jerk limitation). (📖 260)
3.	Only with servo control for asynchronous motors (ASM): Optimising the behaviour of the asynchronous motor in the field weakening range. (📖 265) (For synchronous motors, this function is in preparation)

Tip!

In order to traverse a typical speed profile for optimisation of motor control, you can also use the basic function "Manual jog" with appropriately adapted manual jog parameters if this basic function is supported by the selected technology application. (📖 658)

5.9.4.1 Optimise current controller

Note!

An optimisation of the current controller should generally be carried out unless a power-adapted standard asynchronous motor is used or the motor has been selected from the motor catalogue of the »Engineer«!

An optimisation of the current controller is sensible since the two control parameters gain (C00075) and reset time (C00076) depend on the required maximum current and the set switching frequency.

Parameters	Info	Lenze setting	
		Value	Unit
C00075	Vp current controller	7.00	V/A
C00076	Ti current controller	10.61	ms

- Gain and reset time can be calculated as per the following formulae:

$$V_p = \frac{L_{ss}[H]}{T_E[s]}$$

$$T_i = \frac{L_{ss}[H]}{R_s[\Omega]}$$

V_p = Current controller gain (C00075)
T_i = Current controller reset time (C00076)
L_{ss} = Motor stator leakage inductance (C00085)
R_s = Motor stator resistance (C00084)
T_E = Equivalent time constant (= 500 µs)

5.9.4.2 Optimise speed controller

Note!

With servo control, an optimisation of the speed controller is recommended in order that the connection to a motor rotating with very high speed after controller enable always functions properly.

The speed controller is in the form of a PID controller with an additional differential speed-setpoint gain. For optimum behaviour, the PID speed controller has to be optimised and the overall mass inertia of the drive train has to be determined.

- In the Lenze setting, the configuration of the speed controller provides robustness and moderate dynamics.

Parameters	Info	Lenze setting	
		Value	Unit
C00070/2	SC: Vp speed controller	6.00	
C00071/2	SC: Ti speed controller	50.0	ms
C00072	SC: Tdn speed controller	0.00	ms

Speed controller gain Vp

The gain Vp (C00070/2) of the speed controller is defined in a scaled representation which enables a comparable parameterisation almost independent of the power of the motor or inverter. Here, the speed input difference of the controller is scaled to the rated motor speed whereas the output torque refers to the rated motor torque. A gain of 10 means that a speed difference of 1 % is gained through the P component with 10 % torque.

If the rated data of the motor and the mass inertia of the drive system are known, we recommend the following setting:

$$V_P \approx 1.5 \ldots 3 \cdot \frac{T_M[s]}{0.01[s]}$$

$$T_M[s] = \frac{2 \cdot \pi \cdot n_N[rpm]}{M_N[Nm] \cdot 60} \cdot J_{Drive,\,total}[kgm^2]$$

$$M_N[Nm] = \frac{P_N[W] \cdot 60}{2 \cdot \pi \cdot n_N[rpm]}$$

V_P = Gain of the speed controller (C00070/1)
T_M = Time constant for the acceleration of the motor
M_N = Rated motor torque
n_N = Rated motor speed
$J_{drive,\,total}$ = Total moment of inertia of the drive

[5-16] Recommendation for the setting of the gain of the speed controller

If the mass inertia of the drive is unknown, the optimisation can be achieved as follows:

1. Specify speed setpoint.
2. Increase Vp (C00070/2) until the drive is unstable (observe motor noise).
3. Reduce Vp (C00070/2) until the drive runs stable again.
4. Reduce Vp (C00070/2) to approx. half the value.

Tip!

Values recommended by Lenze for the setting of the (proportional) gain:
- For drive systems without feedback: Vp = 6 ... 20
- For drive systems with a good disturbance behaviour: Vp > 12

Speed controller reset time Ti

Apart from setting the P component, C00071/2 provides the possibility to take influence on the I component of the PI controller.

If the mass inertia of the drive is unknown, the optimisation can be achieved as follows:

1. Specify speed setpoint.
2. Reduce Ti (C00071/2) until the drive is unstable (observe motor noise).
3. Increase Ti (C00071/2) until the drive runs stable again.
4. Increase Ti (C00071/2) to approx. twice the value.

Tip!

Value range recommended by Lenze for the setting of the reset time:

Ti = 20 ms ... 150 ms

Differential time constant Tdn (rate time)

The differential time constant Tdn of the speed controller can be set in C00072.

If the mass inertia of the drive is unknown, the optimisation can be achieved as follows:

- Increase Tdn (C00072) during operation until optimal control mode is reached.

Using the ramp response for setting the speed controller

If the mechanical components cannot be operated at the stability limit, the ramp response can also be used for setting the speed controller.

Stop!

If the controller parameters are preset unfavourably, the control can tend to heavy overshoots up to instability!
- Following and speed errors can adopt very high values.
- If the mechanics are sensitive, the corresponding monitoring functions are to be activated.

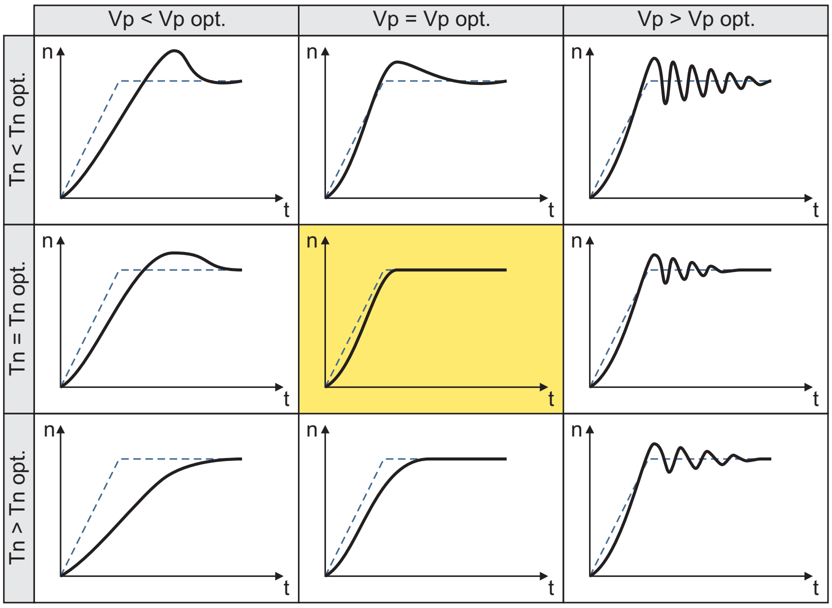
Note!

For an optimal setting, we recommend to determine the mass inertia (optimal response to setpoint changes) first.

▸ Optimise response to setpoint changes and determine mass inertia (📖 256)

How to optimise the speed controller setting by means of the ramp response:

1. Run a typical speed profile and record the ramp response of the speed using the data logger.
   - Motor control variables to be recorded:
     nSpeedSetValue_a (speed setpoint)
     nMotorSpeedAct_a (actual speed value)
2. Evaluate the ramp response:

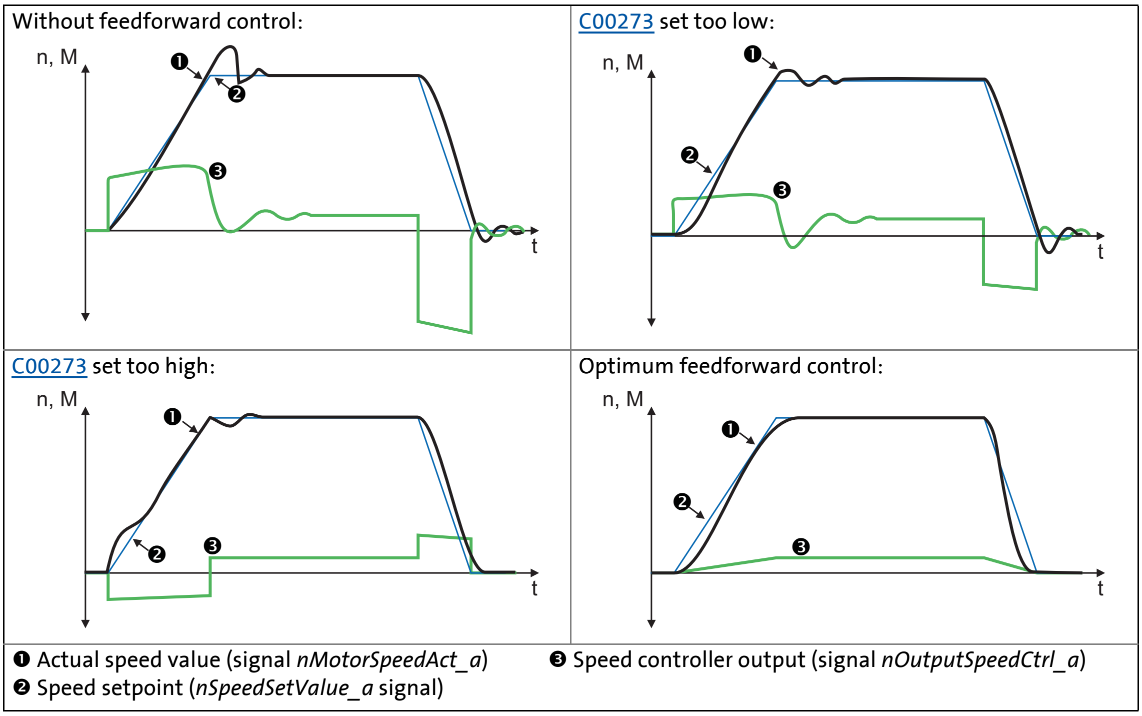


   - Solid line = ramp response (actual speed value)
   - Dash line = speed setpoint
3. Change gain Vp in C00070/2 and reset time Tn in C00071/2.
4. Repeat steps 1 ... 3 until the optimum ramp response is reached.

5.9.4.3 Optimise response to setpoint changes and determine mass inertia

Optimisation at constant mass inertia

Setting the total moment of inertia under C00273 provides the optimum torque feedforward control. Depending on the application, an adjustment of the setting under C00273 may be necessary to optimise the response to position/speed setpoint changes by means of the torque feedforward control.



[5-17] Typical signal characteristics for different settings of the load moment of inertia

How to optimise the torque feedforward control:

1. Run a typical speed profile and record the inputs and outputs of the speed controller with the data logger.
   - Motor control variables to be recorded:
     nSpeedSetValue_a (speed setpoint)
     nMotorSpeedAct_a (actual speed value)
     nOutputSpeedCtrl_a (speed controller output)
2. Estimate the moment of inertia and set it in C00273 in relation to the motor end (i.e. with account being taken of the gearbox factors).
3. Repeat the data logger recording (see step 1).

   Now the data logger should show that part of the required torque is generated by the feedforward control and the speed controller output signal (*nOutputSpeedCtrl_a*) is correspondingly smaller. The resulting following error decreases.
4. Change the setting in C00273 and repeat the data logger recording until the intended response to setpoint changes is reached.
   - The optimisation could aim at the speed controller being completely relieved (see signal characteristics in Fig. [5-17]).
5. Save the parameter set (device command: C00002/11).

Optimisation at variable mass inertia

From version V12.00.00, mass inertia that changes during the process (e.g. a reel) can be taken into account when optimising the response to setpoint changes.

How to proceed:

1. In C00273 the known constant total moment of inertia (motor, gearbox, shaft, etc.) must be set or determined according to previous instructions ("How to optimise ...").
   - The determination requires travelling the typical speed profile without variable mass inertia (e.g. reels).
2. At the LS_MotorInterface SB, the *nInertiaAdapt_a* process signal must be interconnected in a way which ensures that a value of "100 %" is applied at this input.
3. In C00919/1 set the known maximum value of the variable moment of inertia or determine the value according to previous instructions ("How to optimise ...").
   - The determination requires travelling the typical speed profile including variable mass inertia (e.g. maximum reels).
4. The *nInertiaAdapt_a* process signal can be used during the process to dynamically control the percentage of the variable moment of inertia set in C00919/1 which is to be considered for setpoint feedforward control.
   Example:
   - If there is no variable moment of inertia (e.g. no reel), the *nInertiaAdapt_a* process signal must be set to "0 %".
   - If the maximally variable moment of inertia is available (e.g. maximum reels), the *nInertiaAdapt_a* process signal must be set to "100 %".

Tip!

Via the *nTorqueSetValue_a* process signal at the LS_MotorInterface SB, any differential signal can be defined for torque feedforward control. First the speed controller, then this additive torque is connected which is hence not derived from the differential change in speed setpoint.

Other functions for differential setpoint feedforward control

From version V12.00.00 onwards, the following additional functions are available for differential setpoint feedforward control (torque feedforward control):

- In C00653/1, the sensitivity of setpoint feedforward control can be adapted.
- In C00654/1, alternatively to the *nSpeedSetValue_a* process signal, the new *nSpeedSetValueInertia_a* process signal for the setpoint feedforward control can be selected at the LS_MotorInterface SB. Via the *nSpeedSetValueInertia_a* process signal, an optional input value (e.g. setpoint of the position or process controller) for the torque feedforward control can be specified.
- For *bTorqueModeOn* = TRUE, the setpoint feedforward control is added to the torque setpoint *nTorqueSetValue_a*. In this way, feedforward control of torque is also possible for torque-controlled operation (e.g. for winder applications).

From version V18.00.00 onwards, the following additional functions are available for differential setpoint feedforward control (torque feedforward control):

- In C00654/1, an exact torque feedforward control can be set for the nSpeedSetValue_a speed setpoint. In case of a very dynamic application, set C00654/1 = 2.
- From version V15.00.00 onwards, the torque feedforward control behaves differently than in version V14.00.00 or older. In order to achieve the same behaviour as in version V14.00.00 or older, set C00654/1 = 3.

STOP

Stop!

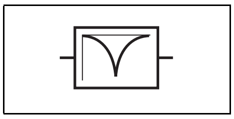
In the Lenze setting of C00654/1, the speed setpoint of the speed controller (*nSpeedSetValue_a*) is used for the torque feedforward control which is why it is also called "differential setpoint feedforward control". A very sharp change of the speed setpoint at the speed controller thus causes a strong torque impulse at the machine!

Causes for a very sharp change of the speed setpoint of the speed controller:

- A control creates the setpoint ramp itself and the speed setpoint is only written every 20 ms to the drive. (In this case, the speed setpoint is changed every 20 ms.)
  - Recommendation: Deactivate the torque feedforward control for the speed setpoint in C00654/1 if the speed setpoint changes very sharply or the mass inertia is unknown (e.g. in case of hoists)!
- The reference speed C0011 is much higher than the rated motor speed set in C0087.
  - Recommendation: Deactivate the torque feedforward control for the speed setpoint in C00654/1 if the reference speed C0011 is 5 times higher than the rated motor speed C0087.

5.9.4.4 Setting the current setpoint filter (band-stop filter)

Due to the high dynamic performance/limit frequency of the closed current control loop, mechanical natural frequencies can be activated which may lead to an unstable speed control loop.



To mask out or at least damp these resonant frequencies, a so-called current setpoint filter is integrated into the speed control loop of the inverter.

Parameters	Info	Lenze setting	
		Value	Unit
C00270	SC: Freq. current setpoint filter	200.0	Hz
C00271	SC: Current setpoint filter width	0.0	Hz
C00272	SC: Current setpoint filter depth	0	dB

- In the default setting of 0 db of the filter depth (C00272), the current setpoint filter is switched off.

Setting of the current setpoint filter

Since the frequency response of the speed controlled system is only rarely known to such an extent that the current setpoint filter can be adjusted to the controlled system in the run-up, the following example describes how to set the current setpoint filter.

How to set the current setpoint filter:

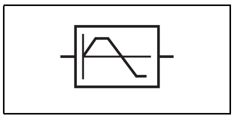
1. Optimise current controller (📖 252).
2. Optimise speed controller (📖 253)
3. Measure the oscillation frequency (observe current or speed).
4. Set the measured oscillation frequency in C00270 as filter frequency.
5. Set "25%" of the filter frequency in C00271 as filter width.
   - Example: Filter frequency = 200 Hz → filter width = 50 Hz.
6. Set "40 dB" in C00272 as filter depth.
   - If the filter depth is set to "0 dB" (default setting), the filter is not active.

Note!

Readjust the speed controller after setting the current setpoint filter. ▸ Optimise speed controller. (📖 253)

The setting of the current setpoint filter reduces the available maximum drive torque.

5.9.4.5 Adapting the max. acceleration change (jerk limitation)



Via the max. acceleration change that can be set in C00274, the change of the setpoint torque can be limited for jerk limitation. Hence, sudden torque step changes can be avoided. The entire speed characteristic is smoothed.

Parameters	Info	Lenze setting	
		Value	Unit
C00274	SC: Max. change in acceleration	400.0	%/ms

In the default setting of 400 %/ms of the max. acceleration change (C00274), jerk limitation is switched off.

The setting defines the permissible maximum torque change per ms (based on the rated motor torque).

Note!

Only activate this jerk limitation for speed-controlled applications!

If table positioning or a free function block interconnection with a positioning operating mode is selected, jerk limitation must be switched off.

- Here, jerk limitation is provided for in the travel profile generator. Setting this jerk limitation in the motor control would lead to following errors!

5.9.4.6 Slip calculation from motor equivalent circuit diagram data

This function extension is available from version 02.00.00!

In order to achieve a better speed stability and torque accuracy, the slip calculation can be either derived from the motor nameplate data (e.g. rated motor speed) or the motor equivalent circuit diagram data (stator resistance, rotor resistance etc.).

In case of servo control for asynchronous motors (C00006 = "2"), the slip calculation is always made

- up to and including version 01.xx.xx from the motor nameplate data.
- from version 02.00.00 onwards in the Lenze setting from the motor equivalent circuit diagram data in order to obtain a higher torque accuracy.

The data to be used for servo control (for asynchronous motors) is selected via bit 1 in C02879/1:

setting		Info
Bit 0	SLVC	In case of sensorless vector control: • "0" ≡ Slip calculation from motor nameplate data (Lenze setting) • "1" ≡ Slip calculation from motor equivalent circuit diagram data
Bit 1	SC_ASM	In case of servo control for asynchronous motors: • "0" ≡ Slip calculation from motor nameplate data • "1" ≡ Slip calculation from motor equivalent circuit diagram data (Lenze setting)
Bit 2 ... 7	Reserved	

Note!

In order that the slip can be calculated from the motor equivalent circuit diagram data, the equivalent circuit data (stator resistance, rotor resistance etc.) must be known as exactly as possible.

- Selecting a motor in the »Engineer« motor catalogue loads the exact motor equivalent circuit diagram data.
- When the motor nameplate data is entered manually and the motor equivalent circuit diagram data is then detected via the motor parameter identification, the "extended identification" (C02867/1 = 2) must be used. ▶ Automatic motor data identification (📖 151)

In case of servo control for asynchronous motors, the slip calculation can also be made from the motor nameplate data (C02879/Bit 1 = "0"). In this case, the slip is too high by a factor of 1.35 as the rated motor speed (C00087) is assumed for a cold machine. This has the following consequences:

- The higher slip provides a better behaviour for the field weakening range.
- In the motor setting range, a considerably too high torque is detected (torque accuracy is worse).
- The achieved maximum torque in the motor setting range is smaller.

From version 15.00.00, the excessive slip with the activated "slip calculation from motor nameplate data" option (C02879/bit 1 = "0") can be corrected by setting bit 7 to "1" in C02864/1. With this setting, the rated motor speed (C00087) for a warm machine is assumed.

- By selecting the motor in the »Engineer« motor catalogue or executing the motor parameter identification, bit 7 is set to "1" in C02864/1.

Note!

Exchange of devices

When devices with a parameterised servo control for asynchronous motors are exchanged, problems regarding the control quality may occur if motor parameters set for a device with version 01.xx.xx are tranferred to a replacement device of a higher version.

- In this case, we recommend to load the very precise motor data available from »Engineer« V2.17 or to execute the "advanced motor parameter identification" (C02867/1 = 2). ▶ Automatic motor data identification (📖 151)
- As an alternative, the slip calculation from the motor nameplate data can be set again in C02879/1 (bit 1 = "0") to obtain the previous behaviour.

5.9.4.7 Temperature compensation within the motor control

This function extension is available from version 02.00.00!

A temperature compensation over the detected motor temperature (display in C00063/1) serves to compensate inaccuracies in the output torque within the motor control in case of temperature changes of the asynchronous or synchronous motor.

Note!

For temperature correction within the motor control, the thermal detector is used that is also set for motor temperature monitoring in C01193.

- The temperature correction requires an error-free temperature detection via PT1000 or KTY. (display in C00063/1)
- **Important**: the temperature detection is only available if a multi encoder or resolver is used as encoder.
- If the value 255 °C is displayed in C00063/1, the temperature detection is faulty or not available. The temperature correction within the motor control is switched off.

Behaviour in case the temperature compensation is not active

If the temperature compensation is not active,

- we assume a motor temperature of 20 °C when using an asynchronous motor. If the asynchronous motor is at operating temperature, however, a too low actual torque value is detected, especially in the field weakening range. This leads to a strong decrease in the torque setting range.
- we assume a motor temperature of 150 °C when using a synchronous motor. If the synchronous motor is cold, however, a too high actual torque value is detected. This leads to a decrease in the torque setting range.

From version 14.00.00 onwards, this negative behaviour in case of warm asynchronous motors or cold synchronous motors can be improved by setting the option "warm machine" (bit= "1") in C02864. We then assume a motor temperature of 90 °C.

The temperature compensation is switched on/off bit-coded in C02878/1:

setting		Info
Bit 0	SC_PSM	Motor temperature compensation via PT1000 or KTY with servo control for synchronous motor: • "0" ≡ off • "1" ≡ On (Lenze setting)
Bit 1	SC_ASM	Motor temperature compensation via PT1000 or KTY with servo control for asynchronous motor: • "0" ≡ off • "1" ≡ On (Lenze setting)
Bit 2 ... 7	Reserved	

General information on temperature behaviour of the asynchronous and synchronous motor

In case of the **asynchronous motor**, the nameplate data always refer to rated data, i.e. on the rated load and thus on the rated temperature of the motor. For the 8400 device series, the equivalent circuit data for stator resistance (C00084) and rotor resistance (C00082) are always given for a cold asynchronous motor (20 °C) and converted to star connection. In case of a cold motor (20 °C), the real slip frequency thus corresponds to the calculated setpoint slip. In case of a warm motor, the real slip frequency is higher. This becomes especially apparent at high torques as an inaccuracy of the torque over the motor temperature.

In case of the **synchronous motor**. the field strength generated by the permanent magnet in the rotor is temperature-dependent. With a rated motor temperature, a rated field strength is generated. A negative temperature coefficient of -0.11 %/°C means that with a cold motor at the same setpoint torque a higher output torque is output than with a warm motor.

These errors must be compensated by means of the temperature compensation for asynchronous and synchronous motors in order to have a constant output torque over the motor temperature independent of the motor temperature.

5.9.4.8 Optimising the behaviour of the asynchronous motor in the field weakening range

The behaviour of the asynchronous motor in the field weakening range is influenced by:

- the field controller
- the field weakening controller
- the field feedforward control unit
- adaptive adaptation of the P component of the field and speed controllers
- An additional limitation of the I component of the controller when the voltage limit is reached

Note!

By default, these control parts are called every 1 ms. However, the 1-ms cycle is not always sufficient to achieve an optimum jerk-free transition to the field weakening range.

From version 14.00.00, the control parts can be called for an optimum jerk-free transition to the field weakening range in a 500-µs cycle. For this purpose, bit 5 has to be set to "1" int in C02864/1.

- With this setting, C00015 and C00576 do not have any influence anymore on the field feedforward control.
- We recommend this setting for new applications.
- In existing applications, this setting may require an increase of the reset times of the field weakening controller (C00578) and the field controller (C00078) by the factor 2 in order to achieve a stable operation.

In general, these settings are pre-optimised so that further optimisation is not required.

Optimisation for special motors (e.g. mid-frequency motors) or for standard asynchronous motors whose power is not adapted can be carried out according to the algorithms shown in the following sections.

From version 16.00.00 onwards, the dynamics of the field weakening control can be improved with C02864/1 (bit 9 = 1) by admitting also negative setpoints of the field current controller at the field controller output. Please note that this setting may cause higher torque peaks which can be reduced again by decreasing the field weakening controller gain (C00577) or increasing the reset time (C00578).

Field controller

The field controller is designed as a PI controller.

Parameters	Info	Lenze setting	
		Value	Unit
C00077	SC: Vp field controller	12.80	
C00078	SC: Tn field controller	256.0	ms

The gain Vp (C00077) of the field controller can be calculated using the motor rotor time constant and the equivalent time constant of the current-controlled motor:

$$V_{PFeld} = \frac{T_R[s]}{a^2 \cdot T_{Replacement}[ms]} \quad [\%]$$

V_{PField}: Gain of the field controller (C00077)
T_R: Motor rotor time constant (C00083)
$T_{Equivalent}$: Motor equivalent time constant (approx. 2.5 ms)
a: Measure for damping (z. B. a = 2)

If the rated data of the motor and the mass inertia of the drive system are known, we recommend the following setting:

$$V_{PFeld} \approx \frac{T_R[s]}{4 \cdot T_{Replacement}[ms]} = \frac{T_R[s]}{0.01[s]}$$

$$T_{NFeld}[ms] = T_R[ms]$$

V_{PField}: Gain of the field controller (C00077)
T_R: Motor rotor time constant (C00083)
T_{Equi}: Motor equivalent time constant
T_{NField}: Time constant of the field controller (C00078)

[5-18] Recommendation for the setting of the gain and the time constant of the field controller

Tip!

The motor rotor time constant depends on the motor rotor resistance, the magnetising inductance, and the leakage inductance.

- For an optimal calculation, we recommend to select the motor from the motor catalogue of the »Engineer« first if a Lenze motor is used. ▸ Selecting a motor from the motor catalogue in the »Engineer« (📖 149)
- If a third party manufacturer's motor is used, motor parameter identification must be carried out previously. ▸ Automatic motor data identification (📖 151)

Field weakening controller

The field weakening controller serves to adapt the magnetising current when the maximum control voltage has been reached so that in steady operation approximately 95 % of the maximally possible control voltage is output. Thus, there is a voltage reserve for dynamic load or speed variations.

Parameters	Info	Lenze setting	
		Value	Unit
C00577	SC: Vp field weakening controller	0.0010	
C00578	SC: Tn field weakening controller	20.0	ms

$$V_{P,\,FS} = 4774.65 \cdot \frac{L_H[mH] + L_{ss}[mH]}{L_H[mH]^2 \cdot n_N[rpm] \cdot I_N[A]}$$

$$C00577 = 4774.65 \cdot \frac{C00092\,[mH] + C00085\,[mH]}{C00092\,[mH]^2 \cdot C00087\,[rpm] \cdot C00088\,[A]}$$

$$T_{n,\,FS} = 20\ ms \cdot \frac{0.0010}{V_{P,\,FS}}$$

$$C00578 = 20\ ms \cdot \frac{0.0010}{C00577}$$

L_m: Mutual motor inductance (C00092)
L_{sl}: Motor stator leakage inductance (C00085)
n_{rated} = rated motor speed (C00087)
I_{rated} = rated motor current (C00088)

[5-19] Recommendation for the setting of the gain and the time constant of the field weakening controller

Tip!

A field weakening controller set too slow can be recognised by the fact that the actual speed value cannot follow the speed setpoint in the field weakening range. In this case, carry out the setting of the field weakening controller as described in the formula. If the speed setpoint still cannot follow the actual speed value, increase the setting in C00577 to maximally twice the value and reduce the setting in C00578 according to the formula mentioned above.

If the field weakening controller is set too fast, vibrations occur in the magnetising current, in the direct-axis current and in the cross current, in the torque and in the speed in the field weakening range. This is also audible by an increased motor noise (humming). The vibrations can be dampened by a decrease of the proportional gain of the field weakening controller.

Procedure:

- Operate the drive with max. required speed in the field weakening operation.
- Reduce Vp (C00577) of the field weakening controller until the vibration response cannot be detected anymore. Increase Tn (C00578) according to the formula.

Note!

Vp (C00577) can also be set to 0. In this case, Tn (C00578) should first be set to 2000 ms.

Field feedforward control

To have enough voltage reserve available for dynamic acceleration processes, a timely weakening of the field is required. The field weakening is controlled by the field feedforward control.

Parameters	Info	Lenze setting	
		Value	Unit
C00576	SC: Field feedforward control	200.0	%

Note!

The function of the field feedforward control and thus the field weakening control requires a correct setting of the V/f base frequency (C00015).

- The selection of the motor from the »Engineer« motor catalogue and the motor parameter identification serve to set the V/f base frequency correctly.
- When third-party motors are entered, either the V/f base frequency (C00015) has to be calculated manually or a motor parameter identification has to be carried out.
  Adapting the V/f base frequency (174)
  Automatic motor data identification (151)

Due to an error in the device software, the field feedforward control of the servo control only operates properly if the V/f base frequency (C00015) is reduced by the following factor after being set:

- 400-V devices: Factor 0.173
- 230-V devices: Factor 0.34

The same factor has to be considered for the field feedforward control (C00576).

From version 14.00.00, the control parts can be called for an optimum jerk-free transition to the field weakening range in a 500-µs cycle. For this purpose, bit 5 has to be set to "1" int in C02864/1.

- With this setting, C00015 and C00576 do not have any influence anymore on the field feedforward control.
- We recommend this setting for new applications.

From version 16.00.00 onwards, the dynamic performance of the field weakening control is improved with C02864/1, bit 9 = "1": like this, negative (field) current setpoints are also permitted at the output of the field controller. If this causes higher torque peaks, they can be reduced by either reducing the field weakening controller gain C00577 or by increasing the reset time C00578.

The field feedforward control reciprocally reduces the magnetising current from the V/f base frequency (C00015) on. The starting point of the reduction can be shifted to low frequencies via the field feedforward control (C00576). Thus, more voltage reserve is available for acceleration processes.

The field feedforward control (C00576) must be specified in %, based on the rated slip of the machine.

Tip!

Generally, the Lenze setting is sufficient for most applications.

- We recommend to increase the field feedforward control for applications with very dynamic acceleration processes in the field weakening range.
- Reduce the field feedforward control for very slow applications if necessary.

Adaptive adaptation of the P component of the field and speed controllers

In the field weakening range, the properties of the drive change due to the reduction of the magnetisation and the voltage limitation of the inverter. To be able to continue to provide a stable, well dampened drive behaviour, the servo control has an automatic adaptation of the P component of the field weakening controller and speed controller.

In the Lenze setting, this function is activated. Depending on the application, this function can be deactivated via C00079/2.

Parameters	Info	Lenze setting
C00079/2	SC: Adaptive field weakening controller	1: On

Tip!

Lenze recommends to always activate the adaptation of the field weakening and speed controller.

Additional limitation of the I component of the speed controller when the voltage limit is reached (anti-wind-up)

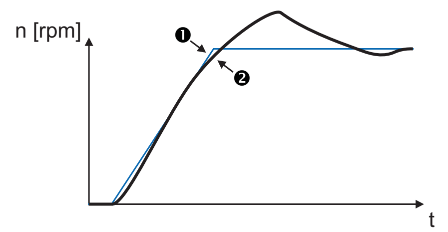
In dynamic acceleration processes in the field weakening range, the acceleration torque is limited due to the limitation of the maximum output voltage of the inverter. Acceleration ramps that are set too high (C00012) may lead to undesired charging of the integrator of the speed controller, which result in very strong overshoots of the actual speed value when the setpoint is reached.

The so-called anti-wind-up function serves to intelligently limit the integrator part in dynamic acceleration processes in the field weakening range so that an overshoot of the actual speed value can be prevented when the setpoint speed is reached.

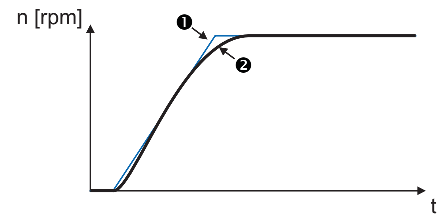
This function is deactivated in the Lenze setting because the robustness of the drive may decrease in the field weakening range. Possible consequences are slight speed variations. If required by the application, the function can be activated via C00079/3.

Parameters	Info	Lenze setting
C00079/3	SC: n-Ctrl Anti-Wind-Up	0: Off

Anti-wind-up" off (without limitation):



Anti-wind-up" on (with limitation):



❶ Speed setpoint
❷ Actual speed value

[5-20] Typical signal characteristics for switched-on/off anti-wind-up function

Tip!

Lenze recommend to activate the additional limitation of the speed controller's I component on reaching the voltage limit (C00079 = 1) if dynamic acceleration processes in the field weakening range are wanted.

- The function should be deactivated again if slight speed variations occur in the field weakening range.
- If this function is activated, the actual speed value may not follow the speed setpoint correctly in the field weakening range.

5.9.4.9 Feedforward control of the current controller

Note!

Function only possible with servo control (SC) for asynchronous motors!

In C00079/1, the feedforward control of the current controller can be switched on. The feedforward control serves to unload the current controller so that the current control loop is able to operate more stable and dynamically. The feedforward control can contribute to dampen vibrations at very high speeds.

Before the feedforward control is activated in C00079/1, we recommend the execution of the motor parameter identification in order to get an optimum result with the feedforward control. ▸ Automatic motor data identification (📖 151)

Note!

If the data of the motor equivalent circuit (motor magnetising inductance, motor stator leakage inductance, motor stator resistance, motor rotor resistance) are not known, the feedforward control of the current controller must not be switched on as otherwise it might cause instabilities in the control loop!

Up to and including version 13.xx.xx the following applies:

The feedforward control of the current controller may only be switched on in C00079/1 if the motor magnetising inductance set in C00092 is not more than 4 % higher than the real magnetising inductance of the connected motor detected by an identification process. Otherwise, instabilities may occur at high speeds in the control loop and the motor can accelerate in an uncontrolled manner up to maximum speed.

- The magnetising inductance of a motor has a tolerance of more than 4 % so that this problem might occur when the motor parameters are loaded from the »Engineer« motor catalogue and the feedforward control of the current controller is activated.
- If the feedforward control is to be used and instabilities occur: Reduce the motor magnetising inductance (C00092) until a stable behaviour occurs or carry out a motor parameter identification.

From version 14.00.00 the following applies:

The feedforward control of the current controller operates properly within the tolerances of the motor. It is sufficient to select the motor from the »Engineer« motor catalog or enter the data of the motor equivalent circuit manually.

5.10 Parameterisable additional functions

5.10.1 Selection of switching frequency

The switching frequency of the inverter that can be selected in C00018 influences the smooth running performance and the noise generation in the connected motor as well as the power losses in the inverter.

The lower the switching frequency the higher the concentricity factor, the smaller the losses, and the higher the noise generation.

Stop!

If operated at a switching frequency of 16 kHz, the inverter output current must not exceed the current limit values specified in the technical data! (See "Rated data" section of the hardware manual.)

Note!

- Operate mid-frequency motors only at a switching frequency of 8 kHz or 16 kHz (var./drive-opt.).
- If operated at a switching frequency of 16 kHz, the Ixt evaluation (C00064) is considered including the required derating to 67 % of the rated device current at switching frequencies of 2.4 and 8 kHz.
- Ensure that the value does not fall below the permissible switching frequency,
  - if the motor is not approved for the setting switching frequency
    - or -
  - if a sinusoidal filter has been connected between the output of the inverter and the motor which is not approved for the setting switching frequency.
    (see section "Preventing a decrease of the switching frequency")

Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value	Unit
C00018	Switching frequency	2: "8 kHz var./drive-opt."	
C00144	Switching frequency reduction (temp.)	1: On	
C00725	Current switching frequency	-	
C00910/1	Max. pos. output frequency	599	Hz
C00910/2	Max. neg. output frequency	599	Hz
C02864/1 Bit 4:	MCTRL: Special settings 2 Option "Motor ident: Switching frequency 8 kHz" (from version 14.00.00)	0x0000 (Off)	
Greyed out = display parameter			

Settable switching frequencies

Selection in C00018			
1	4 kHz var./drive-optimised	21	8 kHz var./drive-opt./4 kHz min
2	**8 kHz var./drive-optimised**	22	16 kHz var./drive-opt./4 kHz min
3	16 kHz var./drive-optimised	23	16 kHz var./drive-opt./8 kHz min
5	2 kHz constant/drive-optimised	31	8 kHz var./min. Pv/4 kHz min
6	4 kHz constant/drive-optimised	32	16 kHz var./min. Pv/4 kHz min
7	8 kHz constant/drive-optimised	33	16 kHz var./min. Pv/8 kHz min
8	16 kHz constant/drive-optimised	Abbreviations used: • "var.": Adaptation of the switching frequency depending on the current • "drive-opt.": drive-optimised modulation ("sine/delta modulation") • "fixed": fixed switching frequencies • "min. Pv": additional reduction of power loss	
11	4 kHz var./min. Pv		
12	8 kHz var./min. Pv		
13	16 kHz var./min. Pv		
15	2 kHz constant/min. Pv		
16	4 kHz constant/min. Pv		
17	8 kHz constant/min. Pv		
18	16 kHz constant/min. Pv		

Tip!

The Lenze setting C00018 = "2: 8 kHz var.drive-opt." is the optimal value for standard applications.

Preventing a decrease of the switching frequency

The value must not fall below the permissible switching frequency if the motor is not approved for the setting switching frequency or if a sinusoidal filter is connected between the output of the inverter and the motor that is not approved for the setting switching frequency.

In order to prevent a decrease of the switching frequency, the following settings have to be made:

1. The temporary switching frequency reduction has to be activated by the setting C00144 = "0: Off".
2. The switching frequency has to be set in C00018 so that the value is not fallen below the permissible switching frequency.
3. If a sinusoidal filter is available, only a drive-optimised switching frequency ("drive-opt.") is permissible.

Example: A sinusoidal filter is connected between the output of the inverter and the motor which only is to be operated with a minimum switching frequency of 8 kHz.

In this case, only the following settings in C00018 are permissible:

Selection in C00018	
7	8 kHz constant/drive-optimised
8	16 kHz constant/drive-optimised
23	16 kHz var./drive-opt./8 kHz min

Note!

Switching frequency for the motor parameter identification

Up to and including version 13.xx.xx, the motor parameter identification is executed with a switching frequency of 4 kHz.

- Thus, a motor parameter identification is not permissible for the previously mentioned example as the sinusoidal filter may only be operated with a minimum switching frequency of 8 kHz!

From version 14.00.00, the motor parameter identification can also be executed with a switching frequency of 8 kHz instead of 4 kHz. For this purpose, the option "Motor ident.: Switching frequency 8 kHz" (Bit 4 = "1") has to be set in C02864/1.

- With this setting, a motor parameter identification would be permissible for the previously mentioned example.

Generally, a switching frequency of 4 kHz is recommended for the motor parameter identification as it serves to obtain the most accurate results.

▸ Automatic motor data identification (📖 151)

Lowering the switching frequency due to high heatsink temperatures

Exceeding the maximally permissible heatsink temperature would lead to an inhibited drive due to the "Overtemperature" error and a torquelessly coasting motor. Therefore, if the Lenze setting is selected, the switching frequency is reduced to the next frequency below when the heatsink temperature has risen to 5 °C below the maximally permissible temperature. After the heatsink has cooled down, the inverter automatically switches to the next frequency above until the set switching frequency is reached.

Switching frequency reduction due to high heatsink temperature can be deactivated via C00144. If the switching frequency reduction is deactivated, the "oH1: Heatsink overtemperature" error message will be issued when the maximally permissible heatsink temperature is reached. An "Error" response is the result and the motor is coasting.

Parameters	Info	Lenze setting
C00144	Switching frequency reduction (temp.)	1: On

Lowering of the switching frequency depending on the output current

"Variable" switching frequencies can be selected for the inverter in C00018, where the inverter automatically lowers the switching frequency depending on the inverter output current. The modulation mode will not be changed.

The changeover thresholds are listed in the **8400 hardware manual** in the rated data. The hardware manual has been stored in electronic form on the data carrier supplied with the 8400 inverter.

When a "fixed" switching frequency is selected, no switching frequency changeover takes place. In case of fixed frequencies, the inverter output current is limited to the permissible value of the corresponding switching frequency. In case of larger load impulses, the overcurrent interruption may be activated, to which the inverter responds with "Error".

Limiting the maximum output frequency

Note!

If the servo control mode (SC) is selected, the drive assumes the 'maximum current limitation' state if the maximum output frequency (C00910) has been reached.

▸ Defining current and speed limits (📖 164)

The maximum output frequency (C00910) of the inverter is not limited depending on the switching frequency. Therefore, adapt the maximum output frequency according to our recommendation:

$$\text{Maximum output frequency} \leq \frac{1}{8}\text{Switching frequency}$$

- At a switching frequency of 4 kHz, for instance, 500 Hz for the maximum output frequency should not be exceeded.

Carry out further measures:

- If required, deactivate the switching frequency changeover by the heatsink temperature via C00144.
- If required, ensure that the changeover threshold of the inverter output current to the next switching frequency below will not be exceeded. If required, select a constant switching frequency in C00018.

Operation at an ambient temperature of 45°C

The inverter is designed so that operation at an ambient temperature of 45° C without derating is permissible at a switching frequency of 4 kHz.

5.10.2 Operation with increased rated power

Under the operating conditions described here and under continuous operation, the inverter can be operated with a higher power motor (increased rated power). The remaining overload capacity of the drive system (for 60 s/3 s) is reduced accordingly to approx. 120 %/160 %.

Typical applications stand out due to low dynamic requirements, e.g. pumps and fans, general horizontal materials handling technology and line drives.

The inverters which can be operated at an increased rated power are listed in the **8400 hardware manual** in the rated data. The hardware manual is stored in electronic form on the data carrier which is supplied with the 8400 inverter.

Stop!

Operation at increased rated power is only permitted ...

- with the inverters listed in the **8400 hardware manual** for this type of operation in the stated mains voltage range.
- at switching frequencies of 2 kHz and 4 kHz.
- at a max. ambient temperature of 40 °C.
- with the types of installation stated in the **8400 hardware manual**.
- with the fuses, cable cross-sections, mains chokes, and filters as required in the **8400 hardware manual** for this operation.
- after parameterisation according to the specifications below.

Required parameterisation

Operation at increased rated power requires the following settings to be made particularly for the V/f characteristic control (VFCplus), but also for all other types of control:

Parameters	Info	Required setting	
C00016	VFC: Vmin boost	adapt to motor (reduce)	
C00018	Switching frequency	1: 4 kHz var./drive-opt.	
C00021	Slip compensation	adapt to motor	
C00120	Setting of motor overload (I²xt)	adapt to motor	
C00123	Device utilisat. threshold (Ixt)	120	%
C00173	Mains voltage	see hardware manual → rated data	

All other types of control require the following settings in addition:

Parameters	Info	Required setting
C00022	Imax in motor mode	higher than rated motor current (max. 160 % rated motor current)
C00081	Rated motor power	adapt motor data (see motor nameplate), then carry out identification run ▸ Automatic motor data identification (📖 151)
C00087	Rated motor speed	
C00088	Rated motor current	
C00089	Rated motor frequency	
C00090	Rated motor voltage	
C00091	Motor cos φ	

5.10.3 Current-dependent stator leakage inductance Ppp(I)

Note!

Function only possible with:
- Servo control (SC) for synchronous motors
- Sensorless control for synchronous motors (SLPSM)

The current controller must be adjusted to the electrical characteristics of the motor stator resistance (C00084) and stator leakage inductance (C00085). In case of modern motors, the stator leakage inductance changes with the height of the current so that a new current controller setting is required for each current height.

When the motor is operated with very low and very high currents (e.g. in *Pick and place* applications), it is not always possible to achieve a satisfactory current controller setting for all operating points. For this purpose, the correction of the stator leakage inductance and current controller parameters is now possible via an adjustable saturation characteristic (17 interpolation points).

Short overview of the relevant parameters:

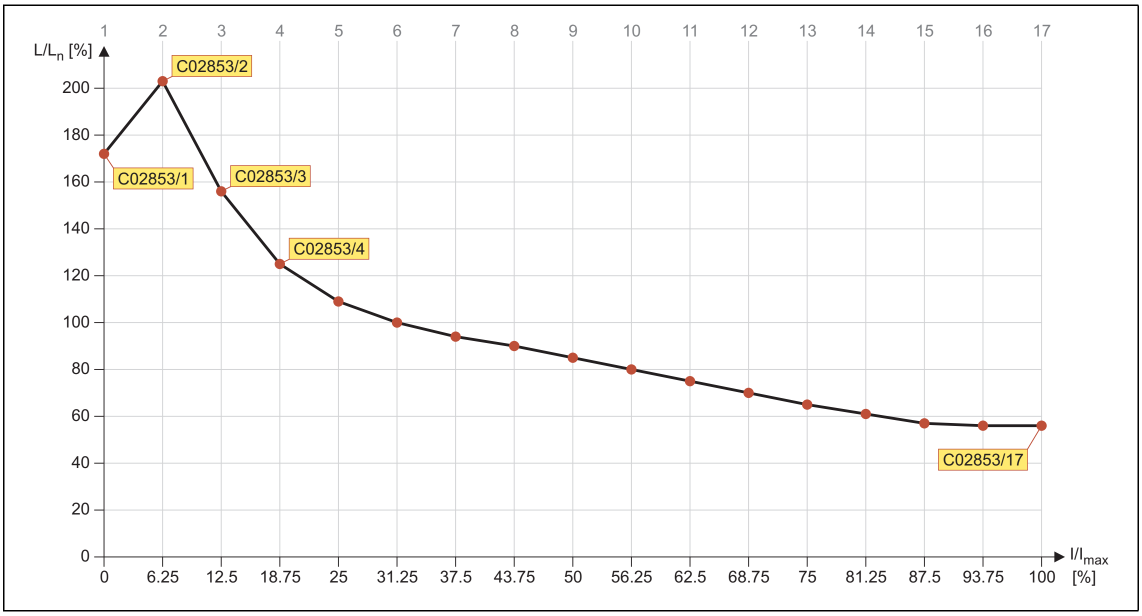
Parameters	Info	Lenze setting	
		Value	Unit
C02853/1...17	PSM: Ppp saturation characteristic (17 interpolation points)	100	%
C02855	PSM: Imax Lss saturation characteristic	3000.0	A
C02859	PSM: Activate Ppp saturation char.	0: Off	

Note!

- The saturation characteristic is not only used for the correction of the current controller but also influences the current controller feedforward control (C00079/1).
- When a Lenze motor is selected from the »Engineer« motor catalogue, the corresponding saturation characteristic is set in C02853/1...17 and – if required – the correction via this saturation characteristic is switched on in C02859.
- For third-party motors: If the current controller becomes unstable with high currents, contact the motor manufacturer to find out whether the stator leakage inductance changes with the current level. If required, the saturation characteristic of this motor must be set in C02853/1...17 and then activated in C02859.

Distribution of the grid points

- The saturation characteristic is represented by 17 interpolation points linearly distributed on the X axis (C02853/1...17).
- Interpolation point 17 represents 100 % of the maximum motor current in the process (C02855).
- The following diagram shows the saturation characteristic stored in the »Engineer« motor catalogue for the Lenze motor "MCS12H15" as an example:



[5-21] Saturation characteristic: Inductance referring to the inductance for rated current

5.10.4 Flying restart function

The flying restart circuit for asynchronous motors uses a simple motor model which requires knowledge of the motor stator resistance RS and the rated motor current.

Note!

- Currently, the flying restart circuit is only available for asynchronous motors. (A flying restart circuit for synchronous motors is in preparation.)
- For a correct functioning of the flying restart circuit, we recommend to perform a parameter identification first. ▶ Automatic motor data identification (📖 151)
- The flying restart function works safely and reliably for drives with great centrifugal masses.
- Do not use the flying restart function if several motors with different centrifugal masses are connected to an inverter.
- After the controller is enabled, the motor can start for a short time or reverse when machines with low friction and low mass inertia are used.
- The flying restart function serves to identify max. field frequencies up to ±200 Hz.
- When power-adapted standard asynchronous motors are used (rated motor power approximately corresponds to the rated inverter power), a motor parameter identification is not required.
- On drive systems with feedback, you do not need to use the flying restart function because the synchronisation to the speed detected by the feedback is always carried out in a jerk-free manner.
- Especially in case of very high power, very high mass inertia and a mains voltage higher than 440 V, an overvoltage in the DC bus may be caused temporarily. Using a brake resistor may avoid this behaviour.
  - From version 15.00.00, frequency can be set as a remedy in C00989 which is added to the found flying restart frequency.
    Here, the setting in C00989 has to be increased in small steps up to double the slip frequency until the behaviour does not occur anymore.

Tip!

In association with the flying restart function, we recommend information provided in this documentation on the following topic:
▶ Automatic DC-injection braking (Auto-DCB) (📖 284)

General information

This function serves to activate a mode which is used to "catch" a coasting motor during operation without speed feedback. This means that the synchronicity between inverter and motor is to be adjusted in such a way that a jerk-free transition to the rotating machines is achieved in the instant of connection.

The inverter determines the synchronicity by identifying the synchronous field frequency.

Duration

The "catching" process is completed after approx. 0.5 ... 1.5 seconds. The duration is influenced by the starting value. If the field frequency is not known, we recommend a fixed starting value of 10 Hz (or -10 Hz with systems rotating in negative direction).

Short overview of the relevant parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C00989	Restart on the fly: Flying restart frequency fd_add (from version 15.00.00)	0.00	Hz
C00990	Flying restart fct.: Activate	Off	
C00991	Flying restart fct.: Process	-n...+n | Start: +10 Hz	
C00992	Flying restart: Start frequency	5	Hz
C00993	Flying restart: Integration time	300	ms
C00994	Flying restart: Current	25.00	%

How to parameterise the flying restart function:

1. Activate the flying restart circuit by selecting "1: On" in C00990.
   - Every time the controller is enabled, a synchronisation to the rotating or standing drive is carried out.

   When the Lenze setting is used, most applications do not require additional inverter settings.

   If additional settings are necessary, proceed as follows:
2. Define the process and hence the speed range/rotational frequency range in C00991 which is to be examined by the flying restart circuit:
   - positive speed range (n $\geq$ 0 rpm)
   - negative speed range (n $\leq$ 0 rpm)
   - entire speed range
3. Define the starting frequency.

   The starting frequency which defines the starting point of the flying restart function is 10 or -10 Hz for processes 0 ... 3 and has been pre-optimised for standard motors.

   If process 4 is selected in C00991, an arbitrary starting frequency can be defined via C00992. This is especially recommended for motors with higher rated frequencies.
   - We recommend to define a starting frequency of approximately 20 % of the rated motor frequency to enable a safe and fast connection to standing drive systems.
   - For systems with a known search speed (e.g. torque-controlled drive systems which are to synchronise to a defined speed) the starting value can be adapted to reduce the flying restart time.
4. Set the flying restart current in C00994.

We recommend setting a flying restart current of 10 % ... 25 % of the rated motor current.

- During a flying restart process, a current is injected into the motor to identify the speed.
- Reducing the current causes a reduction of the motor torque during the flying restart process. A short-time starting action or reversing of the motor is prevented with low flying restart currents.
- An increase of the current improves the robustness of the flying restart function.
- At too high mass inertias and high speeds at the same time, the flying restart circuit may cause an overvoltage in the DC bus ("oU") if no brake resistor is connected. In this case, reduce the flying restart current.

Tip!

Use of motors with higher rated frequencies

For trouble-free operation, we recommend to manually enter a starting frequency of 20 % of the rated motor frequency in C00992 as well as to accelerate the flying restart process (see above) and to use a lower flying restart current (10 % of the rated motor current) if motors with higher rated frequencies are used.

Optimisation of the flying restart time

The duration of the flying restart process can be influenced via the setting of the integration time (C00993). A reduction of the integration time causes the flying restart function to accelerate and thus a reduced flying restart time.

- We recommend not to change the Lenze setting of the integration time.
- When special motor are used (e.g. multi-pole motors or ASM servo motors), a reduced integration time may improve the flying restart behaviour.

Optimising the current controller if the behaviour is unstable

During the execution of flying restart function, peak currents/torques are avoided by controlling the current amplitude.

Gain (C00075) and reset time (C00076) of the current controller can be adapted to improve the jerk-free/torque-free connection of the inverter to the supply of the rotating motor.

- We recommend not to change the Lenze setting of the current controller.
- If the behaviour of the current controller is unstable, gain and reset time can be calculated as per the following formulae:

$$V_p = \frac{L_{ss}[H]}{T_E[s]}$$

$$T_i = \frac{L_{ss}[H]}{R_s[\Omega]}$$

V_p = Current controller gain (C00075)
T_i = Current controller reset time (C00076)
L_{ss} = Motor stator leakage inductance (C00085)
R_s = Motor stator resistance (C00084)
T_E = Equivalent time constant (= 500 µs)

[5-22] Formulae for the calculation of the gain and reset time of the current controller

5.10.5 DC-injection braking

Note!

The DC-injection braking function (manually or automatically) is only possible for the following motor control modes:

- V/f characteristic control (VFCplus) (📖 167)
- V/f characteristic control - energy-saving (VFCplusEco) (📖 186)
- V/f control (VFCplus + encoder) (📖 196)
- Sensorless vector control (SLVC) (📖 204)

Danger!

Holding braking is not possible when this braking mode is used!

For low-wear control of a holding brake, use the basic function "Holding brake control". (📖 701)

DC-injection braking allows the drive to be quickly braked to a standstill without the need to use an external brake resistor.

- The braking current is set in C00036.
- The maximum braking torque to be generated by the DC braking current is approx. 20 ... 30 % of the rated motor torque. It is lower than that for braking in generator mode with an external brake resistor.
- The DC-injection braking can be activated manually via a control input of the application or automatically ("Auto-DCB"). The two different modes are described in the following subchapters.

Tip!

DC-injection braking has the advantage that it is possible to influence the braking time by changing the motor current or the braking torque..

The automatic DC-injection braking (Auto-DCB) improves the starting performance of the motor in case of V/f operation without speed feedback.

Short overview of the relevant parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C00019	Auto-DCB: Threshold • Operating threshold for activating DC-injection braking	3	rpm
C00036	DC braking: Current • Braking current in [%] based on rated device current (C00098)	50	%
C00106	Auto-DCB: Hold time	0.500	s
C00107	DC braking: Hold time • When "999.000 s" is set, the hold time is infinite.	999.000	s

5.10.5.1 Manual DC-injection braking (DCB)

DC-injection braking can be activated manually for the two technology applications "speed actuating drive" and "switch-off positioning" by connecting the *bSetDCBrake* input of the LA_NCtrl or LA_SwitchPos application block to a digital signal source (e.g. via a digital signal source *bCtrl1_B3* of the LP_CanIn1 port block).

- For HIGH-active inputs, DC-injection braking is active as long as the signal is at HIGH level.
- After the hold time (C00107) has expired, the inverter set the pulse inhibit.

Short overview of the relevant configuration parameters:

Parameters	Info	Lenze setting
C00701/4	LA_NCtrl: bSetDCBrake • "Speed actuating drive" application: Selection of the signal source for manually activating DC-injection braking.	Dependent on the selected control mode
C00761/4	LA_SwitchPos: bSetDCBrake • "Switch-off positioning" application: Selection of the signal source for manually activating DC-injection braking.	Dependent on the selected control mode

5.10.5.2 Automatic DC-injection braking (Auto-DCB)

"Automatic DC-injection braking" (referred to in the following as "auto DCB") can be used if there is a requirement that the drive be isolated from the supply at n ≈ 0.

Note!

Deactivate automatic DC-injection braking when a holding brake is used!

- For this purpose, go to C00019 and set the auto DCB threshold to "0".
- Background: Controller inhibit is already activated by the Holding brake control. (📖 701)

Function

For understanding the auto DCB function, it is necessary to distinguish between three different types of operation:

A. The drive has been enabled and, in the course of operation, the speed setpoint falls below the auto DCB threshold.
   - In case of V/f operation without speed feedback, a braking current (C00036) is injected. After the auto DCB hold time (C00106) has expired, the motor is deenergised via the auto DCB function, i.e. pulse inhibit is set.
   - In case of V/f operation with speed feedback, the motor is deenergised via the auto DCB function after the auto DCB hold time (C00106) has expired, i.e. pulse inhibit is set. The braking current which can be parameterised in C00036 does not have any effect during operation with speed feedback.

B. When the controller is enabled, the drive is at standstill (n = 0).
   If the enabled drive is to start, the speed setpoint passed via the acceleration ramp must exceed the auto DCB threshold (C00019). Below this threshold, the motor will not be energised.

C. When the controller is enabled, the motor (still) rotates at a speed which is above the auto DCB threshold. If the speed setpoint reached via the acceleration ramp exceeds the auto DCB threshold (C00019), the motor will be energised and the following action will take place:
   - During V/f operation without speed feedback, the drive is "caught". ▶ Flying restart function (📖 280)
   - During V/f operation with speed feedback, the drive synchronises to the current actual speed value.

Tip!

We recommend to deactivate the auto DCB function during V/f operation with speed feedback via a setting of C00019 = 0.

Auto DCB function during V/f operation with speed feedback

Stop!

If the DC-injection braking operation is too long and the braking current or braking voltage is too high, the connected motor may overheat!

If you want to use the auto DCB function contrary to our recommendation (see above), the auto DCB threshold must not fall below the following values depending on the number of encoder increments:

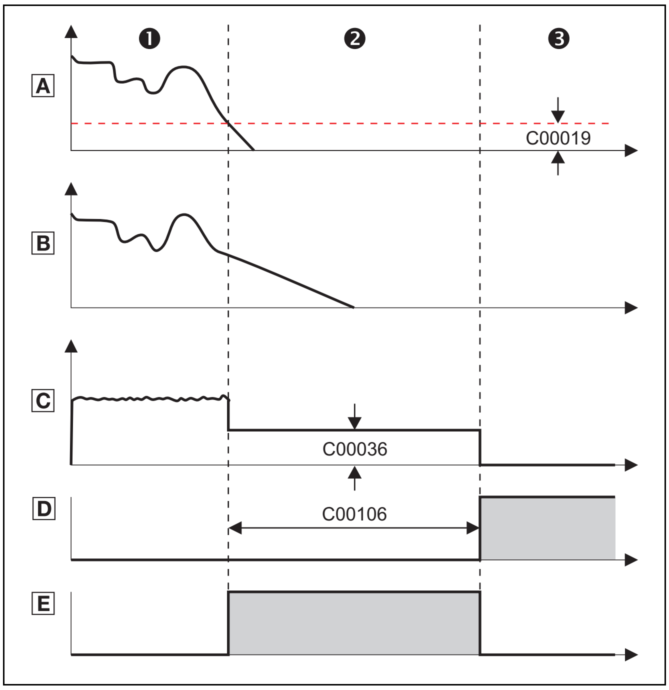
Number of encoder increments (C00420)	Auto-DCB threshold (C00019)
8	16
16	8
32	4
64	2
> 128	No restrictions

Setting the auto-DCB function

How to set the automatic DC-injection braking:

1. Set a hold time in C00106 > 0 s.
   - Automatic DC-injection braking is active for the time set.
   - In case of V/f operation without speed feedback, the braking set in C00036 is injected.
   - After the set hold time has expired, the inverter sets pulse inhibit.
2. Set the operating threshold in C00019.
   - The operating threshold can serve to set a dead band in the setpoint. If DC-injection braking is not to be active then, C00106 must be set to a value of "0".

Explanation of the automatic DC-injection braking function by means of two examples



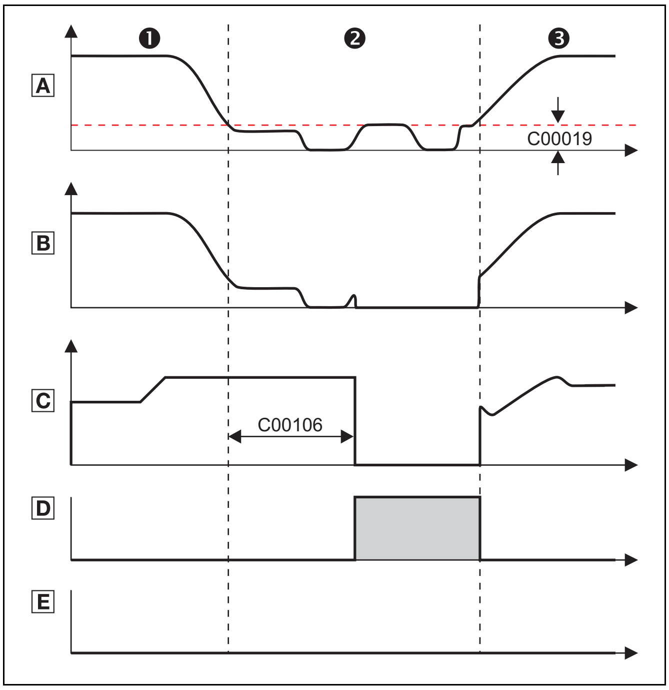
❶ The motor rotates at a specified speed. The current adjusts itself to the load, see C.

❷ The DC braking current set in C00036 is injected.

❸ After the hold time (C00106) has expired, a pulse inhibit is set.

A Speed setpoint
B Actual speed value of the motor
C Controller output current
D Pulse inhibit
E DC-injection braking is active

[5-23] Example 1: Auto-GSB signal characteristic during V/f operation without speed feedback



❶ The motor rotates at a specified speed. The current adjusts itself to the load, see C.

❷ The current sets itself load-dependent until the hold time (C00106) has expired.

❸ The actual speed value of the motor follows the speed setpoint. The resulting current depends on the load.

A Speed setpoint
B Actual speed value of the motor
C Controller output current
D Pulse inhibit
E DC-injection braking is active

[5-24] Example 2: Auto-GSB signal characteristic during V/f operation with speed feedback

5.10.6 Slip compensation

Note!

Slip compensation is only active with the following motor control modes:

- V/f characteristic control (VFCplus) (📖 167)
- Sensorless vector control (SLVC) (📖 204)

Under load, the speed of an asynchronous motor decreases. This load-dependent speed drop is called slip. The slip can partly be compensated for by the setting in C00021.

Parameters	Info	Lenze setting	
		Value	Unit
C00021	Slip compensation	2.11	%

- The setting of C00021 can be done automatically in the course of motor parameter identification. ▶ Automatic motor data identification (📖 151)
- The setting must be made manually if the motor parameter identification cannot be called up.

How to set the slip compensation manually:

1. Set rated motor current (C00088) and rated motor frequency (C00089).
2. Calculate the slip compensation according to motor nameplate data:

$$s = \frac{n_{rsyn} - n_r}{n_{rsyn}} \cdot 100\%$$

$$n_{rsyn} = \frac{f_r \cdot 60}{p}$$

s Slip constant (C00021) [%]

n_{rsyn} Synchronous motor speed [rpm]

n_r Rated motor speed according to the motor nameplate [rpm]

f_r Rated motor frequency according to the motor nameplate [Hz]

p Number of motor pole pairs (1, 2, 3 ...)

3. Transfer the calculated slip constant s to C00021.
4. Correct the setting in C00021 while the drive is running until the load-dependent speed drop does not occur anymore between idling and maximum load of the motor in the desired speed range.

Tip!

The following guide value applies to a correctly set slip compensation:

- Deviation from the rated motor speed ≤ 1% for the speed range of 10 % ... 100 % of the rated motor speed and loads ≤ rated motor torque.
- Greater deviations are possible in the field weakening range.
- If C00021 is set too high, the drive may get unstable.
- Negative slip (C00021 < 0) with V/f characteristic control results in "smoother" drive behaviour at heavy load impulses or applications requiring a significant speed drop under load.

5.10.7 Oscillation damping

Mechanical oscillations are undesirable effects in every process and they may have an adverse effect on the single system components and/or the production output.

Mechanical oscillations in the form of speed oscillations are suppressed by the oscillation damping function.

Mechanical oscillations may occur:

- In the voltage range (output voltage is lower than max. voltage)
  - Here, the oscillations occur in no-load operation.
  - Here, speeds of 40 ... 80 % of the rated speed are typical.
  - See subchapter "Oscillation damping voltage range". (📖 289)
- In the field weakening range (output voltage has reached maximum voltage)
  - Here, the oscillations occur in no-load operation and with load.
  - Here, speeds higher than the rated speed are typical, especially when the output frequency is close to the mains frequency.
  - See subchapter "Oscillation damping in the field weakening range". (📖 290)

Note!

With servo control (SC), the Oscillation damping voltage range has no influence.

Mechanical natural frequencies can be suppressed or at least dampened in the speed control loop of the servo control by means of a current setpoint filter. ▶ Setting the current setpoint filter (band-stop filter) (📖 259)

Oscillation dampening, especially in the field weakening range, is also possible by activating the current controller feedforward control (C00079/1).

5.10.7.1 Oscillation damping voltage range

The oscillation damping voltage range is successfully used with

- unloaded motors (no-load oscillations)
- motors whose rated power deviates from the rated power of the inverter.
  - e.g. during operation at high switching frequency including the power derating involved.
- operation with higher-pole motors
- operation with special motors
- compensation of resonance in the drive
  - At an output frequency of approx. 20 ... 40 Hz, some asynchronous motors can show resonance which causes current and speed variations and thus destabilise the running operation.

Parameters	Info	Lenze setting	
		Value	Unit
C00234	Impact of oscillation damping	5.00	%
C00235	Oscillation damping filter time	32	ms

Note!

Compensate the resonance during operation with feedback (closed loop, feedback of n_{act}) via the parameters of the slip regulator.

▶ Parameterising the slip regulator (📖 200)

How to eliminate speed oscillations in no-load operation at speeds with 40 ... 80 % of the rated speed:

1. Approach the area where the speed oscillations occur.
2. Reduce the speed oscillations by changing C00234 step by step (increment 1 %).
   - The filter time oscillation damping (C00235) should not be changed.
3. These can be indicators for smooth running:
   - Constant motor current characteristic
   - Reduction of the mechanical oscillations in the bearing seat

5.10.7.2 Oscillation damping in the field weakening range

When the max. possible output voltage (full modulation) has been reached, a voltage dip in the DC bus causes a voltage fluctuation in the motor. With load and during no-load operation this voltage fluctuation can cause mechanical oscillations.

The "oscillation damping field weakening" adjustable in C00236 serves to limit the maximum output voltage. This can be used to always compensate voltage dips in the DC bus to the output voltage (constant output voltage). This serves to prevent mechanical oscillations due to these voltage dips.

Parameters	Info	Lenze setting	
		Value	Unit
C00236	Oscillation damping field weakening • Setting "0" ≡ 100 % output voltage can be reached	14	

- With the Lenze setting of C00236 the limitation of the output voltage is set so that voltage dips in the DC bus in the output voltage for the single-phase and three-phase devices can largely be compensated so that no speed oscillations may be expected. Thus, an adaptation of C00236 is not required in the majority of cases.
- Maximum output voltage to be reached with Lenze setting of C00236:
  - Single-phase devices: 98.2 %
  - Three-phase devices: 99.7 %

Note!

The limitation of the output voltage via C00236 in the extreme field weakening range (high speeds) causes a reduction of the max. possible output torque (stalling torque).

- If the output torque to be reached in the extreme field weakening range is not sufficient (motor is stalling too early), reduce the setting in C00236.

With servo control (SC), the Lenze setting of C00236 should not be reduced. Otherwise the field weakening control could not work optimally anymore (the behaviour at high speeds may get worse). When servo control (SC) has been selected, the oscillation damping in C00236 should never be set to "0".

How to eliminate speed oscillations in the field weakening range:

1. Approach the area where the speed oscillations occur.
2. Reduce the speed oscillations by changing C00236 step by step (increment 1).
3. These can be indicators for smooth running:
   - Constant motor current characteristic
   - Reduction of the mechanical oscillations in the bearing seat

5.10.8 Phase sequence reversal for correcting misconnected UVW motor phases

Note!

Up to and including version 02.00.00, this function may only be activated for the following motor control types:

- V/f characteristic control (VFCplus) (167)
- V/f characteristic control - energy-saving (VFCplusEco) (186)

For all other motor control types, this function must not be activated because the set control mode would not work in that case!

From version 12.00.00, this function may be activated for all motor control types with the exception of the ones for synchronous motor.

The activation of this function does not affect the control types for synchronous motor since these control types require an in-phase connection of the synchronous motor.

If the motor phases are misconnected at the inverter output (e.g. phase u takes the place of phase v), the motor will rotate in the wrong direction.

To correct such misconnected motor phases, the rotating field of the inverter's output can be reversed by selecting "1: Inverted" in C00905. In this case, a phase will be reversed at the output of the inverter.

This function does not have any effect on setpoints and actual values, i.e. the polarity of the speed setpoint/actual speed value, actual torque, output frequency, and AngleOffset do not change.

Tip!

Cases of application for this function:

- Phase sequence reversal in case of misconnected motor phases.
- Setting of the correctly signed direction of rotation for inversely mounted motors.

5.10.9 Field weakening for synchronous motors

This function extension is available from version 02.00.00!

Note!

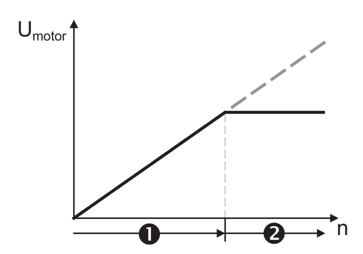
Function only possible with:
- Servo control (SC)
- Sensorless control for synchronous motors (SLPSM) (from version 10.00.00)

In the Lenze setting, the field weakening for synchronous motors is activated in C00079/4.
- If a high energy efficiency is required, keep the field weakening switched off or restrict the field weakening operation via C00938.

Stop!

In the field weakening operation, a current is injected into the synchronous motor even in idle state which can rise to maximum current (C00022).

Ensure that this no-load current does not cause the motor to be heated impermissibly!
- The use of a temperature feedback via PT1000 or KTY is recommended. ▶ Motor temperature monitoring (PT1000 or KTY) (📖 395)



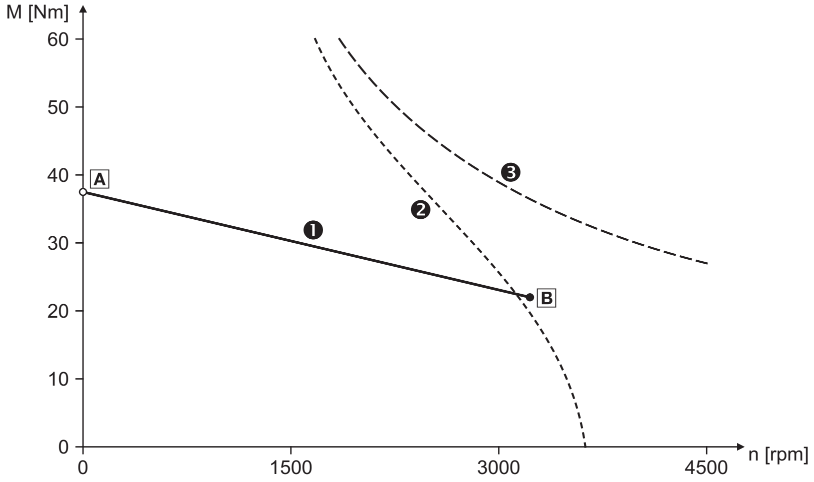
❶ Armature control range
❷ Field weakening range

- When field weakening is switched on, the motor magnetising current is increased from 0 A to the maximally effective magnetising current via an internal control loop when the voltage limit is reached.
- As a result, a higher speed can be reached at the same motor voltage or DC-bus voltage.

[5-25] Voltage/speed characteristic with switched-on field weakening

$$n_{max} = n_{nenn\_mot} \cdot \frac{800\,\mathrm{V}}{\sqrt{2} \cdot U_{nenn\_mot}}$$

[5-26] Calculation of the maximally reachable speed with switched-on field weakening



A Standstill torque
B Rated torque
❶ Characteristic for continuous operation
❷ Limit characteristic at a mains voltage of 400 V without field weakening
❸ Limit characteristic with field weakening

[5-27] Speed/torque characteristics of a synchronous servo motor with field weakening

Short overview of the relevant parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C00079/4	Field weakening	1: On	
C00938	Limitation of maximally effective field-producing motor current • With regard to rated motor current (C00088)	30	%
C00937/1	Maximally effective field-producing motor current	-	A
Greyed out = display parameter			

- The maximally effective field-producing motor current is calculated based on the motor data set in C00085, C00089 and C00098. Then, the value is internally limited to 98 % of the set maximum current (C00022 or maximally permissible current for the permanent switching frequency set in C00018).
- C00938 serves to limit the maximally effective field-producing motor current as well.
  - In the Lenze setting, the field weakening for synchronous motors is active (C00079/4). However, the field-producing motor current is limited via C00938 to 30 % of the rated motor current (C00088). Hence, the maximum speed is limited during field weakening operation and, at the same time, the temperature rise of the motor during field weakening operation and no-load operation is also limited.
  - If a higher speed for the field weakening operation is required or the current in the field weakening operation is to be limited (e.g. since no motor temperature detection is available and/or heating in the field weakening operation is to be limited), the value must be increased or reduced accordingly in C00938.

- In C000937/1, the actually used maximally effective field-producing motor current is displayed.
  - With switched-on and active field weakening: 0.00 A ... -x.xx A
  - With sensorless control for synchronous motors (SLPSM), the injected current is displayed in open-loop controlled operation: 0.00 A ... +x.xx A
  - If neither field weakening nor open-loop controlled operation are active,e "0.00 A" is displayed.

Note!

If a Lenze motor is used:

The inverter is automatically parameterised so that field weakening operates optimally and the maximally permissible speed is monitored.

Stop!

If an OEM motor is used:

If pulse inhibit is set in the inverter, the DC bus is loaded with the voltage that corresponds to the current speed of the machine.

Since with switched-on field weakening higher speeds can be achieved at a correspondingly higher rotor voltage of the motor, the DC bus can be loaded to a voltage higher than the set DC-bus voltage in case of pulse inhibit and a currently high motor speed and even exceed the maximally permissible voltage of 800 V!

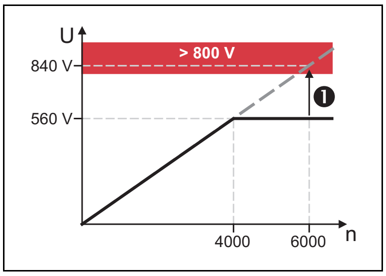
For device protection, either use a brake chopper or parameterise the motor speed monitoring via C00965 in such a way that only a maximum speed is possible which would be also reachable without field weakening with a DC-bus voltage of = 800 V.

▶ Motor speed monitoring (📖 320)

Example: Voltage increase in the DC bus when field weakening is switched off

(For instance by an active setting of the controller inhibit or by tripping a fault or error at high motor speed.)

Field weakening	Speed n	Motor voltage peak value
Switched off	4000 rpm	560 V
	5700 rpm	800 V
	6000 rpm	840 V
Switched on	6000 rpm	560 V



- If pulse inhibit occurs at 6000 rpm and switched-on field weakening, the DC bus is loaded to more than 800 V (❶).
- A speed limitation to 5700 rpm is required since this speed causes a DC-bus voltage of 800 V if field weakening is switched off.

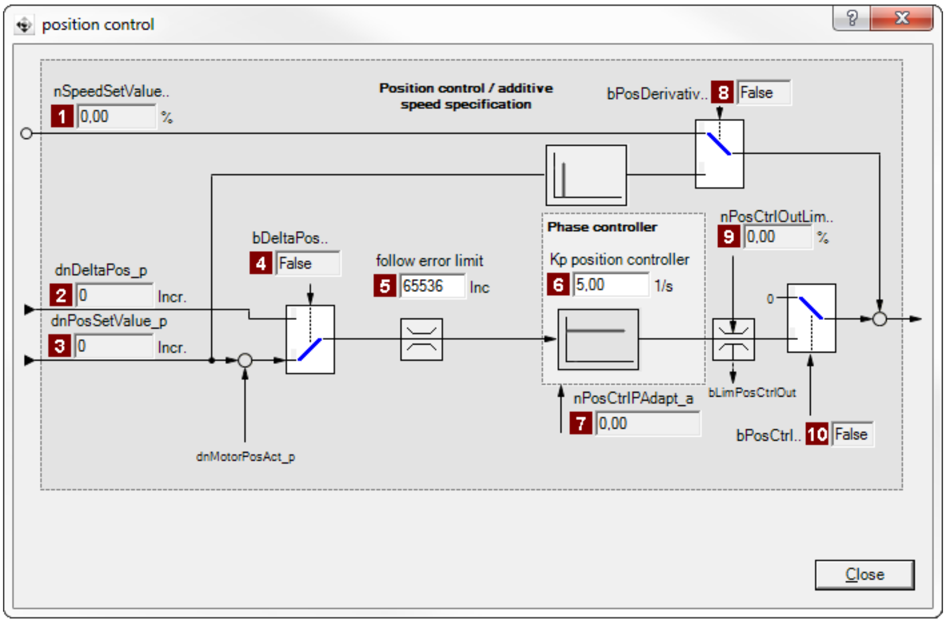
[5-28] Example: Possible DC-bus voltage > 800 V if field weakening gets lost

5.11 Position control/additive speed specification

In the Lenze setting, the position control is only active with TA "Table positioning".

Proceed as follows to open the dialog for parameterising the position control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the **Motor control...** button to change to the *Overview* ➞ *motor control...* dialog level.
4. Click the **Position control** button in the displayed signal flow.



Parameters		Info	
1	C00830/22	MCTRL: nSpeedSetValue_a	Speed setpoint
2	C00834/4	MCTRL: dnDeltaPos_p	Position difference (following error input)
3	C00834/5	MCTRL: dnPosSetValue_p	Absolute position setpoint
4	C00833/35	MCTRL: bDeltaPosOn	TRUE = Position difference is active as setpoint selection
5	C02556/1	Following error limit	Setting of the maximum permissible correcting variable or the maximum permissible following error for the position controller
6	C00254	Kp position controller	Gain for following error compensation
7	C00830/20	MCTRL: nPosCtrlPAdapt_a	Adaptation of the position controller gain
8	C00833/67	MCTRL: bPosDerivativeOn	TRUE = Setpoint for the speed controller is created from the position setpoint
9	C00830/21	MCTRL: nPosCtrlOutLimit_a	Limitation of the position controller output
10	C00833/27	MCTRL: bPosCtrlOn	TRUE = Position/angle control active

5.12 Braking operation/brake energy management

When braking electrical motors, the kinetic energy of the drive train is fed back regeneratively to the DC bus. This energy results in a boost of the DC-bus voltage.

- Several different strategies can serve to avoid DC-bus overvoltages:
  - Use of a brake resistor
  - Stopping of the ramp function generator if brake chopper threshold exceeded (RFG_Stop)
  - Use of the "Inverter motor brake" function
  - Combination of the above named options
- In the case of inverters with a 3-phase supply, the following is also possible:
  - Coupling of the inverters in a DC-bus connection
  - Recovery of regenerative energy with a regenerative module

STOP

Stop!

If the connected brake resistor is smaller than required, the brake chopper can be destroyed!

- Appropriate protective measures are described in subchapter "Avoiding thermal overload of the brake resistor". (📖 305)

If no brake resistor or regenerative module is used, the overvoltage switch-off ("oU") may respond, e.g. in case of short deceleration times during feedback operation. ▶ Error messages of the operating system (📖 743)

Using the integrated brake chopper

We recommend to use the brake chopper (brake transistor) which is integrated into the inverter for the braking operation, regardless of the selected motor mode.

- Connect the required brake resistor to the R_{B1} and R_{B2} terminals of the inverter.
- In C00175, a ramp function generator stop (FB L_NSet_1) can be set for instances when the brake resistor is controlled. This prevents overvoltage deactivation in the case of short deceleration times. ▶ Selecting the response to an increase of the DC-bus voltage (📖 300)

Note!

The brake transistor will be switched off if it remains switched on for a period of 4 seconds.

- If the DC-bus voltage falls under the brake chopper threshold for a short time again, the brake transistor can switch on again for a maximum of 4 seconds, without interruption.
- This protective function is used to prevent the brake chopper from being switched on permanently due to e.g. too high voltages at all times or incorrect interconnection of the *bBrakeChopperOn* signal (from version V12.00.00).
- From version 14.00.00, the limitation of the operating time can be deactivated to max. 4 seconds. For this purpose, set bit 1 to "1" in C02864/1.

The integrated brake chopper is always triggered via the DC-bus voltage with exceedance of the "voltage threshold for braking operation" independent of the device status.

From version 14.00.00, the brake chopper can also be switched off at pulse inhibit or when the "Fault" device status is activated.

- For a switch-off at pulse inhibit, set bit 7 to "1" in C02865/1.
- For a switch-off in the "Fault" device status, set bit 2 to "1" in C02864/1.

DC-bus connection

For a DC-bus connection with other devices, we recommend to connect the regenerative power supply module to terminals +UG and –UG.

Note!

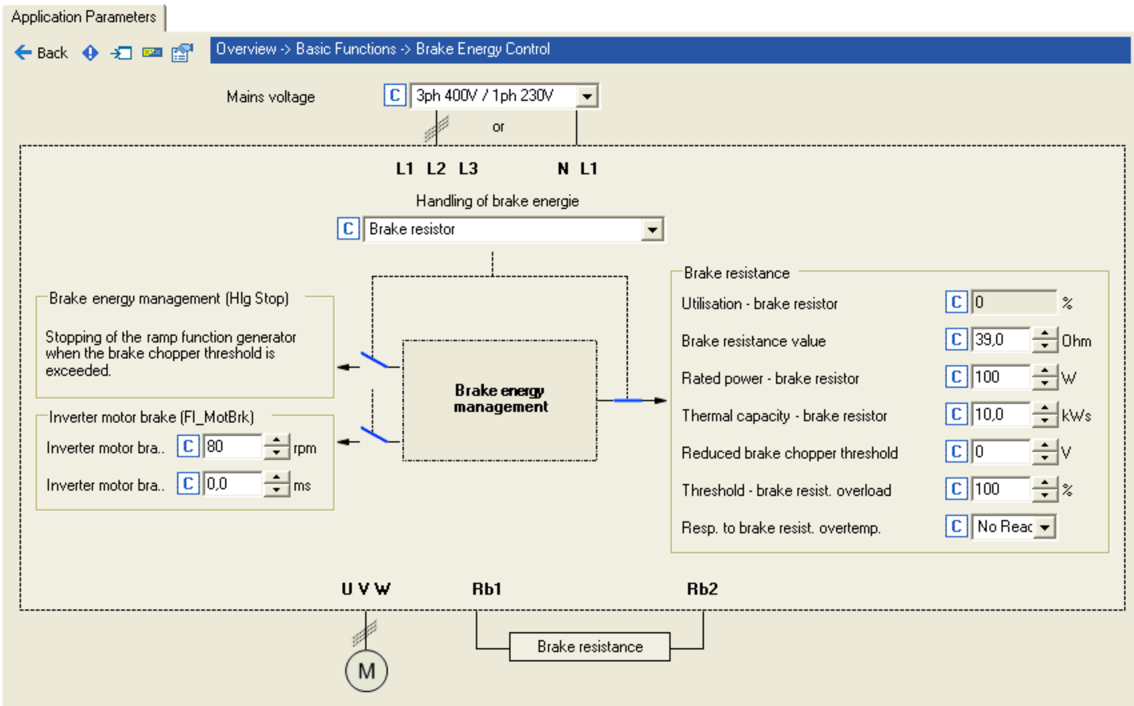
DC-bus system without the use of a regenerative module:

- Up to and including version V02.xx.xx, only one internal brake chopper can be used in the DC-bus system to dissipate regenerative energy.
- From version V12.00.00, all internal brake choppers can be used in the DC-bus system to dissipate regenerative energy ("Master-slave operation"). ▶ Control of multiple internal brake choppers in the DC-bus system (📖 305)

In order to install the regenerative module, follow the notes in the **8400 hardware manual**. The hardware manual has been stored in electronic form on the data carrier supplied with the 8400 inverter.

Proceed as follows to open the dialog for parameterising the brake energy management:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the "basic functions" button.
4. Go to the *Overview* → *basic functions* dialog box and click the **Brake energy management** button.



Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value	Unit
C00173	Mains voltage	3ph 400 V / 1ph 230 V	
C00175	Brake energy management	R_Brake (brake resistance)	
Brake resistor			
C00133	Brake resistor utilisation	-	%
C00129	Brake resistance value	39.0	Ohm
C00130	Rated brake resistor power	100	W
C00131	Thermal capacity - brake resistor	10.0	kWs
C00174	Reduced brake chopper threshold	0	V
C00572	Threshold - brake resist. overload	100	%
C00574	Resp. to brake resist. overtemp.	No response	
Inverter motor brake			
C00987	Inverter motor brake: nAdd	80	rpm
C00988	Inverter motor brake: PT1 filter time	0.0	ms
Greyed out = display parameter			

5.12.1 Setting the voltage source for braking operation

The voltage threshold for braking operation is set via the mains voltage (C00173) and the reduced brake chopper threshold (C00174). When this "brake chopper threshold" is exceeded, the response selected in C00175 takes place in the DC bus. The selected function (e.g. use of a brake resistor) serves to dissipate energy in the DC bus and reduce the DC-bus voltage.

- The "brake chopper threshold" is preset as follows so that it is higher than the specified mains voltage (C00173):

	Mains voltage		Brake chopper threshold	
C00173	**1-phase**	**3-phase**	**1-phase**	**3-phase**
0	1ph 230V	3ph 400V	DC380V	DC725V
1	1ph 230V	3ph 440V	DC380V	DC735V
2	1ph 230V	3ph 480V	DC380V	DC775V
3	1ph 230V	3ph 500V	DC380V	DC790V

- This brake chopper threshold can be reduced by 0 ... 150 V by means of C00174.

STOP **Stop!**

The brake chopper threshold resulting from C00173 and C00174 must not fall below the stabilised DC-bus voltage!

Example:

- A 400 V device has a maximum mains voltage of 420 V AC.
  - Maximum stationary DC-bus voltage: 420 V AC * 1.414 = 594 V DC
  - C00173 has been set with the selection "0" for 400 V AC mains.
- This means that C00174 can be set to a maximum of 131 V DC (725 V DC - 594 V DC).

5.12.2 Selecting the response to an increase of the DC-bus voltage

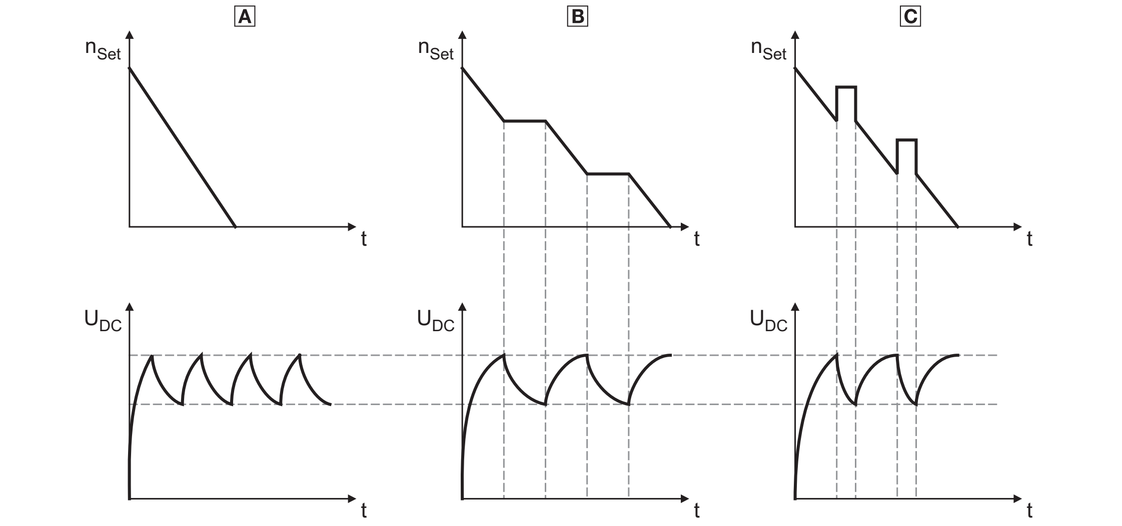
If the brake chopper threshold resulting from C00173 and C00174 is exceeded in the DC bus, the reaction selected in C00175 takes place (use of the brake resistor and/or stop of the ramp function generator and/or inverter-motor brake).

- Optimum following of the actual speed value until the speed setpoint is reached (e.g. the motor is stopped rapidly) is always achieved with the help of a brake resistor.
- Stopping the ramp function generator enables smoother deceleration with lower torque oscillation..
- The inverter-motor brake is available for selection in C00175. This function enables rapid braking without a brake resistor. Torque oscillations can occur due to the traversing dynamics. ▶ Inverter motor brake (📖 302)

Stop!

- The two braking procedures "Stopping of the ramp function generator" and "Inverter motor brake" can only be used for speed-controlled applications without the influence of a position controller!
- When the "inverter motor brake" function is used, the Motor load monitoring (I2xt) is not adapted. If it is braked too frequently, there is a risk of the motor being thermally overloaded or the motor overload monitoring does not work properly!
- The "inverter motor brake" function must not be used with vertical conveyors (hoists) or with active loads!

The way in which the different braking procedures work is demonstrated schematically in the following illustration:



A Use of the brake resistor (C00175 = "0: Brake resistor")
B Stopping of the ramp function generator (C00175 = "1: HlgStop")
C Inverter motor brake (C00175 = "3: FI_MotBrk + RfgStop)

[5-29] Graph of the effective speed setpoint and the DC bus voltage during braking

Tip!

Independent of the selected motor control, all procedures given in C00175 can be used. The actual speed value always follows the speed setpoint in an optimal way when a brake resistor is used.

If it is possible to dispense with exact adherence to the deceleration ramp in simple applications, selection of a braking method without an external brake resistor enables costs to be reduced due to the avoidance of having to use a brake resistor .

With the "inverter motor brake" function, an effective braking torque of 10 ... 20 % of the rated motor torque can be achieved.

A combination of all three braking procedures is also possible, e.g. for emergency braking if the brake resistor fails
(C00175 = "4: Brake resistor + FI_MotBrk + RfgStop").

5.12.2.1 Inverter motor brake

With this braking method, which can be selected as an alternative in C00175, the regenerative energy in the motor is converted as a result of dynamic acceleration/deceleration with down-ramping of the ramp function generator..

Stop!

- This braking method only works without intervention of a position controller in the case of speed-controlled applications!
- When the "inverter motor brake" function is used, the Motor load monitoring (I2xt) is not adapted. If it is braked too frequently, there is a risk of the motor being thermally overloaded or the motor overload monitoring does not work properly!
- The "inverter motor brake" function must not be used with vertical conveyors (hoists) or with active loads!

Tip!

If no brake resistor is used, the DC injection brake can also be used for a braking process in addition to the "inverter motor brake" and "Stopping of the ramp function generator".
▶ DC-injection braking (📖 283)

In applications with high mass inertia and long braking times (> 2 s), we recommend the use of the DC injection brake.

- The DC injection brake provides for an oscillation-minimised braking. The braking process generally takes more time than the "inverter motor brake" function with an optimised setting. Moreover, the function is only recommended for braking to a standstill.

In the following cases we recommend the "inverter motor brake" function:

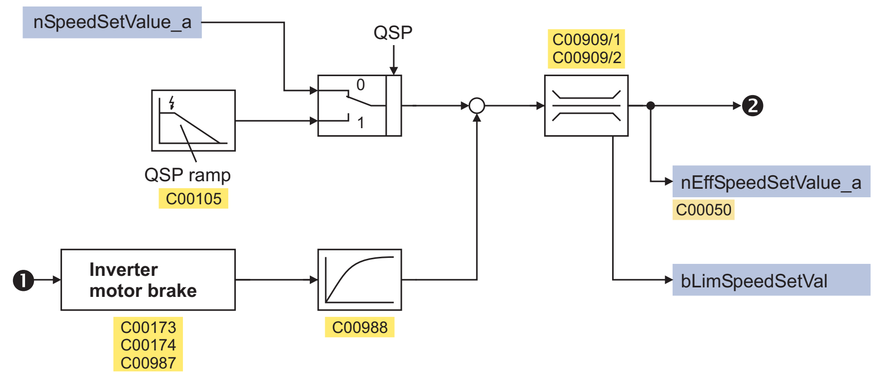
- With servo control (SC).
- For all applications that do not require braking to a standstill (e.g.braking to a lower speed setpoint) or the braking process can be interrupted by selecting a new speed setpoint.
- For applications with low mass inertias and a short braking time (< 1 s).
- For all applications where braking should be as quick as possible.

Operating mode of the inverter motor brake

The ramp function generator is stopped during acceleration. The speed set in C00987 is added to the speed setpoint by means of a hysteresis-type 2-point DC bus voltage controller, whereby the sign of the current actual speed is taken into account. In addition, the ramp function generator is stopped during overvoltage.

If the DC bus voltage falls below a defined DC bus voltage potential of the hysteresis controller, the added speed is subtracted again and the ramp function generator is activated again.

The energy is converted into heat in the motor due to alternating instances of acceleration and deceleration as a result of this switching operation.



❶ DC-bus voltage
❷ Speed setpoint for motor control

[5-30] Signal flow of the "Inverter motor brake" function

- In case of an asynchronous motor, the additive speed setpoint (C00987) should be 1 ... 4 times the slip of the machine:

$$\text{C00987 [rpm]} = 1 \ldots 4 \cdot (n_{Sync}[\text{rpm}] - n_{Rated}[\text{rpm}])$$

$$n_{Sync}[\text{rpm}] = \frac{f_{Rated}[\text{Hz}] \cdot 60}{p}$$

p = number of pole pairs
n_{Rat} = Rated speed of the motor
f_{Rat} = Rated frequency of the motor
n_{Sync} = Synchronous speed of the motor

[5-31] Formula for calculating the additive speed setpoint for an asynchronous motor

- In case of a synchronous motor, the additive speed setpoint (C00987) should be 5 ... 20 % of the rated machine speed.

Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value	Unit
C00173	Mains voltage	3ph 400V / 1ph 230V	
C00174	Reduced brake chopper threshold	0	V
C00175	Resp. to brake resistor control	Brake resistor	
C00987	Inverter motor brake: nAdd • Speed lift which is connected in pulses to the brake ramp when the motor is braked.	80	rpm
C00988	Inverter motor brake: PT1 filter time • PT1 filter time for smoothing the speed lift which is added in pulses.	0.0	ms

Note!

When the "inverter motor brake" function is used, torque oscillations occur which have may have a negative effect on the service life of the components of the mechanical drive train (e.g. gearbox).

- The extent of the occurring oscillations depends on the drive train (mass inertia, natural frequencies, etc.) and the function setting.
- We recommend optimising the "inverter motor brake" function for an oscillation-free operation as described in the following. Usually, this setting does not cause any torque oscillations which affect the service life of the gearbox.
- The settings of implementing a maximum acceleration ramp are only recommended if the inverter motor brake is used infrequently (e.g. in case of quick stop).

How to set the "inverter motor brake" function for an oscillation-reduced operation:

For V/f characteristic open-loop control/closed-loop control (VFCplus):

- Set reduced brake chopper threshold (C00174) to approx. 70 V.
- Set additive speed (C00987) to rated slip speed.
- Adapt the deceleration ramp so that the deceleration time is slightly below (10 ... 30 %) the deceleration time that can be realised with the inverter motor brake.

For sensorless vector control (SLVC) and servo control (SC):

- Set reduced brake chopper threshold (C00174) to approx. 50 V.
- Set additive speed (C00987) to 1 ... 2-fold rated slip speed.
- Adapt the deceleration ramp so that the deceleration time is slightly below (10 ... 30 %) the deceleration time that can be realised with the inverter motor brake.

How to set the "inverter motor brake" function for a maximum acceleration ramp:

For V/f characteristic open-loop control/closed-loop control (VFCplus):

- Set reduced brake chopper threshold (C00174) to approx. 70 V.
- Set additive speed (C00987) to 1,5 ... 2,5-fold rated slip speed.
- Adapt the deceleration ramp so that the deceleration time is slightly below (10 ... 30 %) the deceleration time that can be realised with the inverter motor brake.

For sensorless vector control (SLVC) and servo control (SC):

- Set reduced brake chopper threshold (C00174) to approx. 70 V.
- Set additive speed (C00987) to 2 ... 4-fold rated slip speed.
- Adapt the deceleration ramp so that the deceleration time is slightly below (10 ... 30 %) the deceleration time that can be realised with the inverter motor brake.

5.12.3 Avoiding thermal overload of the brake resistor

- Parameterisation of an error response in C00574 and evaluation of the parameterised error message within the application or within the machine control system.
  - See chapter entitled "Brake resistor monitoring (I2xt)". (📖 314)
- External interconnection using the thermal contact on the brake resistor (e.g. supply interruption via the mains contactor and activation of the mechanical brakes).

5.12.4 Control of multiple internal brake choppers in the DC-bus system

This function extension is available from version 12.00.00!

If an additional control signal is used, all internal brake choppers can be used in the DC-bus system to dissipate regenerative energy ("Master-slave operation").

Stop!

The integration of external brake choppers (e.g. brake chopper 9352) in the above-described "Master-slave operation" is not permissible since the voltage levels for the input and output of the external brake chopper are not suitable for the brake transistor control or, more precisely, for the output of the state of the brake transistor.

If the internal brake choppers of the DC-bus system are not sufficient, they should be replaced by an external brake chopper. It can be synchronised with other external brake choppers, if need be, so that simultaneous switch-on of all external brake choppers is ensured.

Note!

For trouble-free operation, the setting of the mains voltage in C00173 must be identical for all inverters of the DC-bus system since this setting also influences the brake chopper threshold for switching on the brake chopper.

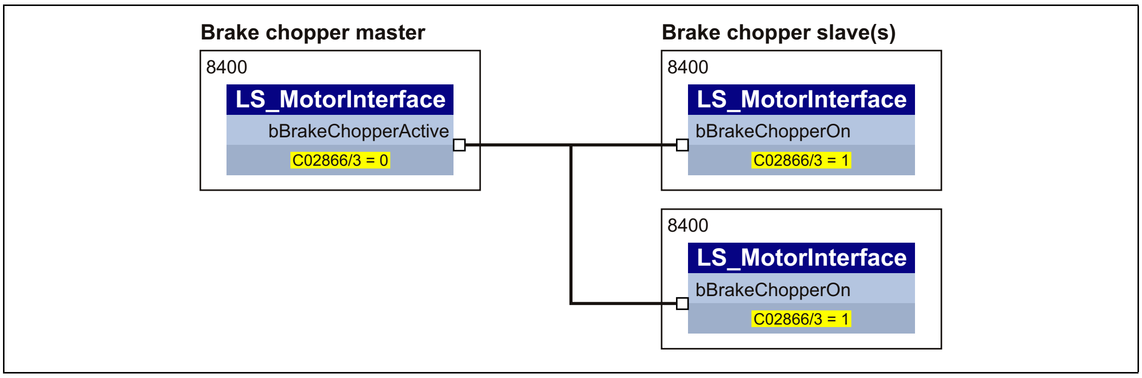
Functional principle

One of the inverters of the DC-bus system is assigned the role of the "brake chopper master".

- For logical reasons, the "brake chopper master" should be the most powerful inverter.
- The "brake chopper master" controls its internal brake chopper via the DC-bus voltage as before. In addition, the "brake chopper master" transmits the *bBrakeChopperActive* status signal of its internal brake chopper control to the other inverters of the DC-bus system via fieldbus or digital output.

All the other inverters of the DC-bus system are "brake chopper slaves".

- The "brake chopper slaves" have the *bBrakeChopperActive* status signal received from the "brake chopper master" connected to the *bBrakeChopperOn* control input.
- If the internal brake transistor of the "brake chopper master" is switched on, the internal brake transistors of the "brake chopper slaves" are switched on at the same time.



[5-32] Functional principle of the "Brake chopper master-slave operation" (simplified representation)

Procedure

1. Assign the role of the "brake chopper master" to one of the inverters of the DC-bus system.
2. Adapt the function block interconnection for the "brake chopper master" so that the *bBrakeChopperActive* status signal of the LS_MotorInterface SB is provided to the other inverters for controlling the internal brake chopper.
   - The *bBrakeChopperActive* signal can e.g. be output via port block to the fieldbus or via digital output.
   - A free output of the application block can be used to transfer the signal from application level to I/O level.
3. Configure all the other inverters of the DC-bus system as "brake chopper slaves". Select "1: Yes" in C2866/3 for these inverters.
   - With this setting, the brake chopper is not controlled via the DC-bus voltage anymore. Its control now depends on the *bBrakeChopperOn* control signal.
4. Adapt the function block interconnection for the "brake chopper slaves" so that the *bBrakeChopperActive* signal received from the "brake chopper master" is connected to the *bBrakeChopperOn* input of the LS_MotorInterface SB.
   - Depending on the output at the "brake chopper master", the signal must be read in e.g. via port block or digital input.
   - A free input of the application block can be used to transfer the signal from I/O level to application level.
   - If the digital inputs/outputs are used for transmitting the signal, they must be connected electrically accordingly.

Note!

If the brake chopper master-slave operation is activated, plausibility monitoring is carried out in the "brake chopper slaves":

- The brake transistor can only be switched on via the *bBrakeChopperOn* control signal if the DC-bus voltage is greater than the brake chopper threshold minus 40 V_{DC}.
- Exception: From a mains voltage of 513 V_{AC} (or 725 V_{DC}) and a mains voltage of 480 V or 500 V set in C00173, the plausibility monitoring is not effective anymore.

Fast discharge of the DC bus

The *bBrakeChopperOn* control input of the "brake chopper master" can be optionally used for a fast discharge of the DC bus after the supply voltage has been switched off. The FB interconnection f the "brake chopper master" has to be adapted so that the *bBrakeChopperOn* control input of the SB LS_MotorInterface is connected to a digital bus or hardware signal (e.g. digital input on HIGH level).

Note!

Only the brake transistor of the "brake chopper master" is switched on with this function (max. 4 seconds, without interruption).

A complete discharge of the DC-bus cannot be achieved using this function!

From version 14.00.00, the limitation of the operating time can be deactivated to max. 4 seconds. For this purpose, set bit 1 to "1" in C02864/1.

- We recommend this setting when this function is used in order to achieve a higher discharge of the DC bus - especially in case of high device power.

5.13 Monitoring

Many monitoring functions that are integrated in the inverter can detect errors and thus protect the device/motor from damage or overload.

- Detailed information on the individual monitoring functions can be found in the following subchapters.

Monitoring	Response		Error message (with activated monitoring)
	Lenze setting	Configuration	
Device overload monitoring (Ixt)	Warning	C00604	oC5
Motor load monitoring (I2xt)	Warning	C00606	oC6
Motor overcurrent monitoring	Fault	-	oC7
Motor temperature monitoring (PTC)	Fault	C00585	oH3
Brake resistor monitoring (I2xt)	No Reaction	C00574	oC12
Motor phase failure monitoring	No Reaction	C00597	LP1
Motor phase error monitoring before operation	No Reaction	C02866/2	
Mains phase failure monitoring	Warning	C00565	Su02
Current monitoring for overload	No Reaction	C00584/1	oC18
Maximum current monitoring	No Reaction	C00609	oC10
Maximum torque monitoring	No Reaction	C00608	ot1
Motor speed monitoring	Fault	-	oS2

Parameterisable responses

If a monitoring function trips, the response set via the corresponding parameter is carried out. The following responses can be selected:

- "No response": Response/monitoring is deactivated.
- "Fault": Change of the operating status by a pulse inhibit of the power output stage.
- "Warning": Operating status of the inverter remains unchanged. Only a message is entered into the Logbook of the inverter.

Related topics:

▶ Device state machine and device states (📖 119)
▶ Diagnostics & error management (📖 719)
▶ Basics on error handling in the inverter (📖 719)
▶ Error messages of the operating system (📖 743)

5.13.1 Device overload monitoring (Ixt)

C00064/1...3 displays the device utilisation (ixt) in [%] in different time intervals:

Parameters	Information
C00064/1	Device utilisation (Ixt) • Maximum value of pulse utilisation (C00064/2) and permanent utilisation (C00064/3).
C00064/2	Device utilisation (Ixt) 15s • Pulse utilisation over the last 15 seconds (only for loads >160 %).
C00064/3	Device utilisation (Ixt) 3 min • Permanent utilisation over the last 3 minutes.
Greyed out = display parameter	

- If the device utilisation reaches the switch-off threshold set in C00123:
  - The error response set in C00604 will be carried out (Lenze setting: "Warning").
  - The "oC5: Ixt overload" error message will be entered into the Logbook.
  - The *bMctrlIxtOverload* status output of the LS_DeviceMonitor system block will be set to TRUE.
- A setting of C00604 = "0: No Reaction" deactivates the monitoring.

5.13.2 Motor load monitoring (I2xt)

The Inverter Drives 8400 are provided with a simple, sensorless, thermal I^2xt motor monitoring of self-ventilated standard motors which is based on a mathematical model.

- C00066 displays the calculated motor load in [%].
- If the calculated motor load reaches the motor load setting (C00120):
  - The error response set in C00606 will be carried out (Lenze setting: "Warning").
  - The "oC6: I2xt motor overload" error message will be entered into the Logbook.
  - The *bMctrlI2xtOverload* status output of the LS_DeviceMonitor system block will be set to TRUE.
- A setting of C00606 = "0: No Reaction" deactivates the monitoring.

Stop!

I^2xt motor monitoring does not provide full motor protection!
As the motor utilisation calculated in the thermal motor model is lost after mains switching, for instance the following operating states cannot be detected correctly:

- Restarting (after mains switching) of a motor that is already very hot.
- Change of the cooling conditions (e.g. cooling air flow interrupted or too warm).

Full motor protection requires additional measures such as the evaluation of temperature sensors that are located directly in the winding or the use of thermal contacts.

For the installation according to UL or UR, the safety instructions provided in the hardware manual must be observed! Among other things, the activation of the motor overload monitoring (I2xt) is required here.

Note!

From version 12.00.00, the thermal motor load displayed in C00066 can be pre-initialised when the device is connected to the mains, optionally using a fixed value or the value used last at the time when the device was switched off. The desired initialisation is selected in C00122. In the Lenze setting of C00122, the behaviour remains unchanged (no initialisation).

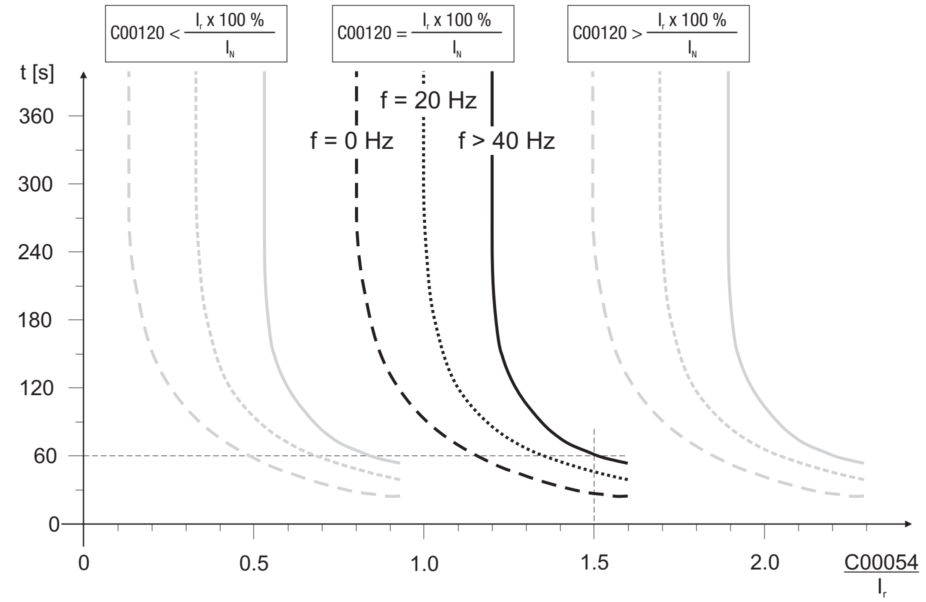
Adjustment of the motor utilisation meter

The motor utilisation meter for indicating the motor load in C00066 begins to count when the apparent motor current (C00054) is greater than the motor overload setting (C00120).

The overload threshold (C00120) is to be set as follows:

$$\text{C00120} = \frac{\text{Rated motor current (C00088)}}{\text{Rated device current (C00098)}} \cdot 100\ \%$$

- If you reduce C00120 starting from the calculated value, the motor utilisation meter will already be counted up before the rated overload threshold is reached.
- If you increase C00120 starting from the calculated value, the motor utilisation meter will not be counted up until the rated overload threshold is reached.



f: Output frequency
t: Release time
I_N: Rated device current at a switching frequency of f = 8 kHz
I_r : Rated motor current (see nameplate of motor)
C00054: Apparent motor current
C00120: Motor overload setting

[5-33] Tripping characteristic of the I^2xt monitoring

Example in Figure [5-33]:

C00120 = I_r / I_{rated} x 100 %

C00054 = 150 % rated motor current

- After approx. 60 seconds, C00066 has reached the final value (100 %) at output frequencies f > 40 Hz.
- The inverter outputs the "oC6: I2xt overload motor" error message and triggers the response set in C00606 (default setting: "Warning").

Tip!

- If forced ventilated motors are used, a premature response of the overload threshold can be avoided by deactivating this function if necessary (C00606 = "0: No Reaction").
- The current limits set in C00022 and C00023 influence the I^2xt calculation only in an indirect way. However, the operation of the motor at maximum possible load can be averted. ▶ Defining current and speed limits (📖 164)

5.13.3 Motor overcurrent monitoring

The ultimate motor current to be parameterised in C00939 is a limit value to protect the motor from destruction, influence of the rated data and demagnetisation.

- This limit value must not be travelled cyclically in the drive process.
- If the instantaneous value of the motor current exceeds the limit value set in C00939, the error response "Fault" occurs to protect the motor and the error message "oC7: Motor overcurrent" is entered into the logbook.
- The maximum currents to be parameterised in C00022 and C00023 should have a sufficient distance to this limit value.

Note!

If a Lenze motor is selected from the catalogue whose plant parameters are transferred into the inverter, the setting of the maximum current in C00022 and C00023 will automatically be adapted to the selected motor.

Related topics:

▶ Maximum current monitoring (📖 320)

5.13.4 Motor temperature monitoring (PTC)

For detecting and monitoring of the motor temperature, a PTC thermistor (DIN 44081/DIN 44082) or a thermal contact (NC contact) can be connected to the terminals X106/T1 and X106/T2.

Stop!

- The inverter can only evaluate one PTC thermistor!
  Do not connect several PTC thermistors in series or parallel.
- If several motors are operated on one inverter, use thermal contacts (NC contacts) connected in series.
- To achieve full motor protection, an additional temperature monitoring with separate evaluation must be installed.

Note!

- In the Lenze setting (C00585 = "1: Fault"), motor temperature monitoring is activated!
- There is a wire jumper between the terminals X106/T1 and X106/T2 by default.
- Lenze three-phase AC motors are provided with a thermal contact on delivery.

- If 1.6 kW < R < 4 kW at the terminals X106/T1 and X106/T2, the monitoring will respond, see functional test below.
- If the monitoring responds:
  - The error response set in C00585 is activated (Lenze setting: "Fault").
  - The "oH3: Motor temperature (X106) triggered" error message is entered into the Logbook.
  - The *bMctrlMotorPtc* status output of the LS_DeviceMonitor system block is set to TRUE.
- A setting of C00585 = "0: No Reaction" deactivates the monitoring.

Tip!

We recommend to always activate the PTC input when using motors which are equipped with PTC thermistors or thermostats. This prevents the motor from being destroyed by overheating.

Functional test

Connect a fixed resistor to the PTC input:

- R > 4 kΩ : Fault message must be activated.
- R < 1 kΩ : Fault message must not be activated.

5.13.5 Brake resistor monitoring (I2xt)

Due to the converted braking power, the brake resistor is thermally stressed and can even be thermally destroyed by excessive braking power.

The monitoring of the I^2xt utilisation of the inverter serves to protect the brake resistor. It acts in proportion to the converted braking power.

Danger!

In the Lenze setting (C00574 = "0: No Reaction"), the response of the monitoring function does not stop the braking process!

In particular for applications such as hoists or applications with a DC-bus connection, it must be checked if a stopping of the braking process due to a setting of C00574 = "1: Fault" is permissible.

Stop!

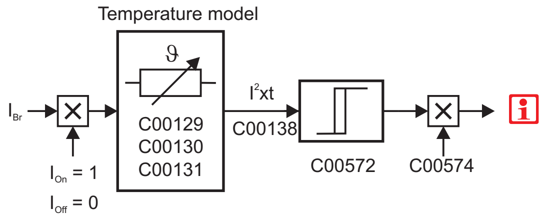
Implement appropriate protective measures against thermal overload of the brake resistor!

Examples:

- Parameterisation of an error response in C00574 and evaluation of the parameterised error message within the application or the machine control system.
- Interruption of the mains supply by means of the temperature contact at the brake resistor and a simultaneous activation of the mechanical brake.

- If the I^2xt utilisation reaches the switch-off threshold set in C00572:
  - The error response set in C00574 will take place.
  - The "oC12: I2xt brake resistor overload" error message is entered into the Logbook.
  - The *bMctrlBrakeChopperFault* status output of the LS_DeviceMonitor system block will be set to TRUE.
- If the system is dimensioned correctly, the monitoring should not be activated. If individual pieces of rated data of the actually connected brake resistor are not known, they have to be identified.
- If the DC-bus voltage exceeds the overvoltage threshold due to a braking energy that is too high, the monitoring for overvoltage in the DC bus is activated ("OU: DC-bus overvoltage" error message).
- Apart from the threshold of the I^2xt utilisation that can be set in C00572, there is the switching threshold of the brake transistor which results from the mains voltage (C00173) and the reduced brake chopper threshold (C00174).

Temperature model



Error message: "oC12: I2xt overload brake resistor"

[5-34] Signal flow for monitoring the brake resistor

The monitoring function calculates the braking current I_{Br} from the current DC-bus voltage U_{DC_act} and the brake resistance parameterised in C00129:

$$I_{Br} = \frac{U_{DC\_act}}{C00129}$$

Note!

The monitoring function can also be triggered due to a value entered in C00129 although a brake resistor is not even connected.

- The calculation considers the thermal utilisation of the brake resistor based on the following parameters:
  - Resistance value (C00129)
  - Continuous power (C00130)
  - Thermal capacity (C00131)
- In the Lenze setting these parameters are preset with the corresponding power-adapted Lenze brake resistor.
- C00133 indicates the calculated utilisation of the brake resistor in [%].
  - A utilisation of 100 % corresponds to the continuous power of the brake resistor depending on the maximally permissible temperature limit.

Related topics:

▶ Braking operation/brake energy management (297)

5.13.6 Motor phase failure monitoring

Note!

In the Lenze setting (C00597 = "0: No Reaction"), the motor phase failure monitoring is not activated!

In the case of a synchronous motor,
- the motor phase failure monitoring is basically deactivated. (Due to the low no-load current, the monitoring would be permanently activated.)
- only the Motor phase error monitoring before operation is active (for Lenze setting C2866/2 = "1: Yes").

In order to safely detect the failure of a motor phase, a certain motor current must flow for the current sensor system. Thus, the response set in C00597 (Lenze setting: "No Reaction") is caused after a delay time of maximally 2 s after controller enable if a current-carrying motor phase U, V, W fails or if motor connection is missing. If the current threshold value set in C00599 is already exceeded within the delay time, the motor phase failure monitoring starts from this point in time.

The monitoring mode checks the current flow for each motor phase as a function of the commutation angle. Monitoring is activated if a commutation angle of approx. 140° is covered without the current set in C00599 being exceeded. Monitoring is activated at an output frequency of 0 Hz if none of the three motor phases reaches the threshold value set in C00599.

- If the motor phase failure monitoring is tripped:
  - The response set in C00597 will take place.
  - The error message "LP1: Motor phase failure" is entered into the logbook.
  - The *bMctrlMotorPhaseFault* status output of the LS_DeviceMonitor system block is set to TRUE.

Note!

If an error response of "1: Fault" is set in C00597, the *bMctrlMotorPhaseFault* status output of the LS_DeviceMonitor SB will be set to TRUE for only 1 second in the event of a motor phase failure because it is no longer possible to detect a motor phase fault via the error response with a pulse inhibit. However, the Logbook and C00561/3...5 still display the cause of the motor phase failure.

- The motor phase failure monitoring is inactive if
  - a controller inhibit is set,
  - connection to a rotating machine is carried out (flying restart circuit or connection to actual speed value),
  - an error is pending due to a DC-bus overvoltage ("OU"),
  - motor parameter identification is carried out,
  - DC-injection braking is active.

5.13.7 Motor phase error monitoring before operation

This function extension is available from version 02.00.00!

This extended motor phase failure monitoring can both detect a phase failure on the basis of test signals and check for the existence of the motor.

- The "motor phase error monitoring before operation" is only directly active after controller enable if
  - an error response is set in C00597 **AND**
  - the motor phase error monitoring is switched on (C2866/2 = "1: Yes").
- The following parameters show the cause of the motor phase failure:
  - C00561/3: Motor phase U
  - C00561/4: Motor phase V
  - C00561/5: Motor phase W

Note!

The motor phase error monitoring before operation must not be connected to a rotating or coasting machine (high compensation currents and effect of the DC injection braking).

- In case of motor control with feedback, no motor phase error monitoring is executed if the actual speed value is > 10 rpm.
- In case of motor control without feedback, the user must ensure that the motor phase error monitoring will only be executed if the speed is 0.

If the motor is at quick stop and the brake is applied, no motor phase error monitoring is executed when quick stop is deactivated (same with "0" speed and applied brake).

If the rated current of the connected motor is lower than 10 % of the rated device current, the motor phase error monitoring can be activated although no motor phase error has occurred. In this case, the motor phase error monitoring must be switched off before operation (C2866/2 = "0: No").

Note!

With automatic brake control:

In case of automatic brake control, the brake will only be released if no motor phase failure exists and the magnetisation of the field-oriented control types is completed.

With manual brake control:

In case of manual brake control and forced release of the brake, the brake will be controlled directly as before.

The user himself must ensure that the brake will only be opened if all of the following conditions are met:

- Motor phase failure monitoring (C00597) and motor phase error monitoring before operation (C2866/2) are active.
- The inverter is enabled (controller enable).
- The *bMctrlMotorPhaseFault* status output of the SB LS_DeviceMonitor is set to FALSE.
- Bit 10 of the *MCTRL_Status3* status word must be set to 0 before the brake opening is triggered.
  - When the controller is enabled, this bit is set to 1 and will not be set to 0 again before the "Motor phase error monitoring before operation" is successfully completed.
  - The *MCTRL_Status3* status word can be integrated in the application via configuration parameters (e.g. C00620) (*MCTRL_Status3* = selection 34906 in Selection list - analog signals).

5.13.8 Mains phase failure monitoring

Stop!

Under load, the mains input of a three-phase inverter can be destroyed if the device is only supplied by two phases (e.g. if a mains phase fails).

The inverter has a simple mains-phase failure detection function with which a mains phase failure can be detected under load.

- In the case of power-adapted machines, approx. 50 % of the rated motor power must be exceeded so that a main-phase failure can be detected.
- If the mains phase failure monitoring is tripped:
  - The error response set in C00565 will be carried out (Lenze setting: "Warning").
  - The "Su02: One mains phase is missing" error message will be entered into the logbook.
  - The *bMctrlMainsFault* status output of the LS_DeviceMonitor system block will be set to TRUE.

5.13.9 Current monitoring for overload

This function extension is available from version 16.00.00!

If the apparent motor current exceeds a defined threshold value C00124/1) for a certain time (C00563/1) an overload has taken place.

Monitoring responds as follows:

- The *bCurrentMonitoringOverload* signal is set to TRUE
  See selection list - digital signals
- The response set in C00584/1 is activated (Lenze setting: "No response")
- The **OC18** error message, current monitoring overload, is entered into the logbook.
- The *bCurrentMonitoringOverload* status output of the **LS_DeviceMonitor** FB is set to TRUE.

If the overload decreases, the apparent motor current has to decrease below the value C00124/1 - 0.05 x I_{rated} in order that the *bCurrentMonitoringOverload* signal can accept the FALSE state.

When *bCurrentMonitoringOverload* = FALSE, the delay time in the resolution is set to the value 0 s again.

5.13.10 Maximum current monitoring

Note!

In the Lenze setting (C00609 = "0: No Reaction"), the maximum current monitoring is not activated!

If a Lenze motor is selected from the catalogue whose plant parameters are transferred into the inverter, the setting of the maximum current in C00022 and C00023 will automatically be adapted to the selected motor.

If the parameterised maximum current is reached, the response set in C00609 is triggered (Lenze setting: "0: No Reaction").

If the activated monitoring is tripped:

- The "oC10: Maximum current reached" error message is entered in the logbook.

Related topics:

▶ Motor overcurrent monitoring (📖 312)

5.13.11 Maximum torque monitoring

Note!

In the Lenze setting (C00608 = "0: No Reaction"), the maximum torque monitoring is not activated!

If the maximum possible torque C00057 is reached at the motor shaft, the response set in C00608 will be carried out (Lenze setting: "0: No Reaction").

If the activated monitoring is tripped:

- The "ot1: Maximum torque reached" error message is entered into the logbook.
- The *bMctrlTorqueMax* status output of the LS_DeviceMonitor system block will be set to TRUE.

5.13.12 Motor speed monitoring

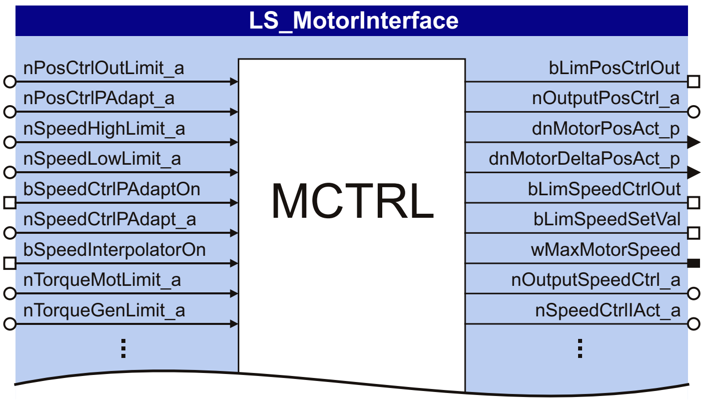
This function extension is available from version 02.00.00!

If the drive reaches the maximally permissible motor speed (C00965):

- The error response "Fault" occurs, i.e. the inverter is inhibited and the motor changes to torque-free operation (coasts down).
- The error message "oS2: Max. motor speed reached" is entered into the logbook.

5.14 Internal interfaces | System block "LS_MotorInterface"

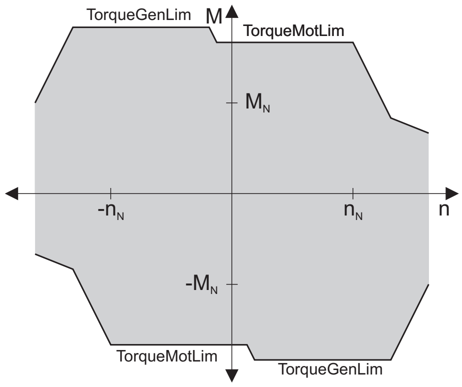
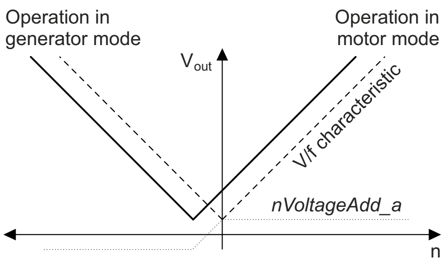
The **LS_MotorInterface** system block provides the internal interfaces to the driving machine in the function block editor.



[5-35] LS_MotorInterface system block (excerpt)

inputs

Designator DIS code | data type	Information/possible settings	
nPosCtrlOutLimit_a C00830/21 | INT	Limitation of the position controller output • Scaling: 16384 ≡ 100 % reference speed (C00011)	
nPosCtrlPAdapt_a C00830/20 | INT	Adaptation of the position controller gain • Scaling: 16384 ≡ 100 % Vp (C00254)	
nSpeedHighLimit_a C00830/88 | INT	Upper speed limit for the speed limitation • During torque-controlled operation only (*bTorquemodeOn* = TRUE) • Scaling: 16384 ≡ 100 % reference speed (C00011) **Note:** For a correct function of the speed limitation in both direction, the following applies: The speed limitation operates with a internal hysteresis of 50 rpm. The upper or lower speed limit resulting from the settings has to be higher than this hysteresis!	
nSpeedLowLimit_a C00830/23 | INT	Lower speed limit for speed limitation • During torque-controlled operation only (*bTorquemodeOn* = TRUE) • Scaling: 16384 ≡ 100 % reference speed (C00011) **Note:** For a correct function of the speed limitation in both direction, the following applies: The speed limitation operates with a internal hysteresis of 50 rpm. The upper or lower speed limit resulting from the settings has to be higher than this hysteresis!	
bSpeedCtrlPAdaptOn C00833/69 | BOOL	Adaptation of the speed controller gain	
	FALSE	Deactivate adaptive adaptation.
	TRUE	Activate adaptive adaptation.
nSpeedCtrlPAdapt_a C00830/25 | INT	Adaptation of the speed controller gain • Scaling: 16384 ≡ 100 % Vp (C00070)	
bSpeedInterpolatorOn C00833/28 | BOOL	Speed setpoint interpolation	
	FALSE	Deactivate interpolation
	TRUE	Activate interpolation

<table>
<tr><th>Designator
DIS code | data type</th><th colspan="2">Information/possible settings</th></tr>
<tr><td rowspan="2">nTorqueMotLimit_a
C00830/29 | INT
nTorqueGenLimit_a
C00830/28 | INT</td><td colspan="2">Torque limitation in motor mode and in generator mode
• The drive cannot output a higher torque in motor/generator mode than set here.
• The applied values (any polarity) are internally interpreted as absolute values.
• If V/f characteristic control (VFCplus) is selected, limitation is indirectly performed via a so-called I_{max} controller.
• If sensorless vector control (SLVC) or servo control (SC) is selected, limitation has a direct effect on the torque-producing current component.
• Scaling: 16384 ≡ 100 % M_{max} (C00057)
From version 18.00.00 onwards:
C02864: Bit 15 = 1: positive torque limitation (nTorqueHighLimit_a) and negative torque limitation (nTorqueLowLimit_a).</td></tr>
<tr><td colspan="2">Torque limits in motor and generator mode:

</td></tr>
<tr><td rowspan="3">bTorqueInterpolatorOn
C00833/29 | BOOL</td><td colspan="2">Torque setpoint interpolation</td></tr>
<tr><td>FALSE</td><td>Deactivate interpolation</td></tr>
<tr><td>TRUE</td><td>Activate interpolation</td></tr>
<tr><td rowspan="3">nVoltageAdd_a
C00830/31 | INT</td><td colspan="2">Additive voltage impression
• This process input serves to specify an additional setpoint for the motor voltage. If there are, for instance, different loads at the motor output end, it is possible to apply a voltage boost at the starting time.
• If the value is negative, the voltage is reduced.
• Scaling: 16384 ≡ 1000 V</td></tr>
<tr><td colspan="2">This process signal also serves to implement a load adaptation.
Example of V/f characteristic control (VFCplus): Adaptation of the voltage characteristic as a function of the load in motor mode/in generator mode in case of CW/CCW rotation


• CW rotation in motor mode: Increase voltage
• CCW rotation in generator mode: Reduce voltage</td></tr>
<tr><td colspan="2">STOP **Stop!**
Values selected too high may cause the motor to heat up due to the resulting current!</td></tr>
<tr><td>bAutoBoostOn
C00833/32 | BOOL</td><td colspan="2">Reserved</td></tr>
</table>

Designator DIS code | data type		**Information/possible settings**
nBoost_a C00830/26 | INT		Process signal for the V_{min} boost • This signal is added to C00016 and has thus an increasing or decreasing effect. • This signal serves to implement a load-dependent V_{min} boost to improve the torque behaviour in different load states, as for instance in case of operation in generator or motor mode. • Scaling: 16384 ≡ 100 % rated device voltage V_{FU} (400 V or 230 V) ▸ Adapting the Vmin boost (📖 176)
		Stop! Values selected too high may cause the motor to heat up due to the resulting current!
bPosCtrlOn C00833/27 | BOOL		Position/angle control
	FALSE	Deactivate position/angle control.
	TRUE	Activate position/angle control.
bDeltaPosOn C00833/35 | BOOL		Activate position difference as setpoint selection • In order to position the motor shaft, the position control function can work within the motor control function with the absolute position setpoint *dnPosSetValue_p* or alternatively with the speed setpoint *nSpeedSetValue_a* and the position difference *dnDeltaPos_p*.
	FALSE	Positioning with position setpoint *dnPosSetValue_p*.
	TRUE	Positioning with speed setpoint *nSpeedSetValue_a* and position difference *dnDeltaPos_p*.
dnDeltaPos_p C00834/4 | DINT		Position difference (following error input) • Difference between setpoint position and actual position in [increments] • Is used for position control if *bDeltaPosOn* = TRUE. • Scaling: 65535 ≡ 1 revolution
dnPosSetValue_p C00834/5 | DINT		Absolute position setpoint in [increments] • Is used for position control if *bDeltaPosOn* = FALSE. • Scaling: 65535 ≡ 1 revolution
bPosDerivativeOn C00833/67 |BOOL		Create a setpoint for the speed controller from the position setpoint • For highly dynamic control systems, the setpoint for the speed controller can be created from the absolute position setpoint *dnPosSetValue_p* instead of the speed setpoint *nSpeedSetValue_a* ▸ Position control/additive speed specification
	TRUE	Create a speed setpoint from the position setpoint. • The absolute position setpoint *dnPosSetValue_p* is differentiated and a speed value is created which is the setpoint for the speed controller. • Internal limitation of 65536 increments/ms.
bSetRefValue C00833/68 | BOOL (from version 18.00.00)		Trigger signal to set the actual position to the home position when the reference setting is requested or the reference signal is detected during the referencing procedure.
	TRUE	Set actual position to home position.
dnPosRefValue_p C00834/6 | DINT		Home position in [increments] • Scaling: 65535 ≡ 1 revolution
bQspOn C00833/33 | BOOL		Quick stop
	FALSE	Deactivate quick stop
	TRUE	Activate quick stop
nPWMAngleOffset_a C00830/32 | INT		Angle step change of output voltage phasor • Scaling: 65535 ≡ 1 revolution
bSpeedCtrlIOn C00833/31 | BOOL		Directly set the I-component of speed controller • In order to statically specify a minimum torque, e.g. when a load is being lifted.
	TRUE	Se the I-component of the speed controller to the value *nSpeedCtrlI_a*.

Designator DIS code | data type	Information/possible settings	

nSpeedCtrlI_a
C00830/24 | INT
	Value of the speed controller integrator
	• Scaling depends on the selected motor control:
	  • V/f control (VFCplus + encoder):
	    16384 ≡ 100 % reference speed (C00011)
	  • Servo control (SC) or vector control (SLVC):
	    16384 ≡ 100 % M_{max} (C00057)	

nSpeedSetValue_a C00830/22 | INT	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)	
bTorquemodeOn C00833/30 | BOOL	Selection: Speed/Torque control	
	FALSE	Speed control with torque limitation
	TRUE	Torque control with speed limitation
nTorqueSetValue_a C00830/27 | INT	Torque setpoint / additive torque • Scaling: 16384 ≡ 100 % M_{max} (C00057)	
bDcBrakeOn C00833/34 | BOOL	Activate DC injection brake	
	FALSE	Deactivate DC-injection braking
	TRUE	Activate DC-injection braking
bTorqueLimitAdaptOn C00833/98 | BOOL	Adaptation of torque limitation	
	TRUE	Activate adaptation of torque limitation.
nTorqueLimitAdapt_a C00830/70 | INT	Value for adaptation of torque limitation • Scaling: 16384 ≡ 100 % *nTorqueMotLimit_a* and *nTorqueGenLimit_a*	
nInertiaAdapt_a C00830/96 | INT (from version 12.00.00 onwards)	Adaptation of the moment of inertia • This process signal can be used during the process to dynamically control the percentage of the variable moment of inertia (e.g. a reel) set in C00919/1 which is to be considered for setpoint feedforward control. • Scaling: 16384 ≡ 100 % moment of inertia - process (C00919/1)	
bBrakeChopperOn C00833/130 | BOOL (from version 12.00.00 onwards)	Switch on the internal brake chopper as "brake chopper slave" (C02866/3 = "1: Yes") when the inverter is configured ▸ Control of multiple internal brake choppers in the DC-bus system (📖 305)	
	TRUE	Switch on internal brake chopper.
nSpeedSetValueInertia_a C00830/97 | INT (from version 12.00.00 onwards)	Input for differential setpoint feedforward control (torque feedforward control) • If the selection "1: nSpeedSetValueInertia_a" is set in C00654/1, this process signal can be used to preselect any input value (e.g. setpoint of the position or process controller) for torque feedforward control. • Scaling: 16384 ≡ 100 % reference speed (C00011)	
bVfcEcoDisable C00833/131 | BOOL (from version 13.00.00)	Deactivate energy optimisation for VFCplusEco ▸ Improving the behaviour at high dynamic load changes	
	FALSE	Activate energy optimisation.
	TRUE	Deactivate energy optimisation.

outputs

Designator DIS code | data type	Value/meaning	
bLimPosCtrlOut BOOL	"Position controller output inside the limitation" status signal	
	TRUE	The position controller output is internally limited
nOutputPosCtrl_a INT	Position controller output • Scaling: 16384 ≡ 100 % reference speed (C00011)	

<table>
<tr><th>Designator
DIS code | data type</th><th colspan="2">Value/meaning</th></tr>
<tr><td>dnMotorPosAct_p
DINT</td><td colspan="2">Current position in [increments]
• If the gearbox factor position encoder (C01203/1..2) is equal to "1", the motor position is output.
• Otherwise, if the gearbox factor position encoder (C01203/1..2) is not equal to "1" and a position encoder has been set in C00490, a position referring to the position encoder is output.</td></tr>
<tr><td>dnMotorDeltaPosAct_p
DINT</td><td colspan="2">Current following error in [increments]
• Following error = Difference between set position and actual position
• If the gearbox factor position encoder (C01203/1..2) is equal to "1", the motor following error is output.
• Otherwise, if the gearbox factor position encoder (C01203/1..2) is not equal to "1" and a position encoder has been set in C00490, a following error referring to the position encoder is output.</td></tr>
<tr><td rowspan="2">bLimSpeedCtrlOut
BOOL</td><td colspan="2">"Speed controller or manipulating variable of the slip regulator inside the limitation" status signal</td></tr>
<tr><td>TRUE</td><td>The speed controller output is internally limited</td></tr>
<tr><td rowspan="2">bLimSpeedSetVal
BOOL</td><td colspan="2">"Reduction or increase of the setpoint speed active" status signal</td></tr>
<tr><td>TRUE</td><td>Reduction or increase of the setpoint speed by the I_{max} controller is active</td></tr>
<tr><td>wMaxMotorSpeed
C00011 | BOOL</td><td colspan="2">Reference speed (C00011)</td></tr>
<tr><td>nOutputSpeedCtrl_a
INT</td><td colspan="2">Speed controller output
• Scaling: 16384 ≡ 100 % M_{max} (C00057)
• In case of the "VFC (+encoder)" motor control, this output is the output of the slip regulator. In this case, the scaling depends on the reference frequency (display in C00059):
  • C00059 < 650 Hz: 16384 ≡ 327.68 Hz (24000 ≡ 480.00 Hz)
  • C00059 > 650 Hz: 16384 ≡ 655.36 Hz (12000 ≡ 480.00 Hz)</td></tr>
<tr><td>nSpeedCtrlIAct_a
INT</td><td colspan="2">Current value of speed controller integrator
• Scaling depends on the selected motor control:
  • V/f control (VFCplus + encoder):
    16384 ≡ 100 % reference speed (C00011)
  • Servo control (SC) or vector control (SLVC):
    16384 ≡ 100 % M_{max} (C00057)</td></tr>
<tr><td>nEffSpeedSetValue_a
INT</td><td colspan="2">Effective speed setpoint
• Scaling: 16384 ≡ 100 % reference speed (C00011)</td></tr>
<tr><td>nMotorSpeedAct_a
C00051 | INT</td><td colspan="2">Actual speed value
• Scaling: 16384 ≡ 100 % reference speed (C00011)</td></tr>
<tr><td>nMotorSpeedAct_v
INT</td><td colspan="2">Actual speed value
• Scaling: 65535 ≡ 1 revolution</td></tr>
<tr><td>nMotorFreqAct_a
C00058 | INT</td><td colspan="2">Current field frequency
• Scaling depends on the reference frequency (display in C00059):
  • C00059 < 650 Hz: 16384 ≡ 327.68 Hz (24000 ≡ 480.00 Hz)
  • C00059 > 650 Hz: 16384 ≡ 655.36 Hz (12000 ≡ 480.00 Hz)</td></tr>
<tr><td rowspan="2">bLimTorqueSetVal
BOOL</td><td colspan="2">"Setpoint torque inside the limitation" status signal</td></tr>
<tr><td>TRUE</td><td>The setpoint torque is internally limited</td></tr>
<tr><td>wMaxMotorTorque
C00057</td><td colspan="2">Maximum motor torque
• wMaxMotorTorque = 10 * M_{max} (C00057)</td></tr>
<tr><td>nInputTorqueCtrl_a
INT</td><td colspan="2">Input value of the torque control (torque setpoint)
• Scaling: 16384 ≡ 100 % M_{max} (C00057)</td></tr>
<tr><td>nMotorTorqueAct_a
C00056/2 | INT</td><td colspan="2">Actual torque
• In the "VFC (+encoder)" motor control mode, this value is determined from the current motor current and only approximately corresponds to the actual torque value..
• Scaling: 16384 ≡ 100 % M_{max} (C00057)</td></tr>
</table>

Designator DIS code | data type	Value/meaning	
nInputJerkCtrl_a INT	Input value of the jerk limitation • Scaling: 16384 ≡ 100 % M_{max} (C00057)	
bLimCurrentSetVal BOOL	"Current setpoint inside the limitation" status signal	
	TRUE	The current setpoint is internally limited
nStatorCurrentIS_a INT	Current stator current/effective motor current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	
nEffCurrentIq_a INT	Current torque-producing cross current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	
nReaktCurrentId_a INT	Current field-producing direct-axis current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	
nActualFluxx_a INT	Current magnetising current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	
nDCVoltage_a INT	Current DC-bus voltage • Scaling: 16384 ≡ 1000 V	
nMotorVoltage_a INT	Current motor voltage/inverter output voltage • Scaling: 16384 ≡ 1000 V	
bQspActive BOOL	"Quick stop active" status signal	
	TRUE	Quick stop is active
bAutoDCBActive BOOL	"Automatic DC-injection braking active" status signal ▸ DC-injection braking (📖 283)	
	TRUE	Automatic DC-injection braking is active
bIdentificationActive BOOL	"Motor parameter identification active" status signal ▸ Automatic motor data identification (📖 151)	
	TRUE	Motor parameter identification is active
bFlyingSyncActive BOOL	"Flying restart function active" status signal ▸ Flying restart function (📖 280)	
	TRUE	Flying restart function is active
bHlgLoad BOOL	Control signal for an additional loading function of the ramp function generator • → L_NSet_1.*bExternalCINH* • To enable the ramp function generator to follow automatically when the controller is inhibited, for jerk-free setpoint connection.	
	TRUE	Set the ramp function generator to a setpoint of *nHlgSetValue_a*
nHlgSetValue_a INT	Setpoint for an additional loading function of the ramp function generator • → L_NSet_1.*nCInhVal_a* • For speed-controlled drive tasks, the current actual speed value (e.g. in case of an active pulse inhibit, flying restart function, controller inhibit) is provided at this output. • Scaling: 16384 ≡ 100 % reference speed (C00011)	
bHlgStop BOOL	Control signal for stopping the ramp function generator (L_NSet_1)	
	TRUE	Stop the ramp function generator
bBrakeChopperActive BOOL (from version 12.00.00 onwards)	Status signal of the internal brake chopper control ▸ Control of multiple internal brake choppers in the DC-bus system (📖 305)	
	TRUE	Internal brake chopper is switched-on.
nVoltageAngleAct_a INT (from version 13.00.00)	Current electrical voltage output angle of the inverter • Scaling: 16384 ≡ 360°	
bLimSpeedTorquemodeOn BOOL (from version 13.00.00)	Status signal of torque control with speed limitation	
	TRUE	Speed limitation for torque control is active.

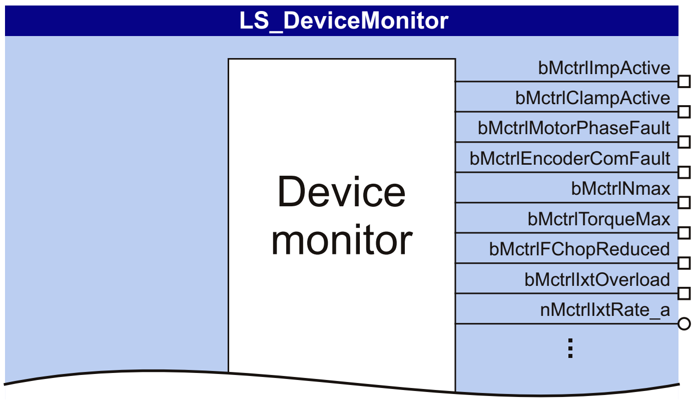
5.15 Internal status signals | System block "LS_DeviceMonitor"

The **LS_DeviceMonitor** system block provides the status signals of the motor control in the function block editor.

Note!

The **LS_DeviceMonitor** system block can only be inserted on the application level.

If status signals of the motor control function are to be output via digital outputs or example, you can use the free *bFreeOut1* ... *bFreeOut8* outputs of the application block to transfer the desired status signals from the application level to the I/O level. On the I/O level, you can then establish the logical link to the digital output terminals.



[5-36] LS_DeviceMonitor system block (excerpt)

outputs

Designator Data type	Value/meaning	
bMctrlImpActive BOOL	TRUE	Pulse inhibit is active
bMctrlClampActive BOOL	TRUE	Clamp current limitation is active
bMctrlMotorPhaseFault BOOL	Status of the Motor phase failure monitoring Regardless of the response setting for motor phase failure monitoring (C00597), bMctrlMotorPhaseFault is always displayed in the DeviceMonitor	
	TRUE	Motor phase fault has been detected
bMctrlEncoderComFault BOOL	TRUE	Encoder error has been detected
bMctrlNmax BOOL	TRUE	Max. speed limitation is active
bMctrlTorqueMax BOOL	Status of the Maximum torque monitoring	
	TRUE	Max. torque limitation is active
bMctrlFChopReduced BOOL	TRUE	PWM frequency reduction is active
bMctrlIxtOverload BOOL	Status of the Device overload monitoring (Ixt)	
	TRUE	Device utilisation (Ixt) ≥ device utilisation threshold (C00123) • Lenze setting: C00123 = 100 %

Designator Data type		Value/meaning
nMctrlIxtRate_a INT	Current device utilisation (Ixt) • Scaling: 16384 ≡ 100 %	
bMctrlI2xtOverload BOOL	Status of the Motor load monitoring (I2xt)	
	TRUE	Thermal motor overload (I^2xt) ≥ motor overload setting (C00120) • Lenze setting: C00120 = 100 %.
nMctrlI2xtRate_a INT	Current thermal motor load (I2xt) • Scaling: 16384 ≡ 100 %	
bMctrlMotorPTC BOOL	Status of the Motor temperature monitoring (PTC)	
	TRUE	Temperature monitoring: An error has been detected.
bMctrlMotorTemp BOOL	Status of the Motor temperature monitoring (PT1000 or KTY)	
	TRUE	Thermal motor overload • One of the following error messages is pending: "oH7: Motor temperature resolver > C121" "oH9: Motor overtemperature resolver" "oH6: Motor temperature MultiEncoder > C121" "oH12: Motor overtemperature MultiEncoder"
bMctrlHeatSinkTemp BOOL	TRUE	Thermal inverter overload • The heatsink temperature (display in C00061) has reached the maximally permissible temperature. The "oH1: Overtemperature heatsink" error message is pending and the "Fault" response is activated. • Furthermore, this output is set if the heatsink temperature has exceeded the maximally permissible temperature for the switching frequency set. A response with an "oH4: Heatsink temp.. > switch-off temp. -5°C" error message can be set in C00582. In the Lenze setting, no response will be effected.
bMctrlMainsFault BOOL	Status of the Mains phase failure monitoring	
	TRUE	Mains phase failure/Mains failure
bMctrlFanFault BOOL	TRUE	Fan monitoring: An error has been detected.
bMctrlNmaxForFChop BOOL	TRUE	The maximum field frequency for the respective switching frequency has been exceeded.
bMctrlShortCircuit BOOL	TRUE	Motor short circuit has been detected
bMctrlEarthFault BOOL	TRUE	Earth fault has been detected
bMctrlUVDetected BOOL	TRUE	An undervoltage has been detected
bMctrlOVDetected BOOL	TRUE	An overvoltage has been detected
bMctrlBrakeChopperFault BOOL	Status of the Brake resistor monitoring (I2xt) • This output is set independent of the set error response of the monitoring function.	
	TRUE	The I^2xt utilisation has reached the switch-off threshold set in C00572. • Lenze setting: C00572 = 100 %.
wUB24V WORD	Current 24 V supply voltage • Scaling: 1000 ≡ 1.000 V	
nMctrlActiveOutputPower_a INT (from version 14.00.00)	Current active output power • Scaling: 16384 ≡ double rated power of the inverter	
nMctrl ApparentOutputPower_a INT (from version 14.00.00)	Current apparent output power • Scaling: 16384 ≡ double rated power of the inverter	

Designator Data type	Value/meaning	
nMctrlCosinePhiAct_a INT (from version 14.00.00)	Current cosine phi • Scaling: 16384 ≡ cosine phi of 1.0	
bWirebreakUfLinearActive BOOL (from version 15.00.00)	Status "V/f emergency operation due to encoder open circuit active" • The internal change-over to "V/f emergency operation" can be suppressed by setting bit 8 to "1" in C02864/1.	
	TRUE	Due to an encoder open circuit, it is internally switched to the encoderless V/f characteristic control in order to avoid impermissible motor movements. This does not influence the setting in C00006.
bCurrentMonitoring Overload BOOL (From version 16.00.00)	Status of the Current monitoring for overload	
	TRUE	The monitoring function is activated and the apparent motor current has exceeded the switch-off threshold set in C00124/1 for the delay time set in C00563/1. **Note:** This status output is reset with a hysteresis of 5 % of the rated device current (C00098).
bSlpsmSpeedopen LoopControl BOOL (from version 17.00.00)	Implementation of an own flat ramp in the open-loop controlled operation of the SLPSM. Setting a flat ramp in the open-loop controlled operation and a steep ramp in the closed-loop controlled operation serves to achieve a considerably higher acceleration of the entire drive.	
	TRUE	Open-loop controlled operation of the SLPSM is active
	FALSE	Open-loop controlled operation of the SLPSM is not active

6 Encoder/feedback system

The 8400 TopLine uses various encoder/feedback systems for the motor controls with speed feedback and the position control:

- Resolver at X7
- Multi-Encoder at X8
  - Incremental TTL encoders
  - Incremental SinCos encoders
  - SinCos absolute value encoder with HIPERFACE® protocol
  - SSI absolute value encoder with Stegmann-SSI protocol
  - Digital frequency coupling (from version 12.00.00)
- HTL encoder at the digital DI1/DI2 or DI6/DI7 input terminals

Danger!

- If the encoder/resolver is used as motor encoder:
  Safe operation of the motor is no longer ensured in the event of an error!
- If servo control (SC) or V/f control (VFCplus + encoder) are used: For safety reasons, always select "Fault" (Lenze setting) in the following parameters as a response for the (open-circuit) monitoring of the encoder!
  - C00586: Resp. open circuit HTL encoder
  - C00603/1: Resp. open circuit MultiEncoder
  - C00603/2: Resp. open circuit resolver
- If an HTL encoder is used at the digital input terminals:
  Observe the maximum input frequencies of the digital inputs!
  - DI1/DI2: max. 100 kHz (from version 02.00.00 onwards: f_{max} = 200 kHz)
  - DI6/DI7: max. 100 kHz
- Generally, we recommend the use of encoder signals with increments ≥ 4
  - With increments < 4, the evaluation may cause implausible values or a number range overflow may occur when further processing the value.
  - Increments > 2 are at least required for encoder signals that are exclusively processed in the FB interconnection.
- To avoid the injection of interference when an encoder is being used, only use shielded motor and encoder cables!

Note!

Speed feedback is essential for the following motor control modes with feedback:

- Servo control (SC)
- V/f control (VFCplus + encoder)

Note!

The encoder position can be saved safe against mains failure in the inverter.

- For this purpose, bit 0 must be set in C02652. The actual position of the motor control is then saved in the inverter (not in the memory module) and thus remains known to the drive control after mains switching.
  ▸ Option "Actual MCTRL position received at mains switch-off" (📖 655)
- In case of a device replacement, this information gets lost and the home position might be approached or set again once. A device replacement can be recognised by using the SB LS_RetainData and a suitable function block interconnection.

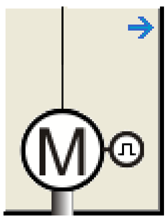
If an absolute value encoder (Multiturn) is used, the actual position will always be reconstructed after mains switch-off/switch-on, no matter if the bit 0 in C02652 has been set to "0" or "1".

Wiring diagram, assignment and electrical data of the terminals for resolvers and multi encoders can be found in the **hardware manual 8400** in the chapter "technical data". The hardware manual is stored in electronic form on the data carrier supplied with the 8400 inverter.

6.1 Encoder selection

How to get to the parameterisation dialog of the encoder/feedback system:

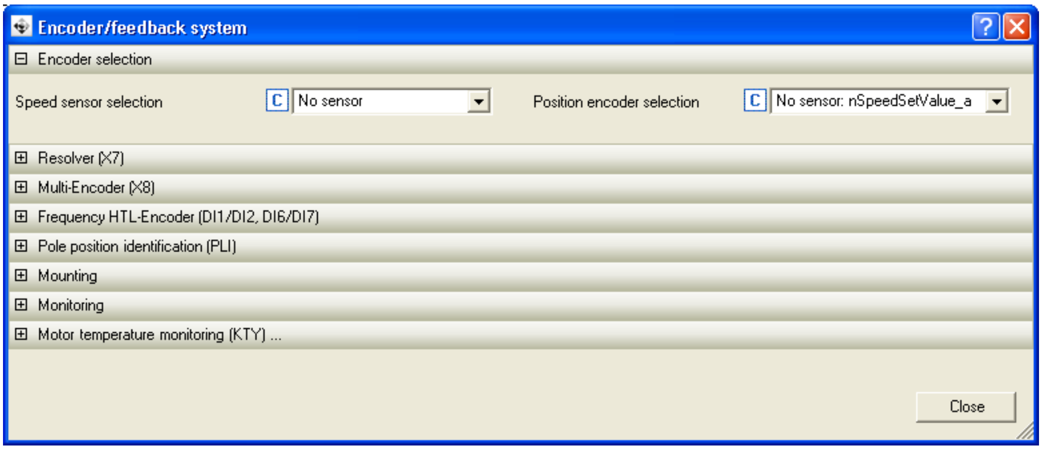
1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the following button:



4. Go to the *Overview* → *Motor data* dialog level and click the **Encoder/Feedback system...** button to open the parameterisation dialog for the encoder/feedback system.

Parameterisation dialog in the »Engineer«

From the »Engineer« V2.19, the following parameterisation dialog is available for parameterising the encoder/feedback system. For a better overview, the parameterisation dialog contains different categories which can be expanded/collapsed by a simple click. First, only the "encoder selection" category is expanded:



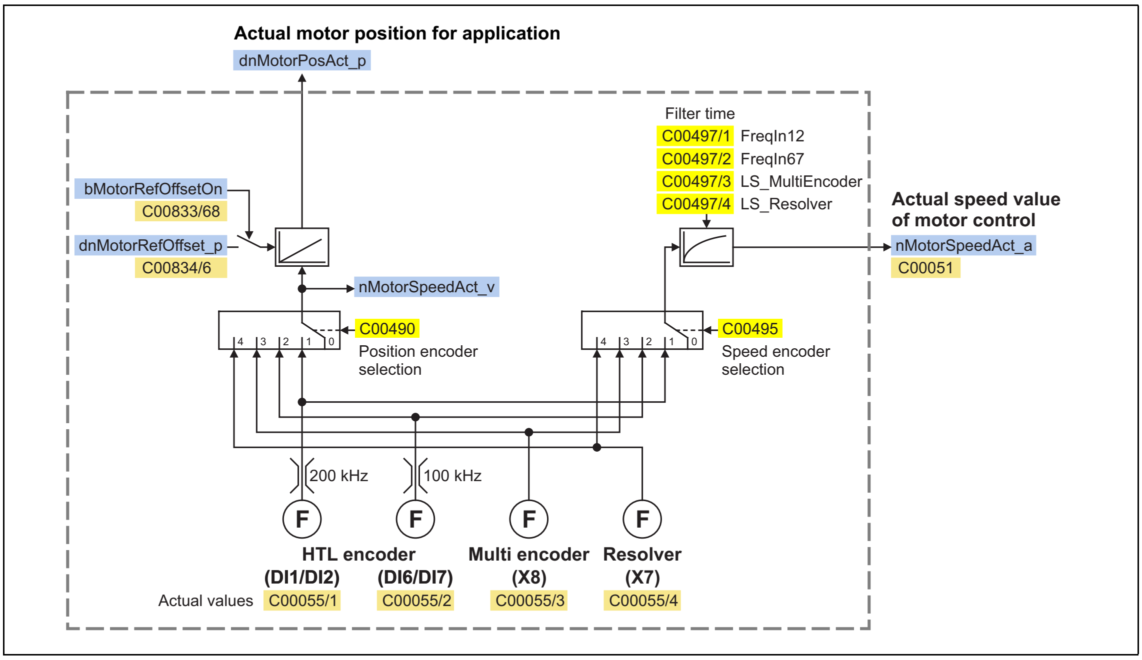
Parameters	Info	Lenze setting
C00495	Speed sensor selection • Source of feedback signal for speed control.	0: No sensor
C00490	Position encoder selection • Source of feedback signal for position control.	0: no sensor: nSpeedSetValue_a

Go to the **Speed sensor selection** list field and select the connected speed sensor. After the selection, the respective category wit the parameters relevant for the selected encoder is automatically expanded.

Detailed information on the parameters of the other categories can be found in the corresponding chapter:

- Resolver (X7) (📖 335)
- Multi-Encoder (X8) (📖 341)
- Frequency HTL encoder (DI1/DI2, DI6/DI7) (📖 372)
- Pole position identification (PPI) (📖 379)
- Mounting (📖 388)
- Monitoring (📖 389)
- Motor temperature monitoring (PT1000 or KTY) (📖 395)

6.1.1 Signal flow - encoder interface



[6-1] Signal flow - encoder interface

Generation of the actual speed value

...depending on the encoder selection and mounting direction:

Speed sensor (C00495)	Position encoder (C00490)	Motor mounting direction (C01206/1)	Direction of rotation of motor shaft (at setpoint = Cw)	Actual speed value (nAct_v)
☐	☐	Not inverted	Cw	ΔSetPos
		inverted	Ccw	
☒	☐	Not inverted	Cw	C00495
		inverted	Ccw	
☒	☒	Not inverted	Cw	
		inverted	Ccw	
☐	☒	Not inverted	Cw	C00490 ➞ ΔActPos
		inverted	Ccw	

☐ no encoder ☒ Encoder set

Up to and including version 02.xx.xx the following applies:

- For motor control types without speed feedback (C00495 = "0: No encoder") a speed-proportional unit is taken for calculating the *nMotorSpeedAct_v* speed signal. This derivation, however, is very imprecise so that in case of applications with synchronous motors without speed feedback it is not possible to calculate the current position from the current *nMotorSpeedAct_v* speed signal.
- If no position encoder is available (C00490 = "0: No encoder"), the *dnMotorPosAct_p* position signal is always derived from the *nSpeedSetValue_a* speed setpoint. This derivation, however, is very imprecise since in this case, speed limitations (e.g. by overcurrent limitations) are not considered.

From version 12.00.00 the following applies:

- For applications with synchronous motors without speed feedback, an *nMotorSpeedAct_v*error-free speed signal is available. This is calculated from the electrical voltage output angle of the inverter considering the number of pole pairs of the *nMotorSpeedAct_v* speed signal.
  - From version 13.00.00, the electrical voltage output angle is shown at the *nVoltageAngleAct_a* output of the LS_MotorInterface SB (scaling: 16384 ≡ 360°). This process signal can for instance be used in applications where a change-over from inverter to mains operation is to be carried out (synchronisation of the inverter output voltage to the mains voltage).
- When synchronous or reluctance motors without feedback are used in the motor control types V/f characteristic control (VFCplus) and Sensorless control for synchronous motors (SLPSM), the *nMotorSpeedAct_v* speed signal can be used to create an error-free position signal via a control if the *nMotorSpeedAct_v* signal is read out by the control in a 1 ms cycle.
- If no position encoder is available (C00490 = "0: No encoder"), the *dnMotorPosAct_p* position signal is continued to be derived from the *nSpeedSetValue_a* speed setpoint. The extended selection text "0: No encoder:nSpeedSetValue_a" in C00490 refers to this behaviour.
- The new selection "10: No encoder: C495 or nMotorSpeedSetAct_v" in C00490 is used to calculate the *dnMotorPosAct_p* position signal either from the set speed feedback (when C00495 > 0) or from the *nMotorSpeedAct_v* speed signal (when C00495 = 0).
  - For all motor control types without speed feedback, this selection serves to improve the creation of the *dnMotorPosAct_p* position signal.
  - When synchronous or reluctance motors without feedback are used, the *dnMotorPosAct_p* position signal can be created correctly.
  - In case of the motor control types with speed feedback, the *dnMotorPosAct_p* position signal is directly created from the speed feedback signal.

6.2 Resolver (X7)

Danger!

- If the resolver is used as motor encoder:
  Safe operation of the motor is no longer ensured in the event of an error!
- If servo control (SC) or V/f control (VFCplus + encoder) are used:
  For safety reasons, always select "Fault" (Lenze setting) in C00603 2 as a response for the (open-circuit) monitoring of the resolver!

6.2.1 Parameterising the resolver

Note!

If a Lenze motor with resolver is used, the resolver is the only item that can be selected as speed sensor in C00495. Further resolver parameter setting is not required.

Moreover, Lenze synchronous motors do not require a pole position identification (PLI) as the resolvers of the Lenze synchronous motors are pre-adjusted.

Note!

Acceptance of the resolver pole position from a Servo Drive 9400

The resolver pole position (C00926/1) cannot be simply accepted from a Servo Drive 9400 if the pole position (C58/1) considerably differs from -90° in the Servo Drive 9400.

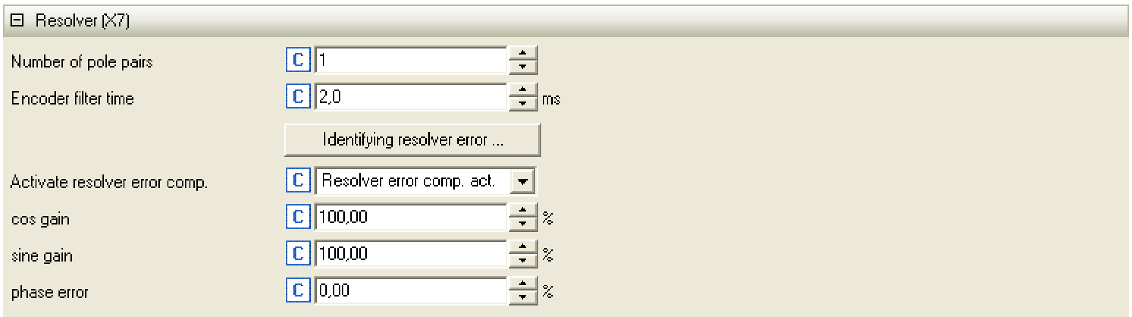
- We always recommend a pole position identification (PLI). (📖 379)
- As an alternative, the following conversion has to be carried out for a transfer of the resolver pole position from a Servo Drive 9400 to the 8400 TopLine:
  Pole position$_{8400}$ (C00926/1) = - (pole position$_{9400_C58/1}$ + 180°)

Note!

In C00926/3, the zero pulse can be shifted by ±179°. The setting only influences the generation of the zero pulse arithmetically. the pole position remains unchanged.
Possible application:

- When referencing to the zero pulse, you can shift the zero pulse in C00926/3 if the distance between pre-stop mark and encoder zero pulse is so small that the home position jitters due to the tolerance of the pre-stop mark. Thus, there is no mechanical shifting of the pre-stop mark and turning of the encoder on the shaft.

Parameterisation dialog (cutout)



Parameters	Info	Lenze setting Value	 Unit
C00925	LS_Resolver: Number of pole pairs	1	
C00497/4	LS_Resolver: Encoder filter time	2.0	ms
C00417	Activate resolver error comp. ▸ Optimising resolver behaviour	0: Activate resolver error compensation	
C02862/1	Resolver: cos gain	100.00	%
C02862/2	Resolver: sine gain	100.00	%
C02863	Resolver: Phase error	0.00	%

How to parameterise the resolver:

1. If number of resolver pole pairs ≠ 1:
   Set number of pole pairs (C00925).
2. Select resolver as speed sensor (C00495) or/and as position encoder (C00490).
3. Adjust filter time of the actual speed value (C00497/4) if necessary.
   - Lenze setting: 2 ms
   - Lenze recommend a filter time between 1 ... 2 ms.
4. If you use the resolver as motor encoder and the servo control with synchronous motor of a original equipment manufacturer: Carry out Pole position identification (PPI). (📖 379)
5. For detecting and monitoring the motor temperature via the resolver cable:
   Parameterise Motor temperature monitoring (PT1000 or KTY) via resolver. (📖 397)
6. Optional: Parameterise Resolver/multi-encoder open-circuit monitoring. (📖 390)

   In the Lenze setting, the resolver cable and/or encoder cable are monitored with regard to open circuit as a function of the selected speed and position encoder.
7. For a permanent acceptance of the parameter settings:

   Save parameter set.

6.2.2 Optimising resolver behaviour

Due to mounting and production tolerances as well as resolver material property leakage, errors may occur which, among other things, result in speed-dependent vibration of the actual speed. These errors are called resolver errors. Resolver errors typically occur in the form of the 1st and 2nd harmonic. They have two different causes:

1. The inductances of the sine and cosine track of the resolver have slightly different values.
2. Sine and cosine track do not magnetise orthogonally to each other.

Resolver errors due to cause 1 can be corrected by adjusting the gains of the digital/analog converters which feed the resolver tracks. In the Lenze setting, the gains of both resolver tracks are preset with identical values.

Resolver errors due to cause 2 can be compensated for by a slight correction of the angle via which both resolver tracks are fed relative to one another.

When the "Resolver error identification" device command (C00002/25) is executed, the gain of the resolver signals and the angular drift of both resolver tracks is corrected to minimise the resolver error.

- Select a speed-controlled operating mode (e.g. servo control) for your machine while you perform a resolver error identification run. During the identification run, speed must be constant and greater than 500 rpm.
- After a successful resolver error identification run, the resolver automatically uses the following resolver error parameters which have been identified during the procedure:

Parameters	Info	Lenze setting	
		Value	Unit
C02862/1	Resolver: cos gain	100.00	%
C02862/2	Resolver: sine gain	100.00	%
C02863	Resolver: Phase error	0.00	%

- The detected gain can have values from 80 ... 120 %. It makes sense to adjust only one of the two gains during a resolver error compensation. The other one remains at 100 % (Lenze setting).
- Save the parameter set afterwards to accept the identified resolver error parameters permanently (device command C00002/11).
- If the resolver error compensation is deactivated (C00417 = "1: Resolver error comp. deact."), the resolver will resume work with the Lenze setting. However, the identified resolver error parameters remain saved.

How to perform a resolver error identification run:

1. Select a motor control mode with speed feedback in C00006 and commission it. ▶ Selecting the control mode
   - Avoid a speed controller setting that is too tight (motor humming).
   - Set the speed ramps as well as the speed setpoint (> 500 rpm).
2. Inhibit the inverter, e.g. via the C00002/16 device command or via a LOW signal at terminal X5/RFR.
   - The inverter must be in the "SwitchedOn" status for the resolver error identification run to be performed.
3. Activate resolver error identification via the C00002/25 device command = "1: On / start".
4. Enable inverter again.
   - The inverter changes to the "Ident" device status.
   - The resolver error identification run starts.
   - The progress of the identification run can be seen in C00002/25.
   - During the resolver error identification run, the direction of rotation of the motor is not important.
   - We recommend to traverse at constant setpoint speed (|nset| > 500 rpm) until resolver error identification has finished.
   - If the application offers only limited space for traversing, reversing is also possible during active resolver error identification. In this case, ensure that the inverter is not inhibited and that the speed profile is traversed for at least 1 second at constant setpoint speed (|nset| > 500 rpm).
   - The identification is completed if the "0: Off / ready" message is displayed in C00002/25.
5. After the resolver error identification has been completed successfully, the inverter can be inhibited again. (When the inverter is inhibited, the "Ident" device state is quit again.)
6. In order to accept the identified resolver error parameters:

   Save parameter set.

Tip!

Resolver error identification can be aborted any time by setting a controller inhibit.

Note!

Some applications require that the resolver error identification can also be carried out during operation.

From version V12.00.00, the resolver error identification can also be started in the "OperationEnabled" device status (controller enable) using the device command C00002/25 = "1: on / start". Then there is no device state change, i.e. the device status remains at "OperationEnabled".

Furthermore, the following applies to the resolver error identification during operation:

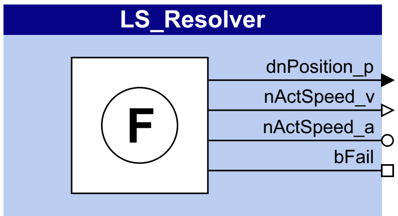
- The progress of the identification run can be seen in C00002/25.
- During the resolver error identification run, the direction of rotation of the motor is not important.
- We recommend to traverse at constant setpoint speed (|nset| > 500 rpm) until resolver error identification has finished.
- If the application offers only limited space for traversing, reversing is also possible during active resolver error identification. In this case, ensure that the inverter is not inhibited and that the speed profile is traversed for at least 1 second at constant setpoint speed (|nset| > 500 rpm).
- The identification is completed if the "0: Off / ready" message is displayed in C00002/25.

Possible causes for resolver error identification failure:

- A controller inhibit was set during resolver error identification.
- A time-out occurred while the algorithm was processed.
  - This error may occur if more than 60 seconds pass by between the setting of the "Resolver error identification" device command and the enable of the inverter.
- The setpoint speed was too small (|nset| < 500 rpm).
- The setpoint speed was not traversed for at least 1 second.

6.2.3 Internal interfaces | "LS_Resolver" system block

The **LS_Resolver** system block provides the application with speed, position and error information from the resolver in the form of process signals.



outputs

Designator Data type	Value/meaning	
dnPosition_p DINT	Actual encoder position in [increments]	
nActSpeed_v INT	Actual speed in [increments/ms]	
nActSpeed_a INT	Actual speed in [%] • Scaling: 16384 ≡ 100 % reference speed (C00011)	
bFail BOOL	TRUE	Group error: There is an error (e.g. resolver wire breakage) according to the *bFail* configuration in C00432/1. From version 14.00.00, the error type is displayed bit-coded in C00452/1. ▶ Monitoring

6.3 Multi-Encoder (X8)

Danger!

- If the resolver is used as motor encoder:
  Safe operation of the motor is no longer ensured in the event of an error!
- If servo control (SC) or V/f control (VFCplus + encoder) are used: For safety reasons, always select "Fault" (Lenze setting) in C00603 1 as a response for the (open-circuit) monitoring of the encoder!

Stop!

Before connecting an encoder, ensure that the encoder supply voltage has the correct setting in C00421:

- Lenze setting: 5 V (setting range: 5 ... 12 V)

If the set supply voltage exceeds the permissible supply voltage of the connected encoder, the encoder may be destroyed!

Note!

With regard to their position resolution, higher-level applications are based on the resolution of the encoder which is activated for position control.

The interface of the Multi-Encoder offers various ways in which the encoders can be used / evaluated:

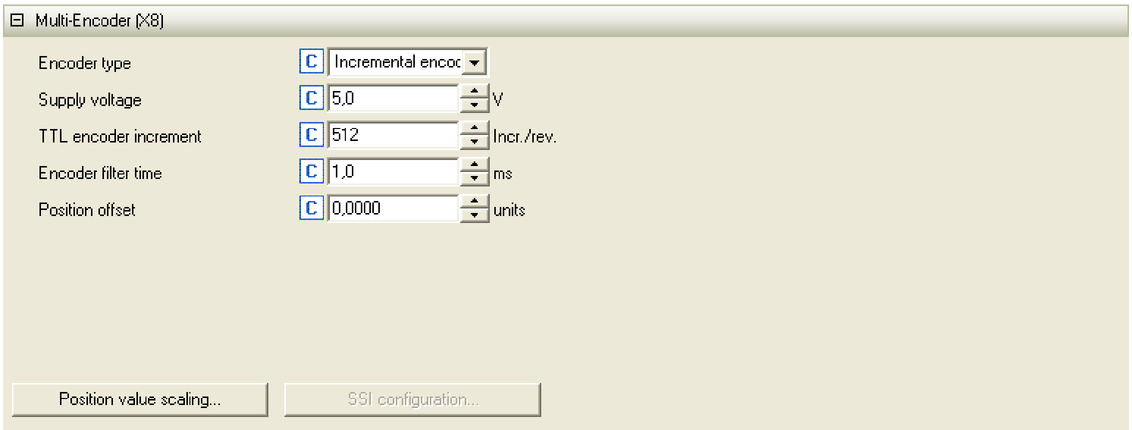
- Incremental TTL and SinCos encoder
- SinCos absolute value encoder with HIPERFACE® protocol
- SSI absolute value encoder with Stegmann-SSI protocol
- Digital frequency coupling (from version 12.00.00)

6.3.1 Incremental TTL and SinCos encoder

Tip!

The use of a SinCos encoder instead of a TTL encoder enables a fine interpolation of the position within one encoder increment due to the analog representation of the track signals A and B. This serves to considerably increase the resolution of the position information of a SinCos encoder compared to a conventional TTL incremental encoder.

Parameterisation dialog (cutout)



Further parameterisation dialogs: ▶ Scaling of position value

Parameters	Info	Lenze setting Value	Unit
C00422	LS_MultiEncoder: Encoder type	0: Incremental encoder (TTL)	
C00421	LS_MultiEncoder: Supply voltage	5.0	V
C00420/3	LS_MultiEncoder: Encoder increment	512	Incr./rev.
C00497/3	LS_MultiEncoder: Encoder filter time	1.0	ms
C01112/1	LS_MultiEncoder: Position offset	0.0000	units

How to parameterise the incremental TTL or SinCos encoder:

1. As encoder type (C00422), select "0: Incremental encoder (TTL)" or "1: Sin/Cos encoder" (depending on the type of the connected encoder).
2. Adapt supply voltage (C00421) depending on the cable length if required (see the following section "Voltage setting as a function of the cable length").
3. Set encoder increment (C00420/3).
4. Adjust filter time of the actual speed value (C00497/3) if necessary.
   - Lenze setting: 1 ms
   - Lenze recommend a filter time between 1 ... 2 ms.
5. If you use the encoder as motor encoder and synchronous motor:
   Carry out Pole position identification (PPI). (📖 379)
6. For detecting and monitoring the motor temperature via the Encoder cable:
   Parameterise Motor temperature monitoring (PT1000 or KTY) via encoder. (📖 398)
7. Optional: Parameterise Resolver/multi-encoder open-circuit monitoring. (📖 390)

   In the Lenze setting, the resolver cable and/or encoder cable are monitored with regard to open circuit as a function of the selected speed and position encoder.
8. For a permanent acceptance of the parameter settings:

   Save parameter set.

Voltage setting as a function of the cable length

The following table shows the settings required for different incremental encoder types as a function of the cable length. The given values apply when Lenze system cables are used at typical ambient temperatures. Other cables, cable cross-sections or extreme ambient temperatures may require metrologically detected adaptations.

Encoder type Product key	Rated voltage U_N	Encoder increment (C00420/3)	Voltage setting in C00421 with a cable length of			
			0 - 30 m	30 - 70 m	70 - 100 m	100 - 150 m
Incremental TTL encoders						
IG2048-5V-T	5 V ±5 %	2048	5.0 V	5.1 V	5.2 V	5.3 V
IG4096-5V-T		4096				
IK2048-5V-T		2048				
IK4096-5V-T		4096				
Incremental SinCos encoders						
IG1024-5V-V	5 V ±5 %	1024	5.0 V	5.1 V	5.2 V	5.3 V
IG2048-5V-S		2048				

6.3.2 SinCos absolute value encoder with HIPERFACE® protocol

SinCos absolute value encoders with HIPERFACE® protocol are feedback systems which transmit an absolute position on request. Moreover, they permanently provide an incremental signal via a sin/cos evaluation. The analog evaluation of the sin/cos tracks provides for a high resolution.

The absolute position is transmitted via an RS485 data interface with 9600 Baud (standard value).This bidirectional parameter channel also serves to request and transmit error codes, encoder data and other information. The analog process data channel is created by a differential transmission of sin/cos signals. The resolutions of these two information channels are generally not identical, thus the channels have to be parameterised independently of each other in the inverter.

Note!

If the RS485 cable is interrupted, a communication to the encoder is permanently tried to be built up. After the maximum initialisation time set in C00430/1 has expired, the error "Sd7: Error encoder communication" is triggered. This error is cancelled if a communication could be built up successfully and the encoder initialisation has been executed.

The selection of an error response in C00603/4 serves to optionally monitor the communication between absolute value encoder and inverter.

If the monitoring is tripped:
- The error response set in C00603/4 is triggered.
- The "Sd7: Encoder communication error" error message is entered into the logbook.

6.3.2.1 Initialisation

A connected SinCos absolute value encoder with HIPERFACE® protocol is initialised in the following cases:

A. When the device is switched on after the waiting time of approx. 1 s has expired.

B. After the encoder wire breakage has been removed.
In case of an encoder wire breakage, a speed and position detection cannot be carried out correctly. A renewed initialisation after removing the encoder wire breakage serves to connect to the correct absolute position.

C. After describing one of the following parameters as the setting has an impact on the internal scaling of the encoder position:

Parameters	Name	Parameters	Name
C00420/3	LS_MultiEncoder: Encoder increment	C00495	Speed sensor selection
C00422	LS_MultiEncoder: Encoder type	C01110	LS_MultiEncoder: Solid measure
C00490	Position encoder selection	C01111/1..3	LS_MultiEncoder: Encoder constant
C00494/1..2	Hiperface: Resolutions	C01204	MCK: Feed constant

Initialisation process

During initialisation, the inverter requests different information from the encoder. The process of this data transfer is as follows:

1. Reading the encoder status
   The encoder can provide up to 4 error codes which are displayed in C00491/1...4 from version 14.00.00.
2. Reading the encoder type (display of the TypeCode in C00492)
3. Reading the absolute position
   A constant encoder position is assumed to initiate the position value correctly. An initialisation is not permissible when the encoder is rotating.
4. Processing the read position data and linking the SinCos-based position and speed detection to the absolute position of the encoder.

Encoders detected automatically

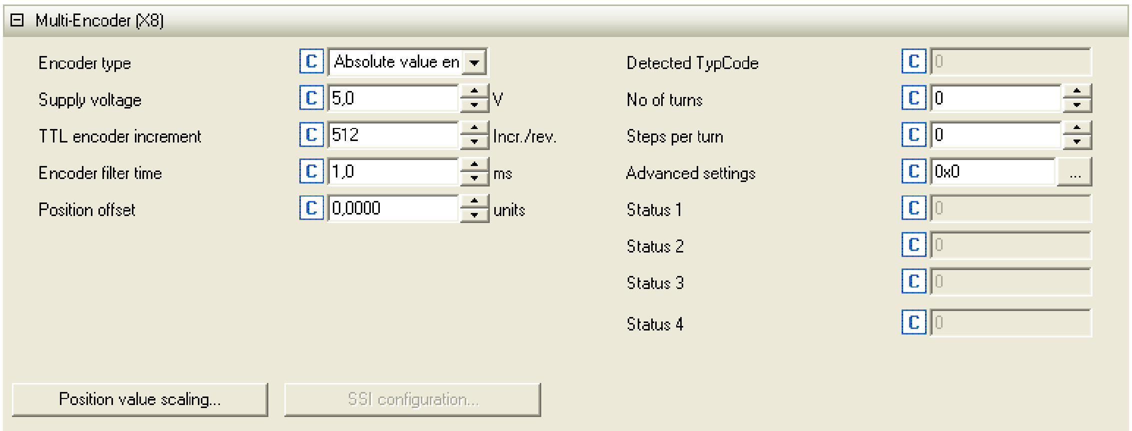
The following encoder types are detected and set automatically by the inverter:

Type	Increments/ revolution	Absolute revolutions	Bits Singleturn	Bits Multiturn	Bits SinCos track	Type code
AM1024-8V-H (SRM50)	1024	4096 (Multiturn)	15	12	10	0x27
AM1024-8V-H (SRM60)						
AM1024-8V-H (SRM64)						
AM512-8V-H (SCM60)	512		14	12	9	0x07
AM512-8V-H (SCM70)						
AM128-8V-H (SKM36)	128		12	12	7	0x37
AM16-8V-H (SEL37)	16		9	12	4	0x47
AM16-8V-H (SEL52)						
AS1024-8V-H (SRS50)	1024	1 (Singleturn)	15	0	10	0x22
AS1024-8V-H (SRS60)						
AS1024-8V-H (SRS64)						
AS512-8V-H (SCS60)	512		14	0	9	0x02
AS512-8V-H (SCS70)						
AS128-8V-H (SKS36)	128		12	0	7	0x32
AS16-8V-H (SEK37)	16		9	0	4	0x42
AS16-8V-H (SEK52)						

- Regarding the saving of position information, it is distinguished between:
  Singleturn: Saving within one revolution
  Multiturn: Saving within a number of revolutions
- In case of encoders with an extended electronic nameplate (TypCode = "0xFF"), the encoder settings currently cannot be read automatically out of the encoder and thus have to be executed manually. See the instruction "How to parameterise an absolute value encoder with an unknown type code" in the following "Parameter setting" subchapter.
- The encoder settings for all other encoders (TypeCode = "0x00") have to be made manually as well. See the instructions "How to parameterise an absolute value encoder with an unknown type code" in the following "Parameter setting" subchapter.

6.3.2.2 Parameterising the encoder

Parameterisation dialog (cutout)



Further parameterisation dialogs: ▸ Scaling of position value

Parameters	Info	Lenze setting Value	Unit
General settings			
C00422	LS_MultiEncoder: Encoder type	0: Incremental encoder (TTL)	
C00421	LS_MultiEncoder: Supply voltage	5.0	V
C00420/3	LS_MultiEncoder: Encoder increment • Setting is relevant for process data channel (incremental SinCos track).	512	Incr./rev.
C00497/3	LS_MultiEncoder: Encoder filter time	1.0	ms
C01112/1	LS_MultiEncoder: Position offset	0.0000	units
Settings for SinCos absolute value encoder with HIPERFACE® protocol			
C00492	Hiperface: Detected TypCode	-	
C00494/1	Hiperface: Number of revolutions • Number of resolved revolutions of the Hiperface track. • Setting is relevant for parameter channel.	0	
C00494/2	Hiperface: Steps per revolution • Resolution of the Hiperface track. • Setting is relevant for parameter channel.	0	
C00499/1	Hiperface: Advanced settings	0x0	
Greyed out = display parameter			

How to parameterise the SinCos absolute value encoder:

1. Set the supply voltage (C00421) of the used encoder.
2. As encoder type (C00422), select "2: Absolute value encoder (Hiperface)".
   - From version 14.00.00, the selection "6: Position encoder (Hiperface)" is possible as well.

   When the encoder has already been connected, a change of C00422 causes a renewed Initialisation of the connected encoder.
   - After the initialisation and with a known encoder, the read-out values are displayed in C00494/1..2 and cannot be overwritten.
   - If the value "0xFF" is displayed as TypeCode in C00492, it is an encoder with an extended nameplate. Currently, the settings for this encoder cannot be automatically read out of the encoder yet and thus have to be carried out manually. See the following instructions "How to parameterise an absolute value encoder with an unknown type code".
   - If the value "0x00" is displayed as type code, the encoder has not been detected automatically. In this case, make the settings manually as well. See the following instructions "How to parameterise an absolute value encoder with unknown type code".
3. Only for version 01.xx.xx: Set the number of sine/cosine periods per revolution as "increment" of the encoder in C00420/3.
   - This value can be obtained from the encoder data sheet.
   - This point must be executed last since it initiates a renewed readout of the encoder.
   - From version 02.00.00, this setting will be obtained and set automatically.
4. If you use the encoder as motor encoder and synchronous motor of a third party manufacturer:
   Carry out Pole position identification (PPI). (379)
5. For detecting and monitoring the motor temperature via the Encoder cable:
   Parameterise Motor temperature monitoring (PT1000 or KTY) via encoder. (398)
6. Optional: Parameterise Resolver/multi-encoder open-circuit monitoring. (390)

   In the Lenze setting, the resolver cable and/or encoder cable are monitored with regard to open circuit as a function of the selected speed and position encoder.
7. For a permanent acceptance of the parameter settings:

   Save parameter set.

How to parameterise a SinCos absolute value encoder with unknown type code:

If the encoder is not detected automatically, the value "0x00" is displayed in C00492 as TypeCode. In this case, make the settings manually:

1. Set the number of steps per revolution in C00494/2.
2. Set the number of revolutions for a multiturn encoder in C00494/1.
3. Set the number of sine/cosine periods per revolutions as "increment" of the encoder in C00420/3.
   - This value can be obtained from the encoder data sheet.
   - This point must be executed last since it initiates a renewed readout of the encoder.
4. If you use the encoder as motor encoder and synchronous motor of a third party manufacturer:
   Carry out Pole position identification (PPI). (379)
5. For detecting and monitoring the motor temperature via the Encoder cable:
   Parameterise Motor temperature monitoring (PT1000 or KTY) via encoder. (398)
6. Optional: Parameterise Resolver/multi-encoder open-circuit monitoring. (390)

   In the Lenze setting, the resolver cable and/or encoder cable are monitored with regard to open circuit as a function of the selected speed and position encoder.
7. For a permanent acceptance of the parameter settings:

   Save parameter set.

6.3.2.3 Display of the current position

In C01119/1, the current actual position of the motor shaft is displayed in [units]. It is calculated from the actual value of the encoder selected in C00422:

$$C01119/1 = \left(Pos \cdot C01204 \cdot \frac{C01202/2}{C01202/1} \cdot \frac{C01203/1}{C01203/2}\right) + C01112/1$$

Pos = encoder position in increments
- For rotative encoders, the following applies: 65536 increments = 1 revolution
- For linear encoders, the following applies: 65536 increment = 1 increment path

[6-2] Actual position calculation

As the term "revolution" does not make any sense in case of linear encoders, the distance covered is defined as the path in which the increments or SinCos periods parameterised in C00420/3 are completely run through once. In of rotative encoders, this corresponds to exactly one encoder revolution.

Internal conversion of the encoder information on the motor shaft position

In the inverter, the information of the Hiperface track are factorised so that a revolution of the encoder corresponds to 65536 increments which means a passage of the increments or SinCos periods parameterised in C00420/3. The same applies to linear encoders.

The factor is automatically calculated from the encoder data (C00494/1..2) that have either been saved in the device or manually configured.

Example: 4096 Hiperface increments per revolution ➞ Factor = 65536/4096 = 16

In case of linear encoders, a conversion to one revolution of the motor or gearbox output shaft is made, too. This conversion is made considering the traversing path set in C01111/3 which is covered during a passage of the increments and sin/cos periods parameterised in C00420/3:

$$Pos_{Mot} = Pos_{lin} \cdot \frac{C01111/3}{C01204}$$

[6-3] Conversion to one revolution of the motor or gearbox output shaft in case of linear encoders

Related topics:

- Scaling of position value

6.3.2.4 Notes on evaluating absolute value encoder of SEL37 and SELxx types by SICK

For version 02.xx.xx the following applies:

- The 8400 TopLine inverter recognises these encoder types, but sporadically reports the error "Sd4: open circuit MultiEncoder" error if the open-circuit monitoring is activated in C00603/1 (Lenze setting).
  - The reason for this is a faulty open-circuit monitoring during the initialisation of the controller.
  - The error can be acknowledged so that the controller is able to start correctly.
- Moreover, the initialisation time of the encoder is critical. This is the time needed internally by the encoder to respond to a read request of the controller.
  - The maximum initialisation time in the inverter is permanently set to 5.2 s. Depending on the encoder used, it may be possible that this time is not sufficient (see data sheet for the encoder manufacturer)!
- If the monitoring (e.g. die open-circuit monitoring) triggers with the "Fault" error response due to an encoder error during operation, the position is not valid anymore in case of absolute value encoders and must be reread from encoder.
  - After the error message is reset, this does not take place automatically which may cause a wrong position!
  - Remedy: After the error message is reset with the SB LS_ParReadWrite_1, rewrite the code C00422 and thus enforce a re-initialisation of the encoder. Then enable the application.

From version 12.00.00 the following applies:

- The initialisation time of the encoder is not critical anymore since the maximum initialisation time can now be adjusted in C00430/1 (to max. 20 seconds).
  - During the set initialisation time, the inverter remains in the "Init" device state and cannot be enabled.
  - The open-circuit monitoring is also suppressed during the set initialisation time so that the error behaviour of the version 02.xx.xx is remedied as well.
- If the monitoring (e.g. die open-circuit monitoring) triggers with the "Fault" error response due to an encoder error during operation, the position is not valid anymore in case of absolute value encoders and must be reread from encoder.
  - After the error message is reset, this takes place automatically.

6.3.2.5 Configuration example 1: Absolute wire draw encoder "XKS09-HTBM0527" by SICK

Data

- Wire length: 5 m
- Hiperface resolution: 4096 digits x 4184 "revolutions"
- SinCos resolution: 128 increments (1 period ≡ 1.1953 mm ➞ 153 mm ≡ one "revolution")
- 1 [unit] = 1 mm
- Feed of the mechanics = 4 mm/revolution

Scaling of the position value

In order to scale the evaluation of the SinCos track, the traverse path in [units] has to be entered in C01111/3 which is covered in order to pass through once the SinCos increment of the encoder (C00420/3) (here 128 increments x 1.1953 mm period length = 153 mm).

The absolute Hiperface track is scaled by the setting of the "steps per revolution" in C00494/2. Here, a revolution is equated to the path which corresponds to the SinCos number of increments of the encoder (C00420/3).

Required settings

Parameters	Info	Setting for example
C00494/1	Hiperface: Number of revolutions	4184
C00494/2	Hiperface: Steps per revolution	4096
C01110	LS_MultiEncoder: Solid measure	1: linearly unipolar
C01111/3	LS_MultiEncoder: Traverse path - encoder increment	153

In "Machine parameters", the feed constant and the gearbox factors for motors and position encoders must be set as follows:

- The position encoder gearbox factor must be set to 1:1.
- When the motor gearbox factor is set to 1:1, the path covered at one motor revolution has to be entered as feed constant in C01204 in [units].
- In case of a different motor gearbox factor, the path covered at one defined "drive roll" revolution has to be entered as feed constant in C01204 in [units].

The feed of the mechanics entered in C01204 (here: 4 mm) is displayed with 65536 increments.

6.3.2.6 Configuration example 2: Distance sensor "DME5000-117" by Sick

Data

from DME5000 operating instructions (8009813/US35/2012-07-02):

Period Length (mm)	Type code	Resolution RS-485 (mm)	Resolution sin/cos when zero crossings are counted (mm)
1	90h	1/32	0.25
2	91h	1/16	0.5
4	92h	1/8	1
8	93h	1/4	2
16	94h	1/2	4

- Maximum distance: 70 m
- Default setting of parameter "Period Length": 1 mm
  (4 zero crossings while passing one period * 0.25 mm = 1 mm)
- Default setting of parameter "Resolution": 125 µm (must not be changed)

These default settings result in the following resolutions:

- 1 "encoder revolution" = 10 mm
- Hiperface resolution: 1/32 mm ➜ 320 digits per "revolution" x 7000 "revolutions"
- Sin/cos resolution: 10 increments per "revolution"
  (1 period ≡ 1 mm ➜ 10 mm ≡ one "revolution" = 10 periods or increments)
- 1 [unit] = 1 mm

Scaling of the position value

In order to scale the evaluation of the SinCos track, the traverse path in [units] has to be entered in C01111/3 which is covered in order to pass through once the SinCos increment of the encoder (C00420/3) (here 10 increments x 1 mm period length = 10 mm).

The absolute Hiperface track is scaled by the setting of the "steps per revolution" in C00494/2. Here, a revolution is equated to the path which corresponds to the SinCos number of increments of the encoder (C00420/3).

Required settings

Parameters	Info	Setting for example
C00494/1	Hiperface: Number of revolutions	7000
C00494/2	Hiperface: Steps per revolution	320
C01110	LS_MultiEncoder: Solid measure	1: linearly unipolar
C00420/3	LS_MultiEncoder: Encoder increment	10
C01111/3	LS_MultiEncoder: Traverse path - encoder increment	10

6.3.3 SSI absolute value encoder with Stegmann-SSI protocol

All SSI encoders which use the Stegmann SSI protocol are supported.

- Supported bit rates for SSI communication: 100 ... 1000 kbits
- Supported data word widths: 1 ... 31 bits (effective)
- Supported output code of the SSI encoder: Gray or binary
- The SSI encoder can be used as position encoder or master encoder with a minimum cycle time of 1 ms.
- The SSI encoder can be supplied via X8 up to a maximum voltage of 12 V and a maximum current of 0.25 A.
- The received SSI data words are provided to the application via the LS_MultiEncoder system block for further processing in the function block editor.

Note!

For some SSi encoder types it is common that despite a high data transfer rate an slow position detection (e.g. every 5, 10 or 20 ms) takes place. In this case, the position control, working in a 1-ms cycle, only functions with low gain and thus low dynamics since the values of the SSI encoder are only updated every x ms.

When an external SSI laser position encoder is used at X8:

Also observe the Notes for using an SSI laser position encoder at the end of this chapter. (📖 358)

The technical data of the synchronous serial interfaces (SSI) can be found in the **8400 hardware manual**.

- The hardware manual has been stored in electronic form on the data carrier supplied with the 8400 inverter.

Tip!

When C00422 = 7 / 8 (position encoder external/SSI), the drive system is directly connected to the position of the feedback.

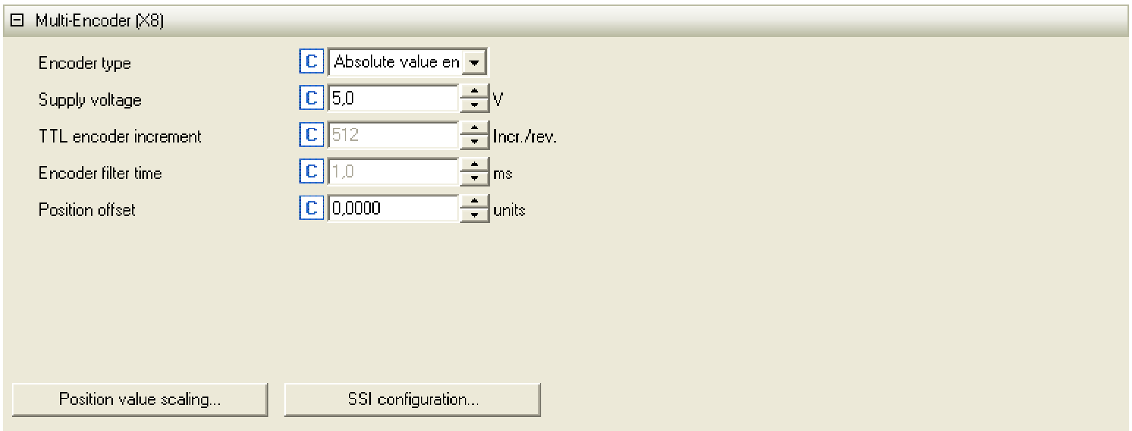
Code C01112/1 (LS_MultiEncoder : Position offset) does not have any impact on the encoder position.

Finally, only

- a once-only homing has to be executed
  ▶ Homing (📖 637), mode 100: SetRef) or
- "mains switching" can be carried out.

6.3.3.1 Parameterising the encoder

Parameterisation dialog (cutout)

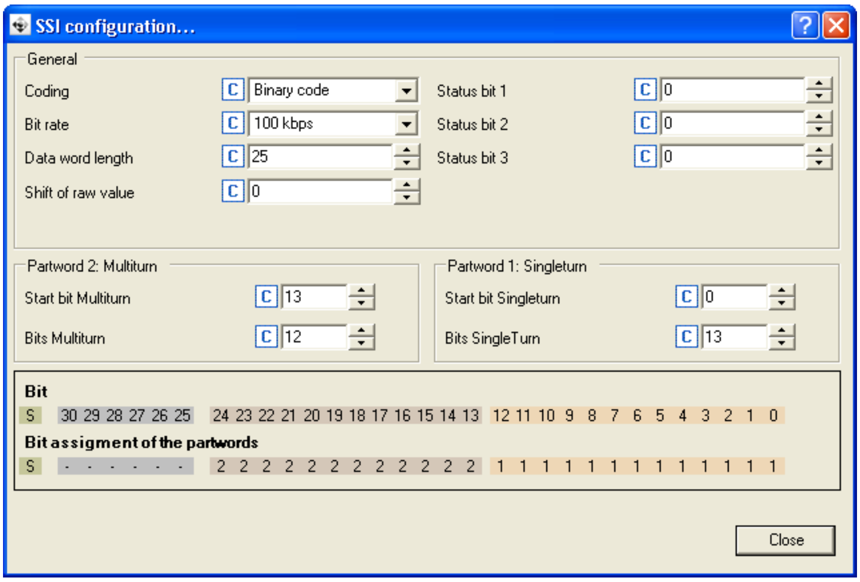


Further parameterisation dialogs: ▸ Scaling of position value ▸ SSI configuration

Parameters	Info	Lenze setting Value	Unit
C00422	LS_MultiEncoder: Encoder type	0: Incremental encoder (TTL)	
C00421	LS_MultiEncoder: Supply voltage	5.0	V
C01112/1	LS_MultiEncoder: Position offset	0.0000	units

SSI configuration

Press the **SSI configuration...** button to get to the parameterisation dialog for configuring the SSI encoder:



Parameters	Info	Lenze setting Value	Unit
General			
C00428	SSI encoder: Coding	0: Binary code	
C00427	SSI encoder: Bit rate	1: 100 kbps	
C00426/1	SSI encoder: Data word length	25	
C00426/7	SSI encoder: Shift of raw value	0	
C00426/6	SSI encoder: Status bit 1	0	
C00426/8	SSI encoder: Status bit 2	0	
C00426/9	SSI encoder: Status bit 3	0	
Partword 2: Multiturn			
C00426/5	SSI encoder: Start bit Multiturn	13	
C00426/4	SSI encoder: Bits Multiturn	12	
Partword 1: Singleturn			
C00426/3	SSI encoder: Start bit Singleturn	0	
C00426/2	SSI encoder: Bits SingleTurn	13	

How to parameterise the SSI encoder:

1. Set the supply voltage (C00421) of the used SSI encoder.

   In case of a higher supply power of encoders, e.g. with laser-based length measuring systems, an external supply of the encoders must be provided.
2. As encoder type (C00422), select "4: Absolute value encoder (SSI)".
3. Set the bit rate (C00427) for SSI communication.
   - For SSI protocols, the permissible baud rate is reduced if the cable length is increased. A safe bit rate must be set, depending on the length of the used encoder cable and the electromagnetic interference level.
4. Set the partitioning of the SSI data word in C00426/1...5 according to the used SSI encoder (see Configuration example 1).
5. If an SSI encoder with Gray coding is used:
   Select "1: Gray code" in C00428 to convert from Gray to binary.
   - In the Lenze setting "0: Binary code", no conversion takes place, i.e. an SSI encoder with binary coding is expected.
6. If a linear SSI encoder is used instead of a rotative encoder or the encoder value is to be interpreted with or without sign:
   Set the corresponding standard in C01110. ▶ Scaling of position value

SSI encoder coding "Position Gray"

From version 14.00.00 onwards, also SSI encoders are supported that transfer the position data in Gray coding and the status bit in binary mode (uncoded). For an encoder with this coding, select "2:Position Gray" in C00428. Then, only the pure position data are converted from Gray to binary code, and status bits provided by the encoder are directly evaluated. The position data converted to binary format are then added to the status bits again and provided as raw value at the outputs *wHighWord* and *wLowWord* of the LS_MultiEncoder. This takes place as a provision of the position raw value in gray code would not make any sense; the value, in case of a linear encoder, would feature jumps instead of a linear characteristic.

Evaluation of further status bits

From version 14.00.00 onwards, two more status bits of the SSI encoder can be evaluated. The bits to be evaluated in the SSI data current have to be set in C00426/8..9.

If the evaluation of a status bit shall be deactivated completely, the value "32" has to be entered into the corresponding code. Thus, the status bit always has the value "0".

Left shift of raw value

The data read-in by the encoder is processed via the settings of the start bits for singleturn and multiturn and the number of bits in such a way that a position signal suitable for the inverter is created that can be fed into the function block interconnection.

The raw value of the received data coming from the SSI encoder is provided at the outputs *wHighWord* and *wLowWord* of the LS_MultiEncoder. It is always provided in binary format, i.e. in case of Gray code encoders after the received data is decoded. The MSB of the data is positioned at bit 16 of the *wHighWord* output.

In order to receive a standard data format (LSB at the lowest position) an offset can be set in C00426/7. The setting C00426/7 = 32 - data word length serves to display the raw value of the received data telegram so that the LSB is at bit 0 of the *wLowWord* output:

Clarification of the data sequence when the data word length is = 26 bits																															
wHighWord																*wLowWord*															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data sequence without offset (C00426/7 = 0):																															
25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
Data sequence with an offset by 6 bits (C00426/7 = 6):																															
						25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

6.3.3.2 Important notes for using an SSI laser position encoder

If you are using an SSI laser position encoder at X8 and, deviating from the Lenze setting, you have activated the position control for quick stop (C00104/1 - bit 0 and/or bit 1 set), please note the following:

If the laser beam is interrupted, due to the "TroubleQuickStop" following error standard response, the drive may move unexpectedly.

- A drive that had already reached the position-controlled standstill can execute a movement.
- A drive that is ramping down can continue running.

Possible countermeasures:

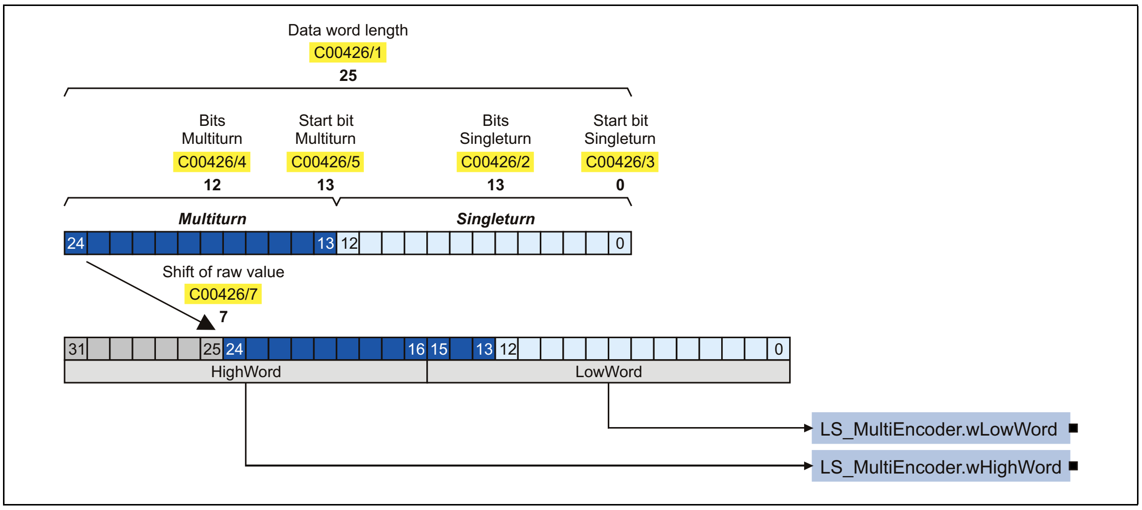
- Do not use the inverter with the above-described parameter setting.
- By implementing suitable other measures, ensure that, in terms of safety aspects, the response described is acceptable for the specific application case.
- Monitor the valid position:
  - Use parameters C01112/2 and C01112/3 to define a value range for valid positions.
  - Configure the LS_MultiEncoder.*bFail* output so that controller inhibit is set when the output at the drive is active. Take other requirements relating to the application into consideration (e.g. application of brake).
  - Resulting drive behaviour: If the value range set is exited, the LS_MultiEncoder.*bFail* output activates controller inhibit. ▶ Encoder value monitoring

Note!

The response of the drive to interruptions of the laser beam is always part of the error scenarios to be checked during commissioning.

6.3.3.3 Configuration example 1: Optical multi-turn rotary encoder "G0M2H Z04" by Baumer

Coding: Gray code with 25 bit data word length (8192x4096)

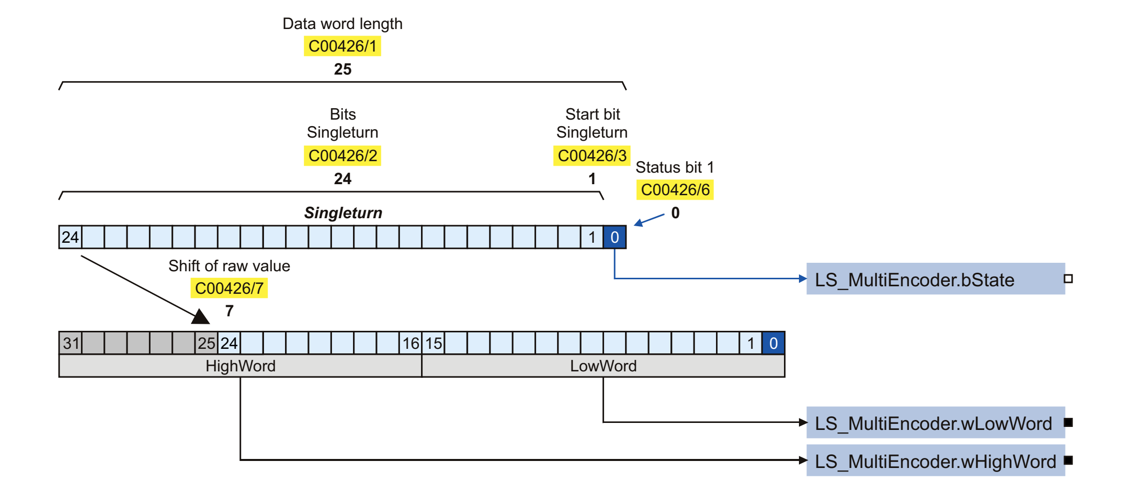


Parameters	Info	Setting for example 1
C00426/1	SSI encoder: Data word length	25
C00426/2	SSI encoder: Bits SingleTurn	13
C00426/3	SSI encoder: Start bit Singleturn	0
C00426/4	SSI encoder: Bits Multiturn	12
C00426/5	SSI encoder: Start bit Multiturn	13
C00426/6	SSI encoder: Status bit 1	0 (32)*
C00426/7	SSI encoder: Shift of raw value	7
C00426/8	SSI encoder: Status bit 2	0 (32)*
C00426/9	SSI encoder: Status bit 3	0 (32)*
C00428	SSI encoder: Coding	1: Gray code
C01110	LS_MultiEncoder: Solid measure	0: rotatively unipolar
* As the encoder data word in this example does not contain any status bits, you the evaluation of the status bits can also be deactivated completely by entering the value "32" in C00426/6, C00426/8 and C00426/9. Thus, the three status bit outputs of the LS_MultiEncoder always have the value "0".		

- In "Machine parameters", the feed constant and the gearbox factors for motor and position encoder must be set correctly. (📖 611)
- One encoder revolution is displayed with 65536 increments.

6.3.3.4 Configuration example 2: Distance sensor "DME5000-111" by Sick

Coding: Gray code with 25 bits of data word length
(Bit 0 = status bit in binary format, bit 1 ...24 = position, resolution 1 bit = 0.1 mm)



Parameters	Info	Setting for example 2
C00426/1	SSI encoder: Data word length	25
C00426/2	SSI encoder: Bits SingleTurn	24
C00426/3	SSI encoder: Start bit Singleturn	1
C00426/4	SSI encoder: Bits Multiturn	0
C00426/5	SSI encoder: Start bit Multiturn	0
C00426/6	SSI encoder: Status bit 1	0
C00426/7	SSI encoder: Shift of raw value	7
C00426/8	SSI encoder: Status bit 2	0 (32)*
C00426/9	SSI encoder: Status bit 3	0 (32)*
C00428	SSI encoder: Coding	2: Position Gray
C01110	LS_MultiEncoder: Solid measure	1: linearly unipolar

* As the encoder data word in this example contains only one status bit, the evaluation can be deactivated completely by entering the value "32" in C00426/8 and C00426/9. Thus, the status bit outputs *bState2* and *bState3* of the LS_MultiEncoder always have the value "0".

In "Machine parameters", the feed constant and the gearbox factors for motors and position encoders must be set as follows:

- The position encoder gearbox factor must be set to 1:1.
- When the motor gearbox factor is set to 1:1, the path covered at one motor revolution has to be entered as feed constant in C01204 in [units].
- In case of a different motor gearbox factor, the path covered at one defined "drive roll" revolution has to be entered as feed constant in C01204 in [units].
- The feed of the mechanics entered in C01204 is displayed with 65536 increments.

Scaling of the position value

In order to scale the position change provided by the position encoder at this feed to 65536 as well, enter the traverse path in [units] in C1111/1 and the corresponding difference of the raw value provided by the encoder in C1111/2.

- Only the bits that contain the position data may be taken from the data word for the raw value. If e.g. the bit 0 contains status information, the raw value must be shifted to the right by this bit since only then the SLB of the position value is positioned correctly.

Example for scaling

1 [unit] = 1 mm

Feed constant (C01204) = 4 mm/revolution

Encoder resolution = 50 µm ➞ raw value difference = 20 [digit]/1 [unit]

➞ Difference - traverse path (C1111/1) = 1 [unit]

➞ Difference - encoder value (C1111/2) = 20 [digit]

Thus, at a feed of the mechanics of 4 mm, the encoder provides a position difference of 65536 increments.

$$\text{Position} = \text{Raw value} \cdot \frac{\text{Encoder feed constant}}{65536}$$

$$\text{with encoder feed constant} = \frac{\text{C01111/1} \cdot 65536^2}{\text{C01111/2} \cdot \text{C01204}}$$

$$\text{Position} = \text{Raw value} \cdot \frac{\text{C01111/1} \cdot 65536}{\text{C01111/2} \cdot \text{C01204}}$$

6.3.4 Solid measure

For an encoder at the multi encoder terminal X8, the solid measure in C01110 (LS_MultiEncoder: solid measure) provides the following selection:

Parameters	Info
C01110 = 0	rotatively unipolar
C01110 = 1	linearly unipolar
C01110 = 2	rotatively bipolar
C01110 = 3	linearly bipolar
C01110 = 4	Inverted rotatively in unipolar mode
C01110 = 5	Inverted linearly in unipolar mode
C01110 = 6	Inverted rotatively in bipolar mode
C01110 = 7	Inverted linearly in bipolar mode

6.3.4.1 Rotary solid measure

The "rotary" solid measure is selected for encoders that are directly mounted onto the motor shaft or as external position encoder onto a machine axis/machine shaft. In case of external position encoders, the ratio defined by the type of mounting can be parameterised in code C01203. If the encoder is directly mounted onto the motor shaft, select the ratio 1:1.

The SSI data word must be parameterised with any status bit (C00426/6 C00426/8, C00426/9, usually bit 0 ... n), the position data for "Bits Singleturn" (C00426/2, C00426/3) and "Bits Multiturn" (C00426/4, C00426/5) and the data word length (C00426/1).

Example

Data encoder: 4096 revolutions × 4096 steps, 3 status bits, Gray coding

Parameter setting:

Parameters	Info	Setting for example 2
C00426/1	SSI encoder: Data word length	23 (10 + 10 +3)
C00426/2	SSI encoder: Bits SingleTurn	10
C00426/3	SSI encoder: Start bit Singleturn	3
C00426/4	SSI encoder: Bits Multiturn	10
C00426/5	SSI encoder: Start bit Multiturn	13
C00426/6	SSI encoder: Status bit 1	0
C00426/7	SSI encoder: Shift of raw value	9 (32-23)
C00426/8	SSI encoder: Status bit 2	1
C00426/9	SSI encoder: Status bit 3	2
C00428	SSI encoder: Coding	2: Position Gray
C01110	LS_MultiEncoder: Solid measure	0, 2, 4 or 6

6.3.4.2 Linear solid measure

The "linear" solid measure is selected if an external linear position "linear" measuring system is used, e.g. a laser measuring system, magnet tape encoder or bar code reading measuring system.

Code C01203 is used to parameterise the ratio 1:1. Code C01204 must be parameterised in such a way that the machine feed corresponds to one motor shaft revolution, irrespective of a gearbox mounted to the motor.

The codes C1111/1 (LS_MultiEncoder: traverse path difference) and C1111/2 (LS_MultiEncoder: encoder value difference) serve to parameterise the conversion for the position encoder value in the inverter. In C1111/1, the number of encoder digits is entered by which the raw value delivered by the encoder changes with the distance of the machine feed entered in C1111/2.

The SSI data word must be parameterised with any status bit (C00426/6 C00426/8, C00426/9, usually bit 0 ... n), the position data for "Bits Singleturn" (C00426/2, C00426/3) and "Bits Multiturn" (C00426/4, C00426/5) and the data word length (C00426/1).

Example

Encoder resolution: 100 µm

Parameter setting:

The feed constant is set such that 1 unit corresponds to a machine feed of 1 mm.

- LS_MultiEncoder: traverse path difference: C1111/1 = 1 (unit = mm)
- LS_MultiEncoder: encoder value difference: C1111/2 = 10 (10 × 100 µm 1 mm)

Procedure if the resolution of the encoder is unknown

1. Read the raw value at the outputs *wHighWord* and *wLowWord* of the LS_MultiEncoder.
2. Generate the difference of the raw values for the feed of a known path.
3. Calculate resolution by dividing the known path by the difference.

This approach requires that the SSI telegram must be parameterised correctly. For this purpose, C00426/7 must be parameterised to "32 Bit Singleturn". Thus, the lowest position bit is at bit 0 of the output words and the raw value can be read easily.

- Raw value = 65536 × wHighWord + wLowWord

The data bits for the position data must be detected with the codes C00426/2 (SSI encoder: Bits Singleturn), C00426/3 (SSI encoder: start bit Singleturn) and the correct coding in C00428.

Configuration check of position data

A known encoder resolution and an unknown position of the position bits serve to check the correct configuration of the position data by generating the difference of the raw data for the feed of a known path.

Example

Encoder resolution: 100 µm

Feed path: 1000 mm

Calculated raw value difference: 40000

Expected raw value difference: 10000

The start bit for the Singleturn data is 2 bits too low, 2 bits too many have been detected and included in the calculation.

6.3.4.3 Unipolar/bipolar setting

In case of "unipolar" setting, the data area of the encoder is interpreted from 0 ... $2^{\text{encoder bits-1}}$.

Example of 12 bits Singleturn and 10 bits Multiturn:

- Data area from 0 ... 67108863 incr. (65536 = 1 motor revolution)

In case of "bipolar" setting, the data area of the encoder is interpreted from $-2^{\text{encoder bits/2}}$ to $+2^{\text{(encoder bits/2)-1}}$.

Example of 12 bits Singleturn and 10 bits Multiturn:

- Data area from -33554432 ... +33554431 incr. (65536 = 1 motor revolution)

Caution - special case:

When a singleturn encoder is used, no overflow takes place at the data area limit of the encoder but the full data area is used.

- Unipolar: 0 ... 2147483647 ($2^{31\ -1}$)
- Bipolar: -2147483648 ... 2147483647

6.3.5 Digital frequency coupling

This function extension is available from version 12.00.00!

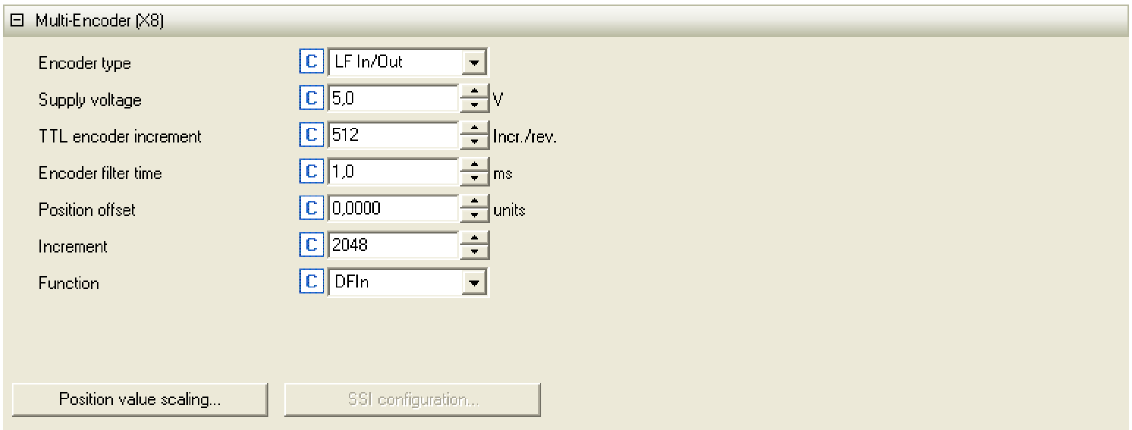
The multi-encoder interface (X8) can also be reconfigured for the output of encoder signals with TTL level (0 ... 500 kHz) in order to realise a "digital frequency coupling".

- The "digital frequency coupling" term describes a digital setpoint transmission and evaluation path between a setpoint source and one or several inverters.

Note!

When the digital frequency coupling is used, the multi-encoder interface is not available anymore for further encoder functions! Only encoder signals with TTL level are permissible as input signals at X8!

Parameterisation dialog (cutout)



Further parameterisation dialogs: ▶ Scaling of position value

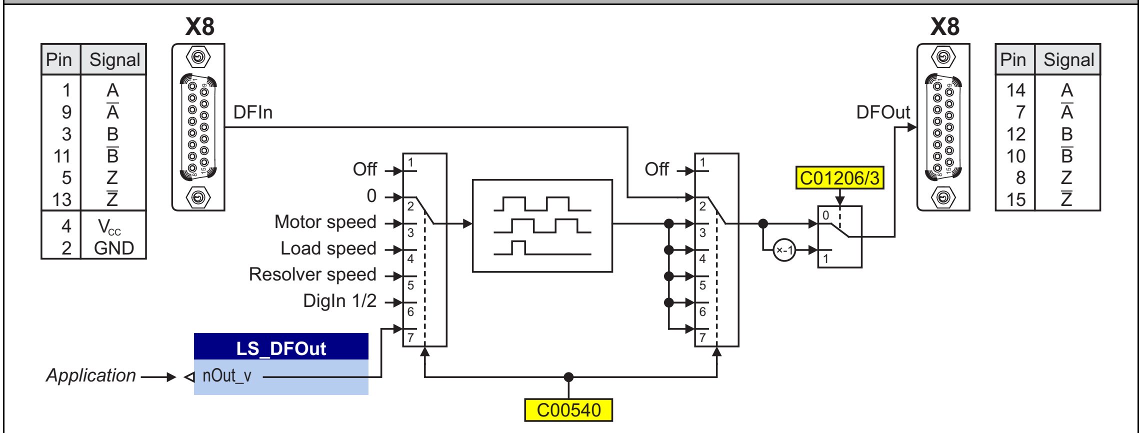
Parameters	Info	Lenze setting Value	Unit
C00422	LS_MultiEncoder: Encoder type	0: Incremental encoder (TTL)	
C00421	LS_MultiEncoder: Supply voltage	5.0	V
C00420/3	LS_MultiEncoder: Encoder increment	512	Incr./rev.
C00497/3	LS_MultiEncoder: Encoder filter time	1.0	ms
C01112/1	LS_MultiEncoder: Position offset	0.0000	units
C00030/1	LS_DFOut: number of increments	2048	
C00540	LS_DFOut: Function	2: DFIn	

How to parameterise the master frequency output:

1. As encoder type (C00422), select "5: LF In/Out" to activate the master frequency output.
2. Set the number of increments for the digital frequency output in C00030/1.

   The number of increments determines after how many output increments a zero pulse will be generated. Each zero pulse defines a covered "revolution" of the rotary transducer simulated by the digital frequency output.
3. Set the desired functions of the digital frequency output in C00540 and C1206/3 (see the following section).

Selecting the functions of the digital frequency output



Function (C00540)	Info
1: Off	Digital frequency output not active • The frequency "0" is output at the digital frequency output. • All tracks remain on the level output last. • After switching on the inverter, the tracks A, B and Z are set to HIGH level.
2: DFIn	The TTL input signals at X8 are connected through to the digital frequency output.
3: MotorSpeed	Output of the motor encoder • The angle of rotation derived from the motor encoder in [increments] is output as frequency signal after being evaluated with the number of increments set in C00030/1.
4: LoadSpeed	Output of the load encoder • The angle of rotation derived from the load encoder in [increments] is output as frequency signal after being evaluated with the number of increments set in C00030/1.
5: Resolver	Output of the resolver angle • The angle of rotation derived from the resolver input in [increments] is output as frequency signal after being evaluated with the number of increments set in C00030/1. • It is irrelevant for the output whether the resolver input is used as load encoder, motor encoder or not at all within the motor control.
6: DigIn 1/2	Output of the digital inputs DI1/DI2 • The angle of rotation derived from the digital inputs in [increments] is output as frequency signal after being evaluated with the number of increments set in C00030/1.
7: AFB input	Output of a speed signal of the application • The speed signal defined via the *nOut_v* input of the LS_DFOut system block is integrated and output as frequency signal after being evaluated with the number of increments set in C00030/1.
Function (C1206/3)	Info From version 17.00.00 onwards
Not inverted	The digital frequency setpoint is not inverted.
inverted	The digital frequency setpoint is inverted.

6.3.5.1 Problem description - speed variations

As digital frequency, rectangular encoder signals are processed. The speed can be determined by counting edges and a defined time interval (1 ms). This defined time interval of 1 ms and a finite number of encoder increments per revolution cause the calculated speed signal to only accept discrete values. This gives the impression of a very disturbed and incorrect signal. This effect increases with a lower number of increments.

Example: An encoder with 2048 increments at a speed of 60 rpm is to be simulated via the digital frequency output.

- Setpoint speed *nOut_v* = 60 rpm = 1 $Hz_{mech.}$
- Simulated encoder: 2048 increments
  (by 4-fold evaluation, 4 * 2048 = 8192 edges are counted per revolution)
- Output frequency = *nOut_v* * number of increments (C00030/1) = 1 $Hz_{mech.}$ * 2048 = 2048 Hz

At a measurement time of 1 ms, 8,192 edges per ms are achieved. As only integer counting processes can occur, sometimes 8 and sometimes 9 increments are counted. Hence, the speed is calculated as follows:

$$n_{mess\_1} = 60\ \text{rpm} \cdot \frac{8}{8.192} = 58.59\ \text{rpm}$$

or

$$n_{mess\_2} = 60\ \text{rpm} \cdot \frac{9}{8.192} = 65.91\ \text{rpm}$$

The perceived speed variation amounts to:

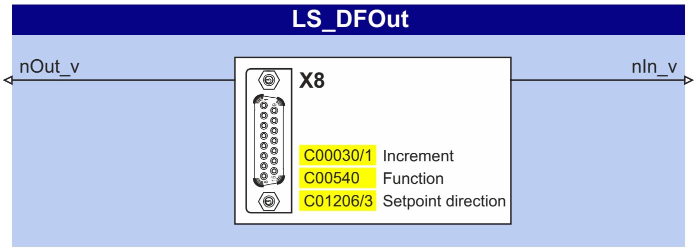
$$\Delta n_{mess} = n_{mess\_2} - n_{mess\_1} = 7.32\ \text{rpm}$$

General formula for calculating the expected speed variation

$$\text{ss} = \frac{15000}{\text{Number of increment}}$$

6.3.5.2 Internal interfaces | System block "LS_DFOut"

The **LS_DFOut** system block maps the digital frequency output in the FB Editor:



inputs

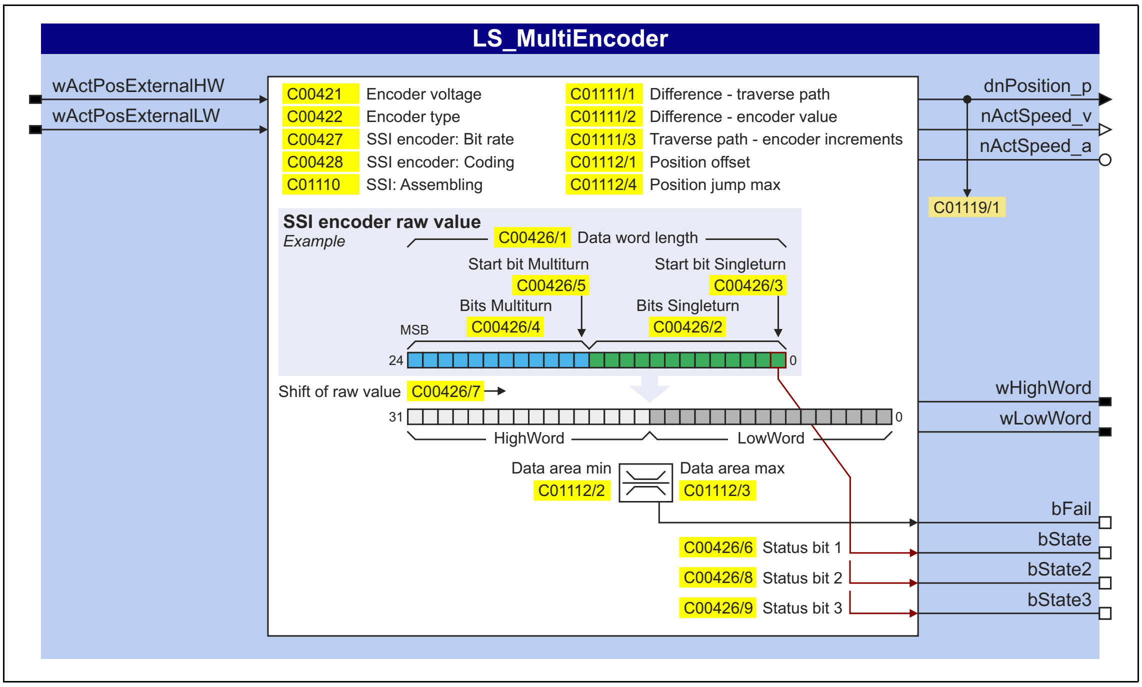
Designator Data type	Information/possible settings
nOut_v INT	Speed in [increments/ms] from the application which is to be output via the digital frequency output X8 in the form of encoder signals with TTL levels. • Scaling: 16384 ≡ 15000 rpm • For selecting this signal source, C00540 must be set to "7: AFB input".

outputs

Designator Data type	Value/meaning
nIn_v INT	Speed in [increments/ms] that has been detected via the digital frequency input X8 in the form of encoder signals with TTL levels. • Scaling: 16384 ≡ 15000 rpm

6.3.6 Internal interfaces | "LS_MultiEncoder" system block

The **LS_MultiEncoder** system block provides the application with speed, position and error information from the encoder connected to the multi encoder interface X8 in the form of process signals.



inputs

Designator Data type	Information/possible settings
wActPosExternalHW wActPosExternalLW WORD (from version 14.00.00)	These inputs serves to feed in the position signal of an external encoder (e.g. via CAN or another fieldbus) and use it as position signal. Scaling: LW = 1 revolution, HW = number of revolutions Thus, an encoder revolution is displayed with 65536 increments.

outputs

Designator Data type	Value/meaning
dnPosition_p DINT	Actual position value from the encoder in [increments]
nActSpeed_v INT	Actual speed from the encoder in [increments/ms] The *nActSpeed_v* output signal is accurate to increments.
nActSpeed_a INT	Actual speed from the encoder in [%] • 100 % ≡ reference speed (C00011) The *nActSpeed_a* output signal is not accurate as to increments and is therefore not suitable for being able to generate a position or path from it!
wHighWord wLowWord WORD	32-bit data telegram from the encoder (raw data) ▸ SSI absolute value encoder with Stegmann-SSI protocol (📖 353) The output is always made in binary format, i.e. in case of Graycode encoders after the received data is decoded. The MSB of the data is at bit 16 of the *wHighWord* output. In order to obtain a standard data format (LSB at the lowest position), an offset can be set in C00426/7. ▸ Left shift of raw value (📖 357)

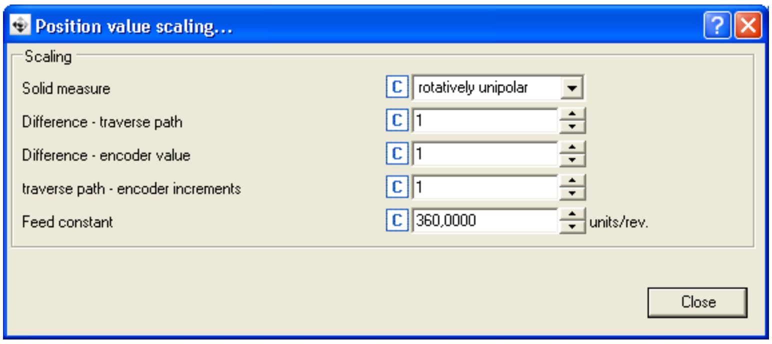
Designator Data type		Value/meaning
bFail BOOL	TRUE	Group error: There is an error (e.g. encoder wire breakage) according to the *bFail* configuration in C00431/1. From version 14.00.00, the error type is displayed bit-coded in C00451/1. ▶ Monitoring
bState BOOL (from version 02.00.00)		Status bit 1 of the encoder • The bit position of the status bit within the 32-bit data telegram can be set in C00426/6. • If C00426/6 = "32", the evaluation of the status bit 1 is deactivated and FALSE is always output at *bState*.
bState2 BOOL (from version 14.00.00)		Status bit 2 of the encoder • The bit position of the status bit within the 32-bit data telegram can be set in C00426/8. • If C00426/8 = "32", the evaluation of the status bit 2 is deactivated and FALSE is always output at *bState2*.
bState3 BOOL (from version 14.00.00)		Status bit 3 of the encoder • The bit position of the status bit within the 32-bit data telegram can be set in C00426/9. • If C00426/9 = "32", the evaluation of the status bit 3 is deactivated and FALSE is always output at *bState3*.

Display parameter

Parameters	Info	Lenze setting Value	 Unit
C01119/1	LS_MultiEncoder: Current position	-	units
C01119/2	LS_MultiEncoder: Maximum travel distance	-	units
Greyed out = display parameter			

6.3.6.1 Scaling of position value

If you click the **Scaling of position value...** button in the parameterisation dialog for the encoder/feedback system in the category "Multi-Encoder (X8), you get to the respective parameterisation dialog:



Short overview of the relevant parameters:

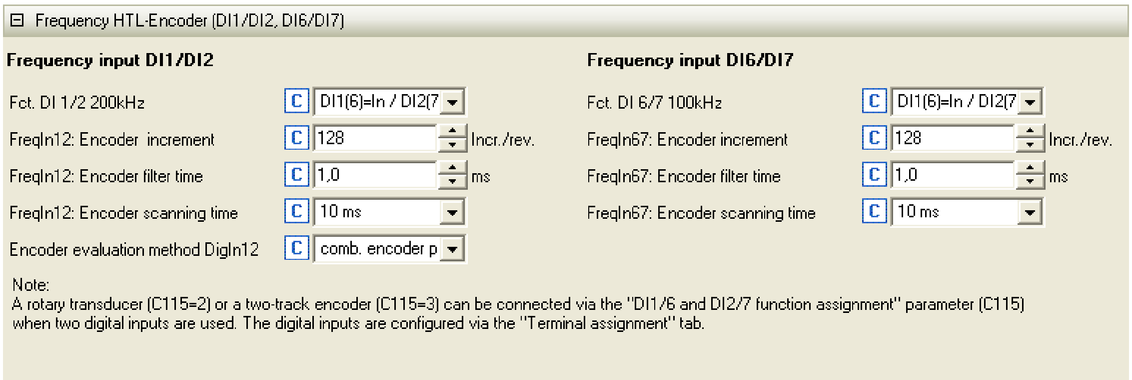
Parameters	Info	Lenze setting	
		Value	Unit
C01110	LS_MultiEncoder: Solid measure • Setting of rotative or linear encoders. **Note:** The selection "rotative" or "linear" only has an impact if the encoder has not been detected automatically or the encoder data definitely are to be taken from the codes (bit 0 in C00499/1 is set to "1"). The "unipolar" or "bipolar" selection definitely has an impact.	0: rotatively unipolar	
Settings for linear SinCos absolute value encoder with HIPERFACE® protocol			
C01111/1	LS_MultiEncoder: Difference - traverse path • Specification of the traverse path while passing through the step number of the Hiperface track given in C01111/2. • Setting is relevant for parameter channel.	1	units
C01111/2	LS_MultiEncoder: Difference - encoder path • Step number of the Hiperface track. • Setting is relevant for parameter channel.	1	Steps
C01111/3	LS_MultiEncoder: Traverse path - encoder increment • Specification of the traverse path while passing through the number of increments of the SinCos track given in C00420/3. • Setting is relevant for process data channel (incremental SinCos track).	1	units
Further settings			
C01204	MCK: Feed constant	360.0000	units/rev.

6.3.6.2 Setting of a static position offset

If, for example, the encoder cannot be mounted mechanically in order that the supplied position matches the mechanics or if for other reasons, a static position offset is required, this value can be set in C01112/1.

6.4 Frequency HTL encoder (DI1/DI2, DI6/DI7)

Parameterisation dialog (cutout)



Parameters	Info	Lenze setting Value	Unit
Settings for HTL encoder at DI1/DI2			
C00115/1	Fct. DI 1/2 200kHz • Function of the digital inputs DI1 and DI2	0: DI1(6)=In / DI2(7)=In	
C00420/1	FreqIn12: Encoder increment • If the digital inputs DI1 and DI2 are used as encoder inputs.	128	Incr./rev.
C00497/1	FreqIn12: Encoder filter time • If the digital inputs DI1 and DI2 are used as encoder inputs.	1.0	ms
C00425/1	FreqIn12: Encoder scanning time • If the digital inputs DI1 and DI2 are used as encoder inputs.	10	ms
C00496	▸ Encoder evaluation method DigIn12 (📖 375)	2: Comb. encoder method	
Settings for HTL encoder at DI6/DI7			
C00115/2	Fct. DI 6/7 100kHz • Function of the digital inputs DI6 and DI7	0: DI1(6)=In / DI2(7)=In	
C00420/2	FreqIn67: Encoder increment • If the digital inputs DI6 and DI7 are used as encoder inputs.	128	Incr./rev.
C00497/2	FreqIn67: Encoder filter time • If the digital inputs DI6 and DI7 are used as encoder inputs.	1.0	ms
C00425/2	FreqIn67: Encoder scanning time • If the digital inputs DI6 and DI7 are used as encoder inputs.	10	ms

6.4.1 Parameterising digital inputs as encoder inputs

The function of the digital inputs DI1/DI2 and DI6/DI7 is defined via C00115/1...2.

To be able to use the digital inputs as encoder inputs, select 2, 3, or 4 (Lenze recommendation: 2) in C00115/1 or C00115/2, depending on the input terminals used.

Selection in C00115/1...2	Function
2: DI1(6)&DI2(7)=FreqIn (2-track)	DI1/6 and DI2/7 = 2-track frequency input • Permits a two-track evaluation of the encoder including correct detection of the direction of rotation.
3: DI1(6)=FreqIn / DI2(7)=Direction	DI1/6 = 1-track frequency input DI2/7 = specification of direction
4: DI1(6)=CountIn / DI2(7)=In	DI1/6 = counter input DI2/7 = digital input

Danger!

- For single-track evaluation, make sure that the sign is correctly specified. Otherwise, the motor may overspeed.
- If servo control (SC) or V/f control (VFCplus + encoder) are used: For safety reasons, always select "Fault" (Lenze setting) in C00586 as a response for the (open-circuit) monitoring of the encoder!
- If an HTL encoder is used at the digital input terminals:
  Observe the maximum input frequencies of the digital inputs!
  - DI1/DI2: max. 200 kHz
  - DI6/DI7: max. 100 kHz

Note!

If the digital inputs are parameterised as encoder inputs, the corresponding output signals (*bIn1*/*bIn2* and *bIn6*/*bIn7*) at the LS_DigitalInput system block are automatically set to FALSE.

Wiring diagram, assignment and electrical data of the digital input terminals can be found in the **hardware manual 8400** in the chapter "technical data". The hardware manual is stored in electronic form on the data carrier supplied with the 8400 inverter.

General procedure

(if the encoder is connected to the digital inputs DI1 and DI2)

1. Define the function of the digital inputs DI1 and DI2 in C00115/1.
2. Set the encoder increments in C00420/1.
3. Select "1: Encoder signal FreqIn12" in C00495/1.
4. Adapt the filter time of the speed measurement in C00497/1.
5. In the case of encoders with a very low resolution (number of increments < 120 increments): Change the encoder evaluation procedure in C00496 if necessary.

Related topics:

- Digital input terminals (📖 401)
- Using DI1(6) and DI2(7) as frequency inputs (📖 405)

6.4.2 HTL encoder at DI1/DI2

Note!

At the digital terminals DI1 and DI2, only encoders with HTL level can be used.

In spite of the selected operating mode without encoder feedback, the actual speed value (C00051) is calculated if an encoder is connected and "1: Encoder signal FrqIn12" is selected in C00495.

Encoder evaluation method DigIn12

Depending on the encoder used at the digital inputs DI1 and DI2, the following table specifies which evaluation method should be selected in C00496:

Selection in C00496	Encoder evaluation method

0: High-resolution encoder	High-precision procedure for high-resolution encoders (>=512 increments)
	• Method for speed measurement with automatic scan time setting (0.5 ... 500 ms).
	• Evaluation with automatic scanning time minimisation for an optimum dynamic performance.
	• Particularly suited for high-resolution encoders (≥ 1024 inc) with good signal quality, i.e.
	  • good scanning ratio 1:1
	  • exactly 90°-phase offset between track A and B (error ≤ ± 10°)
	• Not suited for encoders with poor signal quality.
	• Wiring according to EMC (e.g. motor and encoder cable shielding) is required!

1: Low-resolution encoder (StateLine)	High-precision procedure for low-resolution encoders (<=128 increments) • Exact method for speed measurement with automatic scanning time setting (0.5 ... 500 ms) for low-resolution encoders in the range of 4 .... 128 increments. • Evaluation with automatic scanning time minimisation for an optimum dynamic performance. • Method is also suited for encoders with poor signal quality, e.g. for encoders with high error rate in scanning ratio and phase offset. • This method requires an equidistant period length per encoder increment. • Wiring according to EMC (e.g. motor and encoder cable shielding) is required!
2: Comb. encoder method (Lenze setting)	Combination of the first two procedures as a function of the speed (recommended procedure) • For a high-precision speed measurement suited for encoders with an arbitrary number of increments (4 ... 1024 increments). • Low input frequencies at the encoder inputs: The method is used for low-resolution encoders. • High input frequencies at the encoder inputs: The method is used for high-resolution encoders. • This method is suited for encoders with average to good signal quality. • Evaluation with automatic scanning time minimisation for an optimum dynamic performance. • This method requires an equidistant period length per encoder increment. • Wiring according to EMC (e.g. motor and encoder cable shielding) is required!
3: Edge-counting procedure	Simple edge counting procedure with adjustable scanning time (C00425) • Speed measurement by means of the edges of tracks A and B measured per scanning interval. • Integrated correction algorithm for EMC interference. • Limited suitability for systems with unshielded encoder and/or motor cable. • Limited suitability for encoders with poor signal quality, i.e. high error rate in scanning ratio and phase offset.

Tip!

- We recommend to use the preset combined encoder method (C00496 = 2).
- Use one of the first three procedures (C00496 = 0, 1, or 2) for dynamic applications (e.g. operating mode: servo control).
- For dynamic speed control or positioning processes, use an HTL encoder with 1024 increments.

Low speeds (except for edge counting)

For the first three methods (C00496 = 0, 1, or 2), the minimum speed that can be measured depends on the encoder resolution.

The quantisation error

- is independent of the encoder resolution,
- exclusively depends on the encoder quality (encoder errors).
- at least amounts to 0.5 rpm.

Internal arithmetic operations automatically maintain the minimally required value of the scanning time in order to achieve maximum dynamics.

Encoder resolution (Number of increments)	Min. measurable speed in [rpm]
8	16
16	8
32	4
64	2
128	1
256	0.5
≥ **512**	0.25

Low speeds with edge counting

The minimum speed that can be measured and the quantisation error of speed measurement in the edge-counting procedure (C00496 = 3) depend on the scanning time that can be set in C00425/1 and the encoder resolution.

Depending on accuracy and the requirements with regard to the dynamic performance, the respective scanning time must be selected and set in C00425/1:

Encoder resolution (Number of increments)	Scanning time [ms]									
	1	**2**	**5**	**10**	**20**	**50**	**100**	**200**	**500**	**1000**
	Min. measurable speed in [rpm]									
8	1875	938	375	188	93.8	37.5	18.8	9.4	3.8	1.9
16	938	469	188	94	46.9	18.8	9.4	4.7	1.9	0.9
32	469	234	94	46.9	23.4	9.4	4.7	2.3	0.9	0.5
64	234	117	46.9	23.4	11.7	4.7	2.3	1.2	0.5	0.2
128	117	58.6	23.4	11.7	5.9	2.3	1.2	0.6	0.2	0.12
256	58.6	29.3	11.7	5.9	2.9	1.2	0.6	0.3	0.12	0.06
512	29.3	14.6	5.9	2.9	1.5	0.6	0.3	0.15	0.06	0.03
1024	14.6	7.3	2.9	1.5	0.7	0.3	0.15	0.07	0.03	0.01

6.4.3 HTL encoder at DI6/DI7

Note!

Single-track evaluation of the digital DI6/DI7 terminals as speed feedback (C0115/2 = 1 or 3) is not possible. Hence, a single-track encoder cannot be used for speed control at the digital DI6 terminal!

Low speeds with edge counting

The speed measurement is evaluated at the digital terminals DI6/DI7 with the edge-counting procedure and a fixed scanning time that can be set in C00425/2.

The minimum speed that can be measured and the quantisation error of speed measurement in the edge-counting procedure depend on the scanning time that can be set in C00425/2 and the encoder resolution.

Depending on accuracy and the requirements with regard to the dynamic performance, the respective scanning time must be selected and set in C00425/2:

Encoder resolution (Number of increments)	Scanning time [ms]									
	1	2	5	10	20	50	100	200	500	1000
	Min. measurable speed in [rpm]									
8	1875	938	375	188	93.8	37.5	18.8	9.4	3.8	1.9
16	938	469	188	94	46.9	18.8	9.4	4.7	1.9	0.9
32	469	234	94	46.9	23.4	9.4	4.7	2.3	0.9	0.5
64	234	117	46.9	23.4	11.7	4.7	2.3	1.2	0.5	0.2
128	117	58.6	23.4	11.7	5.9	2.3	1.2	0.6	0.2	0.12
256	58.6	29.3	11.7	5.9	2.9	1.2	0.6	0.3	0.12	0.06
512	29.3	14.6	5.9	2.9	1.5	0.6	0.3	0.15	0.06	0.03
1024	14.6	7.3	2.9	1.5	0.7	0.3	0.15	0.07	0.03	0.01

Maximum speeds with edge counting

Due to the lower maximum input frequency compared to the terminals DI1/DI2 (200 kHz), operation with maximum speed at the terminals DI6/DI7 (100 kHz) is limited. ▶ Digital input terminals (🕮 401)

Encoder resolution (Number of increments)	Max. measurable speed in [rpm]
8	No restrictions
16	37500
32	18750
64	9375
128	4688
256	2344
512	1172
1024	586

6.4.4 4-track evaluation of an HTL encoder

This function extension is available from version 15.00.00!

From version 15.00.00 onwards, an HTL encoder can also be evaluated on 4 tracks. Advantages:

- Better open-circuit monitoring.
- In the lower speed range, the levels of the single tracks are checked logically.
- In the upper speed range, the frequencies measured at the digital input pairs are compared to each other.

Typical terminal assignment:

- DI1: track A
- DI2: track B
- DI6: track A inverted ("A not")
- DI7: track B inverted ("B not")

Parameter setting:

- Set C00115/1 = "2: DI1(6)&DI2(7)=FreqIn (2-track)"
- Set C00115/2 = "2: DI1(6)&DI2(7)=FreqIn (2-track)"
- Enter the correct encoder increments In C00420/1 and C00420/2.
- Set C00495 = "5: encoder signal FreqIn1267" to activate the evaluation of all tracks.

6.5 Pole position identification (PPI)

Note!

Only required:
- For servo control (SC) with synchronous motor of a third-party manufacturer.
- For servo control (SC) with synchronous motor and use of incremental encoders (TTL or sin/cos encoders as well as multi-pole pair resolvers).
- After changes of the motor feedback system, e.g. encoder exchange.

For the sensorless control of synchronous motors (SLPSM), a pole position identification is not required.

Note!

Acceptance of the resolver pole position from a Servo Drive 9400

The resolver pole position (C00926/1) cannot be simply accepted from a Servo Drive 9400 if the pole position (C58/1) considerably differs from -90 ° in the Servo Drive 9400.
- We always recommend a pole position identification (PLI).
- As an alternative, the following conversion has to be carried out for a transfer of the resolver pole position from a Servo Drive 9400 to the 8400 TopLine:
  Pole position$_{8400}$ (C00926/1) = - (pole position$_{9400_C58/1}$ + 180 °)

For the control of permanent-magnet synchronous machines, the pole position – the angle between the motor phase U and the field axis of the rotor – must be known.

- For Lenze motors with absolute value encoder or resolver, the pole position is already set correctly in C00926/1...2.
- When incremental encoders (TTL or sin/cos encoders) are used, a pole position identification (PPI) is always required after mains switching, even with Lenze motors.
- The inverter can also evaluate multi-pole-pair resolvers.
  - When the number of motor pole pairs is an integer multiple of the number of pole pairs of the resolver, a pole position identification must only be executed once.
  - When the number of motor pole pairs is no integer multiple of the number of pole pairs of the resolver, a pole position identification must be executed after every mains switching.

Parameterisation dialog (cutout)

Parameters	Info	Lenze setting
C00002/34	Pole position identification (360°)	0: Off / ready

The **Identify pole position...** button or the "pole position identification (360°)" device command serve to detect the pole position for the motor encoder currently activated in C00495.

Detailed information on the process and execution of the pole position identification can be found in the following subchapter "Pole position identification 360°".

6.5.1 Pole position identification 360°

Procedure for "pole position identification 360°"

If all conditions are met, the motor is energised with a direct current corresponding to the lower of the following two values:

$\sqrt{2}$ · Rated device current
or
$\sqrt{2}$ · Rated motor current

- The rotor is aligned through the current flow. This is absolutely necessary for the procedure.
- To ensure that the torque-neutral axis is not accidentally energised and the rotor stops, a 45° current vector is (electrically) generated for a short instant and then (electrically) switched back to 0° (≡ phase U).
  - Then a DC current of the above-mentioned value could be measured in this motor phase.
- If a resolver or an optical encoder without absolute track is used, the difference between the preselected current angle and the mechanical rotor angle is determined. After this, the current vector is (electrically) turned by another 22.5° and the difference between current angle and rotor angle is determined once again.
  - The procedure is repeated 16 times. This corresponds to one electrical revolution. The machine rotates by 360° (mech.)/pole pair number.
  - Take the average value of the 16 measurements to compensate for asymmetries.

Adjustment of the pole position identification 360°

The pole position identification can be adjusted to the respective machine and the prevailing moments of inertia by means of the parameters described below:

Parameters	Info	Lenze setting	
		Value	Unit
C00644/1	PLI 360° traversing direction	right rotating field	
C00645/2	PLI 360° max. error tolerance	0.0	°
C00646/2	PLI 360° current amplitude	100	%
C00647/2	PLI 360° ramp time	100	%

- The current amplitude can be adjusted proportionally in C00646/2.
  - For large machines and high mass inertia values or for linear direct drives, the current amplitude usually has to be increased.
  - The Lenze setting "100 %" corresponds to the smaller of the two following values:

$\sqrt{2}$ · Rated device current
or
$\sqrt{2}$ · Rated motor current

Stop!

If there is no temperature monitoring in the motor and/or the I2xt motor monitoring and the maximum current monitoring are not parameterised correctly, the motor may be permanently damaged when the current amplitude is set too high (e.g. to the maximum value)!

- Motor load monitoring (I2xt) (310)
- Maximum current monitoring (320)

Note!

If the current amplitude is set to > 100 % in C00646/2, the device utilisation (Ixt) monitoring and/or one of the motor monitoring functions may respond and cause the abort of the pole position identification.

- The ramp time can be adjusted proportionally in C00647/2.
  - For large machines and high mass inertia values, the ramp time usually has to be increased.
  - For small machines, a reduction of the ramp time can speed up the pole position identification process.
- In some situations it may be helpful to reverse the travel direction in C00644/2 for the pole position identification (e.g. for linear motor at the end stop).
- The "pole position identification 360°" procedure comprises a plausibility check. If the rotor position determined via the encoder system does not correspond to the controlled output position:
  - the pole position identification procedure is aborted.
  - the response parameterised in C00643/1 (Lenze setting: "Fault") is activated.
  - the error message "Id5: Pole position identification error" is entered into the logbook.
- The preset fault tolerance for the plausibility check can be changed in C00645/2.

Execute pole position identification 360°

Danger!

The machine must not be braked or blocked during the pole position identification! For this reason, the pole position identification is not permitted for hanging loads!

During the pole position identification the rotor aligns itself. The motor shaft moves by max. one electrical revolution which causes the corresponding movement of the connected mechanical components!

Stop!

Check the correct parameterisation of the max. motor current monitoring before carrying out the pole position identification to prevent the motor from being permanently damaged.

▶ Maximum current monitoring (📖 320)

Note!

By means of controller inhibit, the procedure started can be cancelled anytime, if required, without carrying out a change in C00926.

If the pole position identification is aborted, the response parameterised in C00643/1 is activated (Lenze setting: "Fault").

- If this behaviour is not wanted, deactivate the monitoring by selecting "0: No response" in C00643/1.

Preconditions for the execution

- A synchronous motor has been selected.
- No other identification is active.
- No error has occurred.

How to execute the pole position identification:

1. Inhibit the inverter if it is enabled, e.g. via the C00002/16 device command or a LOW signal at the X5/RFR terminal.
2. Start the pole position identification with the device command C00002/34 = "1: On / start".
3. Inverter is re-enabled.
   - The pole position identification starts.
   - The progress of the identification run can be seen in C00002/34.
   - The identification is completed if the "0: Off / ready" message is displayed in C00002/34.
4. Inhibit inverter again.

After successful completion...

...the controller is inhibited automatically and the pole position determined for the activated feedback system is set in the corresponding subcode of C00926.

- For a permanent acceptance of the identified pole position, the parameter set must be saved (C00002/11 = "1: On / start").
- The next controller inhibit and subsequent controller enable serve to cancel the controller inhibit automatically set by the procedure (e.g. by first executing the device command C00002/16 = "0: Off / ready" and then executing the device command C00002/16 = "1: On / start").

In the event of an error

If an error occurs during the procedure or the pulse inhibit gets active (e.g. due to short-time undervoltage), the procedure is terminated with controller inhibit without making a change in C00926.

If the machine was braked or blocked during the procedure, this will be recognised at the end of the measurement and no change is made in C00926.

- If the pole position identification is aborted:
  - the response parameterised in C00643/1 (Lenze setting: "Fault") is activated.
  - the error message "Id5: Pole position identification error" is entered into the logbook.

6.5.2 Pole position identification without motion

This function extension is available from version 02.00.00!

Note!

Function only possible with:
- Servo control (SC) for synchronous motors
- Sensorless control for synchronous motors (SLPSM)

From version 02.00.00, a pole position identification without motion is also possible in case of servo control and sensorless control.

- For sensorless control of synchronous motors (SLPSM), this function is already activated in the Lenze setting, i.e. with every controller enable the rotor displacement angle is identified and thus jerks in the machine after controller enable can be avoided.
  - In order to achieve the same behaviour as before, set bit 0 to "0" in C02874.
- For servo control (SC), this function can be activated for various events in C02874 via the bits 1 ... 3:
  - Bit 1: Pole position identification after mains connection
  - Bit 2: Pole position identification after controller enable
  - Bit 3: Pole position identification after error acknowledgement

Note!

- The "Pole position identification without motion" cannot completely replace the "pole position identification 360°"!
  - The electrical rotor displacement angle can only be electrically identified with an error of up to 10°. This inaccuracy can cause a worse torque accuracy and a worse energy efficiency.
- The identification takes motor-dependent 1 ... 15 ms. The setpoint enabling of the device is reduced by this time.
- The default values of the function in the Lenze setting ensure that, in most cases, it is not necessary to make any further settings.

Tip!

In case of servo control (SC), it is generally sufficient to activate this function only once after mains connection of the device (C02874: Bit 1 = "1").

Typical application case: A speed sensor with an unknown pole position is used (e.g. encoder). The pole position identification 360°, however, cannot be used since, e.g. the motor is locked, the application does not permit it or the identification takes too long.

Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value	Unit
C02874	PLI without movement	0x1 (bit coded)	
C02872	PLI without movement: adaptation of time duration	0	
C02875	PLI without movement: adaptation of ident angle	0	°
C02870	PLI without movement: degree of optimisation	-	%
C02871	PLI without movement: runtime	-	ms
C02873	PPI without motion: Identified rotor displacement angle	-	°
Greyed out = display parameter			

Note!

For synchronous motors with a stator time constant < 1 ms, the pole position identification is not carried out because the resulting test current pulse may exceed the permissible motor current.

- This, however, concerns only few synchronous motors with a very low power (e.g Lenze motor MDSKS-020-13-300 with a rated power of 40 W).
- A pole position identification that has not been carried out is indicated by C02870 = 0 % and C02871 = 0 ms.

- The stator time constant can be calculated on the basis of the following formula:

$$T_s[\mathrm{ms}] = \frac{L_{ss}[\mathrm{mH}]}{R_s[\Omega]}$$

T_S = stator time contant
L_{ss} = Motor stator leakage inductance (C00085)
R_s = Motor stator resistance (C00084)

Optimising the pole position identification

Stop!

When the setting in C02872 is too high, an impermissible motor current may flow whilst the pole position identification is carried out. In this case, the "Fault" error response is triggered, and the "Id5: pole position identification error" error message is entered into the logbook.

With a setting in C02872 that is excessively high:

- The following other current monitoring functions may be activated:
  - oC7: Motor overcurrent
  - oC11: Clamp operation active
  - oC1: Power section - short circuit
- In C02870, the degree of optimisation "0 %" is shown.
- In C02871, the time period "0 ms" is shown.

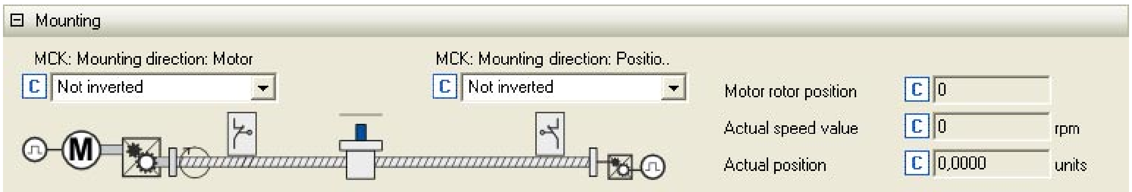
How to optimise the pole position identification without movement:

1. For the optimisation, execute controller enable at different rotor displacement angle.
2. After every controller enable, check the degree of optimisation shown in C02870.

   The pole position identification is set optimally if a degree of optimisation in the range of 70 ...130 % is displayed in C02870 after every controller enable.
3. If the degree of optimisation is > 130 %:
   reduce the setting in C02872 step by step and execute controller enable at different rotor displacement angles until a degree of optimisation < 130 % is shown.
4. If the degree of optimisation is < 70 %:
   increase the setting in C02872 step by step and execute controller enable at different rotor displacement angles until a degree of optimisation > 70 % is shown.
5. Optionally: via C02875, the electrical rotor displacement angle identified can be increased or reduced. Due to the accuracy of the identification, this can for instance serve to prevent the motor from rotating backwards, if this is required by specific applications.

6.6 Mounting

In case of an inverted mounting direction of the speed sensor and/or position encoder, the following parameters have to be adapted accordingly:



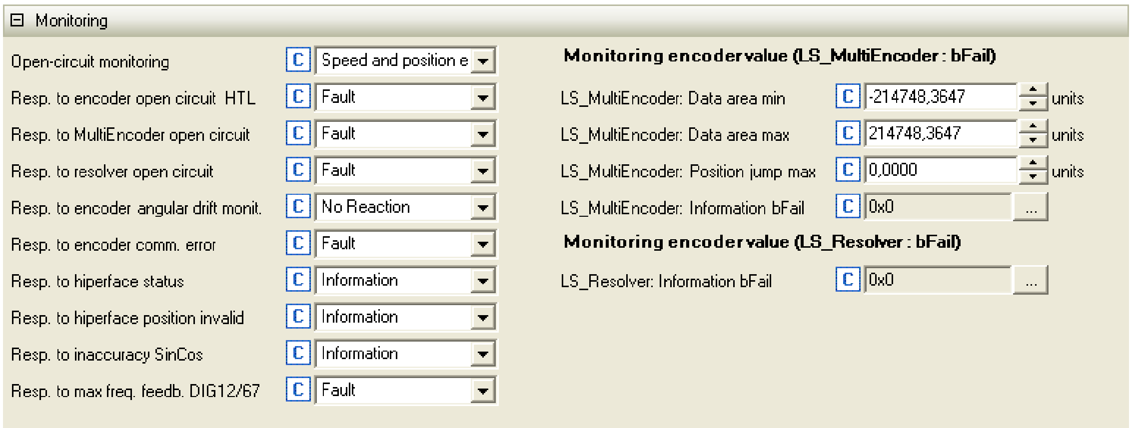
Parameters	Info	Lenze setting Value	Unit
C01206/1	MCK: Mounting direction: Motor	0: Not inverted	
C01206/2	MCK: Mounting direction: Position encoder	0: Not inverted	
C00927	Motor rotor position	-	°
C00051	MCTRL: Actual speed value	-	rpm
C01210/3	MCK: Actual position	-	units
Greyed out = display parameter			

Related topics:

- Machine parameters

6.7 Monitoring

Parameterisation dialog (cutout)



Parameters	Info	Lenze setting Value	Unit
C00498	Open-circuit monitoring	0: Speed encoder and position encoder	
C00586	Resp. open circuit HTL encoder	1: Fault	
C00603/1	Resp. to MultiEncoder open circuit	1: Fault	
C00603/2	Resp. to resolver open circuit	1: Fault	
C00603/3	Up to and including version 02.xx.xx: Resp. to encoder comm. error	1: Fault	
	From version 12.00.00: Resp. to encoder pulse deviation	0: No Reaction	
C00603/4	Up to and including version 02.xx.xx: Resp. to encoder pulse deviation	1: Fault	
	From version 12.00.00: Resp. to encoder comm. error	1: Fault	
C00603/5	Resp. to Hiperface status	6: Information	
C00603/6	Resp. to invalid Hiperface position	6: Information	
C00603/7	Resp. to SinCos inaccuracy	6: Information	
C00607	Resp. to max. freq. feedb. DIG12/67	1: Fault	
Monitoring of encoder value (LS_MultiEncoder: bFail)			
C01112/2	LS_MultiEncoder: Data area min	-214748.3647	units
C01112/3	LS_MultiEncoder: Data area max	214748.3647	units
C01112/4	LS_MultiEncoder: Max. position jump	0.0000	units
C00451/1	LS_MultiEncoder: Information bFail	-	
Monitoring of encoder value (LS_Resolver: bFail)			
C00452/1	LS_Resolver: Information bFail	-	
Greyed out = display parameter			

6.7.1 Resolver/multi-encoder open-circuit monitoring

In the Lenze setting, the resolver cable and/or encoder cable are monitored with regard to open circuit as a function of the selected speed and position encoder.

Danger!

- For safety reasons, always select "Fault" (Lenze setting) as a response for the (open-circuit) monitoring of the resolver/encoder!
- To avoid the injection of interference when an encoder is being used, only use shielded motor and encoder cables!

Tip!

We recommend the Lenze setting "0: Speed sensor and position encoder" for open-circuit monitoring (C00498). Further possible settings make sense if the respective encoder is not used as speed sensor or position encoder but for other tasks in the application. Moreover, a selective switch-off of the monitoring is possible if other settings are made.

Open-circuit monitoring (C00498)	Function
0: Speed encoder and position encoder (Lenze setting)	Open-circuit monitoring is active for the speed encoder selected in C00495 and the position encoder selected in C00490. • Depending on whether a speed encoder or position encoder has been selected, the monitoring mode for the resolver and/or encoder is activated. • If no speed encoder and position encoder have been selected, open-circuit monitoring is deactivated.
1: Resolver only	Open-circuit monitoring is only active for the resolver, independent on the selection of the speed encoder and position encoder. • Open-circuit monitoring for the encoder is deactivated.
2: Encoder only	Open-circuit monitoring is only active for the encoder, independent on the selection of the speed encoder and position encoder. • Open-circuit monitoring for the resolver is deactivated.
3: Resolver and encoder	Open-circuit monitoring is active for the resolver and encoder, independent on the selection of the speed encoder and position encoder.

When does the open-circuit monitoring system respond?

Resolver	Multi encoder
• If there is an open circuit in the encoder cable. • If the impedance of the resolver is too high. • In the case of interference injections (EMC interferences).	• If there is an open circuit in the encoder cable.

Response to open circuit

Resolver	Multi encoder
The error response set in C00603/2 is triggered (Lenze setting: "0: No Reaction"). Logbook entry: "Sd2: Open circuit - resolver"	The error response set in C00603/1 is triggered (Lenze setting: "0: No Reaction"). Logbook entry: "Sd4: Open circuit - MultiEncoder"

6.7.2 Open-circuit monitoring - SSI encoder

The following functions can be used to monitor a correctly connected SSI encoder.

Monitoring of plausibility of the position value provided by the encoder

The position provided by the encoder must be in a position window limited by C1112/2 and C1112/3. Define this window in such a way that it tightly frames the possible traversing range. In case of wire breakage in the encoder cable or beam interruption in a laser measurement system, a position value is determined that corresponds to the maximally possible binary value and is thus outside the defined window.

- When a binary-coded SSI encoder is used and an error occurs, the value results in e.g. 0x1FFFFFF at 25 data bits.
- When a gray-coded SSI encoder is used and an error occurs, the value results in e.g. 0xAAAAAA80 at 25 data bits.
- When you exit the defined position window, *LS_MultiEncoder: bFail* is reported at the output. For this purpose, the bit 2 (position outside C1112/2..3) must be set in C431/1.

Note!

This method cannot be used if a binary-coded encoder is evaluated in bipolar mode. In the event of an error, the detected output value is within the valid traversing range. In this case, a Gray-coded encoder provides a significantly deviating position value.

Monitoring of the speed value provided by the encoder

The position jump of the two last different positions detected by the encoder is determined. This position jump must not exceed the value defined in C1112/4. In case of wire breakage in the encoder cable or beam interruption in a laser measurement system, an inadmissibly high position jump is recorded and output as an error.

- The record of a too high position jump is reported at the *LS_MultiEncoder: bFail* output. For this purpose, the bit 7 (max. encoder speed) must be set in C431/1.

Note!

This method cannot be used if a binary-coded encoder is evaluated in bipolar mode. In the event of an error, the encoder may be positioned very close to position "-1" which corresponds to the detected output value. A risk assessment is required. In this case, a Gray-coded encoder provides a significantly deviating position value.

Tip!

The error messages can also be read out in C451/1 without the need of setting bit 121 or bit 7 in C431/1.

6.7.3 Open-circuit monitoring - HTL encoder

Open-circuit monitoring is available for the HTL encoder cable.

Danger!

For safety reasons, always select "Fault" (Lenze setting) as a response for the (open-circuit) monitoring of the HTL encoder!

Note!

- The 4-track evaluation of the HTL encoder provides better open-circuit monitoring.
  4-track evaluation of an HTL encoder (378)
- If you do not use an HTL encoder, deactivate the monitoring (C00586 = "0: No Reaction").

When does the open-circuit monitoring system respond?

The open-circuit monitoring will trigger if

- an open circuit occurs in the encoder cable.
- an extreme overload (e.g. blocked motor shaft) occurs during the start-up phase of the motor.
- highly dynamic reversion of the motor occurs.
- the motor still rotates at high speed in the "SC servo control ASM" mode when the controller enabled.

Which measured values lead to an actuation of the open-circuit monitoring system?

The following measured values checked for plausibility lead to an actuation of the open-circuit monitoring system:

1. If the total deviation between actual speed and setpoint speed is higher than f = 40 Hz for a time > 0.1 s .
2. If the actual speed detected is f = 0 Hz or n = 0 rpm and the I_{max} controller or the torque limitation for servo control (SC) is active for t ≥ 0.1 s.
3. If the sign of the injected frequency and the actual speed is not the same, the I_{max} controller is active and this status is active for 0.1 s. Usually this is the case when A/B tracks are reversed.

Response to open circuit

- If the open-circuit monitoring is tripped:
  - The error response set in C00586 is activated (Lenze setting: "Fault").
  - The "Sd3: Open circuit - feedback system" is entered into the Logbook.
  - The *bMctrlEncoderComFault* status output of the LS_DeviceMonitor SB is set to TRUE.
- A setting of C00586 = "0: No Reaction" deactivates the monitoring.

6.7.4 Encoder angular drift monitoring

This function extension is available from version 12.00.00!

Note!

The encoder angular drift monitoring is implemented for encoders without absolute information.

In the Lenze setting (C00603/3 = "0: No response"), the angular drift monitoring of the encoder is not activated!

The encoder angular drift monitoring monitors a possible deviation of the real encoder angle from the angle calculated by counting increments in the encoder evaluation.

- If a deviation higher than 45° (electrical) is recognised when monitoring is activated:
  - The error response set in C00603/3 is triggered.
  - The "Sd8: Encoder angular drift monit." error message is entered into the logbook.
  - The "Reference known" status of the "Homing" basic drive function is reset (if this status was set before)

Tip!

A deviation may occur, for instance, by incorrect parameter setting of the encoder increments, by lines in the form of interferences caused by EMC or loss of lines caused by EMC.

Functional principle

For an encoder without absolute information, the number of incoming encoder lines between two zero pulses (one revolution) is monitored. This value must equal the encoder increments set in C00420/3 .

Note!

After mains switching, monitoring is only active after second incoming zero pulse since the first line difference to be used can only be calculated with the second and first zero pulse.

When the motor (and thus the encoder) is replaced, it is very likely that a angular drift error occurs within the first revolution after acknowledging the encoder error since the monitoring function cannot recognise that the encoder has been replaced.

Due to the type of monitoring, accuracy (distance of Z pulse to the detected position) can only be detected while the machine is running. As a permanently pending error would prevent the error cause to be removed, a detected error will be automatically reset by the inverter after a short time. In order that the motor does not continue to rotate in this case, an engaging error response (e.g. "Fault" or "TroubleQSP") has to be set. If an error is detected again, it is entered again into the logbook.

6.7.5 Encoder value monitoring

For detecting invalid encoder values, a value range can be defined via the parameter C01112/2 and C01112/3. When the encoder signal leaves the set value range, the *bFail* status signal of the SB LS_MultiEncoder is set to TRUE and the corresponding information is displayed bit-coded in C00451/1.

- **Example 1: SSI laser distance meter**
  The *bFail* status signal can be used to detect an interruption of the laser beam or a dirty mirror.
- **Example 2: SSI bar code scanner**
  In an application, an SSI bar code scanner is used to detect the workpiece ID and the workpiece position. A setting of sensible limit values serves to use the *bFail* status signal to determine whether the ID or the position of the SSI bar code scanner has been detected.

6.7.6 Motor temperature monitoring (PT1000 or KTY)

The motor temperature can also be detected and monitored via the resolver cable and/or the encoder cable.

- The feedback system to be used for the motor temperature is selected in C01193.
- The respective motor temperature monitoring includes an early warning stage as well as an open-circuit monitoring for the thermal sensor.

Note!

In the future, Lenze motors will only be equipped with motor temperature sensors PT1000. This is taken into account in the current motor catalogue for »EasyStarter« or »Engineer«.

When motors are replaced in existing plants or in new plants with previous applications, please check whether a motor with KTY or PT1000 is used (motor nameplate or order designation). If a motor with PT1000 is used, adapt the parameterisation of the device.

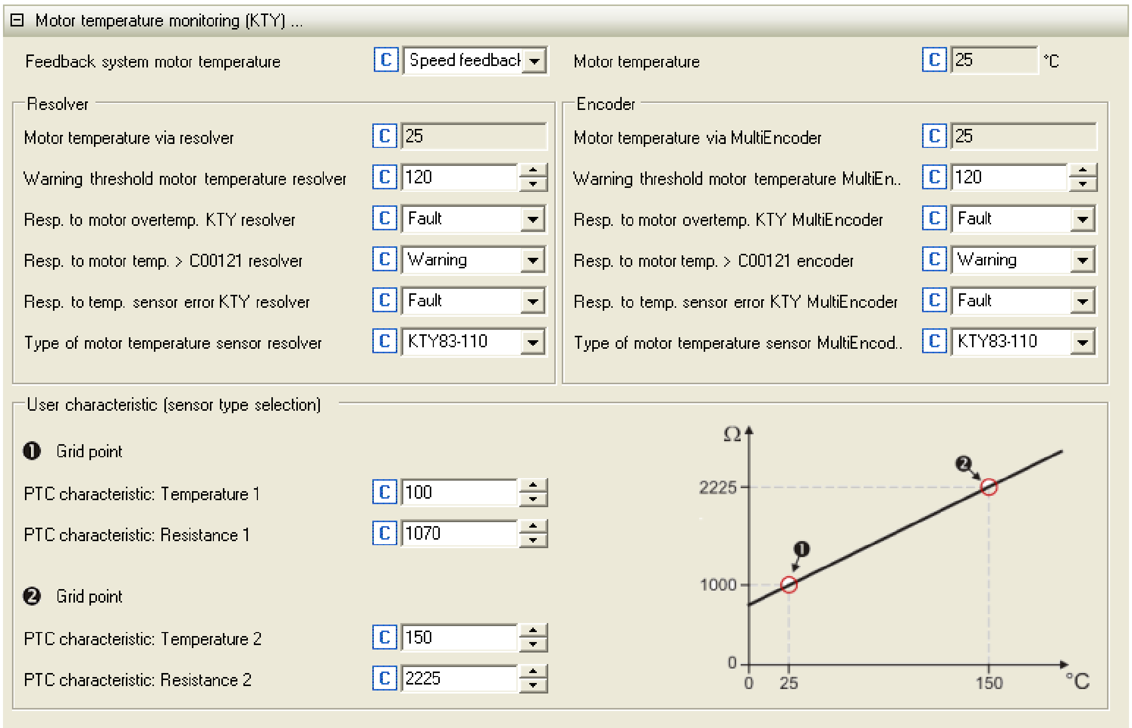
- Up to version V16.xx.xx, you must use the Specific characteristic for the motor temperature sensor.
- From version V17.00.00 onwards, you can select PT1000 in C01190.

Note!

In the Lenze setting of C00583/1...6, the motor temperature monitoring functions are switched on! These monitoring functions are only active with speed encoder selection "3: Multi encoder" or "4: Resolver" in C00495.

From version V02.00.00 with servo control (SC), the temperature compensation within the motor control is activated in the Lenze setting of C02878/1. However, temperature compensation is only active with speed sensor selection "3: Multi encoder" or "4: Resolver" in C00495 PT1000 as well as error-free KTY temperature detection (display in C00063/1 ≠ 255 °C).

Parameterisation dialog (cutout)



Parameters	Info	Lenze setting Value	Unit
C01193	Motor temp. feedback system	0: Speed feedback	
C00063/1	Motor temperature	-	°C
Resolver			
C00063/2	Motor temperature via resolver	-	°C
C00121/1	Warning threshold motor temperature resolver	120	°C
C00583/1	Resp. to motor overtemp. KTY resolver	1: Fault	
C00583/3	Resp. to motor temp. > C00121 resolver	5: Warning	
C00583/5	Resp. to temp. sensor error KTY resolver	1: Fault	
C01190/1	Type of motor temperature sensor resolver	0: KTY83-110	°C
Multi encoder			
C00063/2	Motor temperature via MultiEncoder	-	°C
C00121/2	Warning threshold motor temperature MultiEncoder	120	°C
C00583/2	Resp. to motor overtemp. KTY MultiEncoder	1: Fault	
C00583/4	Resp. to motor temp. > C00121 encoder	5: Warning	
C00583/6	Resp. to temp. sensor error KTY MultiEncoder	1: Fault	
C01190/2	Type of motor temperature sensor MultiEncoder	0: KTY83-110	°C
User characteristics (sensor type selection)			
C01191/1	PTC characteristic: Temperature 1	100	°C
C01191/2	PTC characteristic: Temperature 2	150	°C
C01192/1	PTC characteristic: Resistance 1	1070	Ohm
C01192/2	PTC characteristic: Resistance 2	2225	Ohm
Greyed out = display parameter			

Motor temperature monitoring (PT1000 or KTY) via resolver

The sensor type has to be selected in C01190/1.

- If the winding temperature detected via the motor temperature sensor reaches the limit value set in C00121/1 (Lenze setting: 120 °):
  - An advance warning in the form of the error response set in C00583/3 is issued.
  - The "oH7: Motor temperature - resolver > C121" error message is entered into the logbook.
- If the fixed limit value of 150 °C is reached:
  - The error response set in C00583/1 is triggered.
  - The "oH9: Motor temperature - resolver" error message is entered into the logbook.
- If open circuit is detected for the motor temperature sensor:
  - The error response set in C00583/5 is triggered.
  - The "Sd6: Thermal detector error - resolver" error message is entered into the logbook.

Motor temperature monitoring (PT1000 or KTY) via encoder

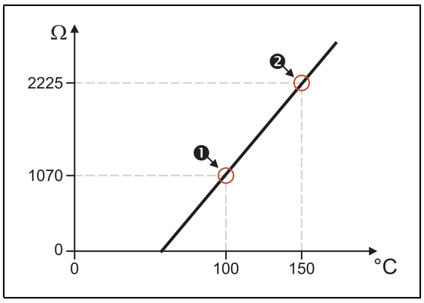
The sensor type has to be selected in C01190/2.

- If the winding temperature detected via the motor temperature sensor reaches the limit value set in C00121/2 (Lenze setting: 120 °):
  - An advance warning in the form of the error response set in C00583/4 is issued.
  - The "oH6: Motor temperature - MultiEncoder > C121" error message is entered into the logbook.
- If the fixed limit value of 150 °C is reached:
  - The error response set in C00583/2 is triggered.
  - The "oH12: Motor overtemperature - MultiEncoder" error message is entered into the logbook.
- If open circuit is detected for the motor temperature sensor:
  - The error response set in C00583/6 is triggered.
  - The "Sd12: Thermal detector error - MultiEncoder" error message is entered into the logbook.

Specific characteristic for the motor temperature sensor

If required, you can define and activate a special characteristic for the motor temperature sensor.

- The specific characteristic is defined on the basis of two grid points which must be set in C01191 and C01192. Those two points define a line which is extrapolated to the right and to the left.
- The specific characteristic can be activated by selecting "1: Spec. characteristic" in C01190/1 for the resolver and in C01190/2 for the encoder.
- In the Lenze setting, the specific characteristic is defined as follows:



- Interpolation point❶
  - C01191/1 = 100 °C
  - C01192/1 = 1070 Ω
- Interpolation point❷
  - C01191/2 = 150 °C
  - C01192/2 = 2225 Ω

[6-4] Lenze setting of the special characteristic

Note!

If a motor is selected from the motor catalogue, parameters C01190, C01191, and C01192 are overwritten!

Up to version 16.xx.xx: Settings for a motor with PT1000

This setting serves to detect the motor temperature via the PT1000 with the same accuracy as in case of the KTY.

- C1190 = "Spec. characteristic"
- C1191/1 = 0 °C
- C1191/2 = 200 °C
- C1192/1 = 1000 ohms
- C1192/2 = 1790 ohms

Up to version 16.xx.xx: Settings for a motor with PT1000 + 2 x PTC

Mit diesen Einstellungen wird die Motortemperatur über PT1000 + 2 x PTC etwas ungenauer erfasst, als über KTY + 2 x PTC. Zudem erfolgt die Übertemperaturabschaltung ca. 5 °C eher als bei KTY + 2 x PTC. Ist eine frühere Übertemperaturabschaltung oder eine größere Ungenauigkeit beim Drehmoment nicht akzeptabel, ist ein Softwareupdate auf Version V17.00.00 oder höher notwendg.

- C1191/1 = 0 °C
- C1191/2 = 200 °C
- C1192/1 = 1143 ohms
- C1192/2 = 1865 ohms

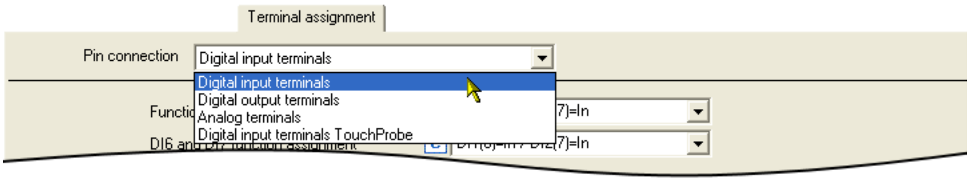
From version 17.00.00: Settings for a motor with PT1000 or PT1000 + 2 x PTC

- C01190 = PT1000 or PT1000 + 2 x PTC

7 I/O terminals

This chapter provides information on the function, possible parameter settings, and technical data of the input/output terminals of the inverter.

In the »Engineer«, the input and output terminals are parameterised on the **Terminal assignment** tab. To do this, go to the **Control terminals** list field and select the terminals that you wish to parameterise:



You can find further information in the respective subchapter:

- Digital input terminals (401)
- Digital output terminals (422)
- Analog terminals (425)
- Touch probe detection (435)

Note!

The input and output terminals of the inverter have already been functionally assigned in the default setting ("Lenze setting"). The preconfigured assignment depends on the technology application selected in C00005 and the control mode selected in C00007:

- **TA "Actuating drive speed":** Terminal assignment of the control modes (465)
- **TA "Table positioning":** Terminal assignment of the control modes (529)
- **TA "Switch-off positioning":** Terminal assignment of the control modes (555)

Wiring diagram, assignment and electrical data of the input and output terminals can be found in the **hardware manual 8400** in the chapter "technical data". The hardware manual is stored in electronic form on the data carrier supplied with the 8400 inverter.

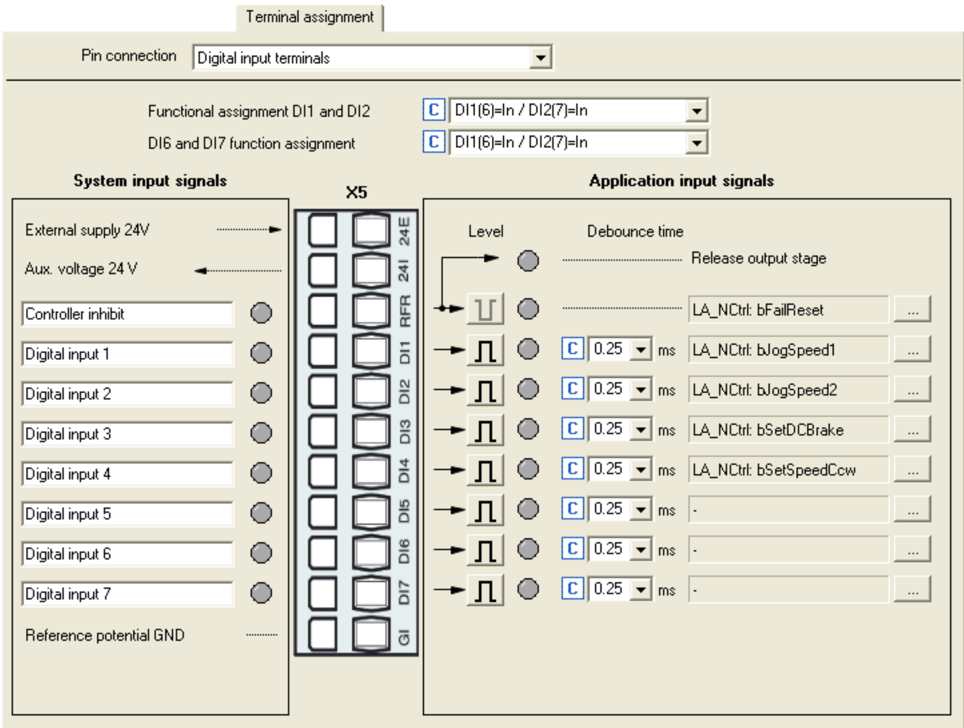
Tip!

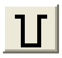
How you can alter the preconfigured assignment of the input and output terminals is described in the chapter entitled "User-defined terminal assignment". (445)

7.1 Digital input terminals

The inverter has 7 parameterisable input terminals (DI1 ... DI7) for detecting digital signals. The RFR control input for enabling the inverter is permanently connected to the device control unit.

Parameterisation dialog in the »Engineer«:



Button	Function
	Indicates the polarity of the input is HIGH active. The polarity can be changed from HIGH active to LOW active by clicking this button.
	Indicates that the polarity of the input is LOW active. The polarity can be changed from LOW active to HIGH active by clicking this button.
...	Open the parameterising dialog for assigning application inputs to the digital input. ▸ Changing the terminal assignment with the »Engineer« (📖 449)

Short overview of parameters for the digital input terminals:

Parameters	Info	Lenze setting Value	Unit
C00115/1	Fct. DI 1/2 200kHz ▸ Change function assignment (📖 403)	0: DI1(6)=In / DI2(7)=In	
C00115/2	Fct. DI 6/7 100kHz ▸ Change function assignment (📖 403)	0: DI1(6)=In / DI2(7)=In	
Digital inputs DI1 ... DI7			
C00114	DigInX: Inversion	Bit coded	
C02830/1...7	DI1...DI7: Debounce time	1: 0.25	
C00443/1	DIx: Terminal level	-	
C00443/2	DIx: Output level	-	
Greyed out = display parameter			

Related topics:

▸ Touch probe detection (📖 435)
▸ User-defined terminal assignment (📖 445)

7.1.1 Change function assignment

The internal processing function of the digital input terminals DI1/DI2 and DI6/DI7 can be reconfigured in C00115 if necessary. In this way, these input terminals can alternatively be used as frequency or counting inputs in order to implement the following functions:

- Detection of the input frequency
- Detection and processing of two unipolar input frequencies to one bipolar frequency
- Counting of input pulses
- Evaluation of the speed feedback (HTL encoder) for the motor control (speed-controlled operation)

C00115/1: Function assignment DI1 and DI2 C00115/2: Function assignment of DI6 and DI7		Function assignment	
		DI1 / DI6	**DI2 / DI7**
0	**DI1(6)=In / DI2(7)=In**	**Digital input**	**Digital input**
1	DI1(6)=FreqIn / DI2(7)=In	Frequency input	Digital input
2	DI1(6)&DI2(7)=FreqIn (2-track)	Frequency input (2-track)	
3	DI1(6)=FreqIn / DI2(7)=direction	Frequency input (speed)	Frequency input (direction)
4	DI1(6)=CountIn / DI2(7)=In	Count input	Digital input

Note!

- In the Lenze setting of C00115, the digital input terminals DI1/DI2 and DI6/DI7 have been configured as "normal" digital inputs.
- The digital input terminals DI3 ... DI5 are generally designed as "standard" digital inputs.
- Very high pulse frequencies can be measured at the DI1/DI2 and DI6/DI7 input terminals if the latter have been configured as frequency or counting inputs in C00115. Scanning is then carried out within less than µs instead of the otherwise usual scanning rate of 1 kHz (1 ms).

You can find detailed information on the respective function assignment in the following subchapters:

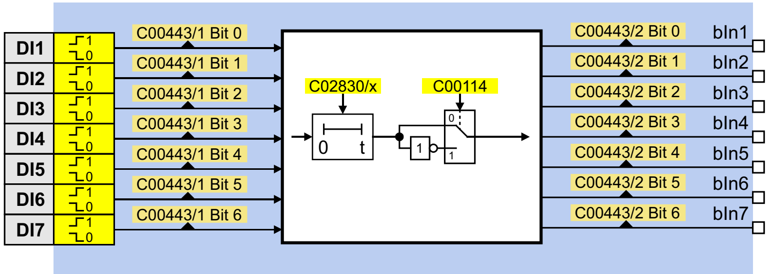
▶ Using DI1(6) and DI2(7) as digital inputs (📖 404)
▶ Using DI1(6) and DI2(7) as frequency inputs (📖 405)
▶ Using DI1(6) as counting input (📖 410)

7.1.1.1 Using DI1(6) and DI2(7) as digital inputs

Function assignment 0: DI1(6)=In / DI2(7)=In

With this setting in C00115 , the digital input terminals have been configured as "normal" digital inputs.

- For each digital input, the debounce time (C02830/1...7) and the terminal polarity (C00114) can be set individually.
- The current terminal level at the input of the internal processing function is shown in C00443/1 in bit-coded form.
- The output level for the application is shown in C00443/2 in bit-coded form.



Internal interfaces to the application

- Relevant outputs at the LS_DigitalInput system block:

Output DIS code | data type	Value/meaning
bIn1 ... bIn7 C00443/2 | BOOL	Digital input DI1 ... DI7

Related topics:

▶ Using DI1(6) and DI2(7) as frequency inputs (📖 405)
▶ Using DI1(6) as counting input (📖 410)
▶ Internal interfaces | System block "LS_DigitalInput" (📖 414)

7.1.1.2 Using DI1(6) and DI2(7) as frequency inputs

General information on using the input terminals as frequency inputs

The frequency inputs serve to detect HTL encoders with any number of increments and single-track and two-track signals. Single-track signals can be evaluated with or without rotation signal.

Note!

- Make sure that, when motor control with speed feedback is in use, the maximum input frequency of the respective input terminal is not exceeded.
  - DI1/DI2: f_{max} = 100 kHz (from version 02.00.00: f_{max} = 200 kHz)
  - DI6/DI7: f_{max} = 100 kHz
- If the encoder signal is used as an actual speed value:
  Number of encoder pulses / revolution ≤ **8192!**

Example of DI6/DI7 (in accordance with the preceding note):

- Encoder increment: 512 pulses / motor revolution
- Reference speed (C00011): 1500 rpm
- Speed setpoint: 100 %

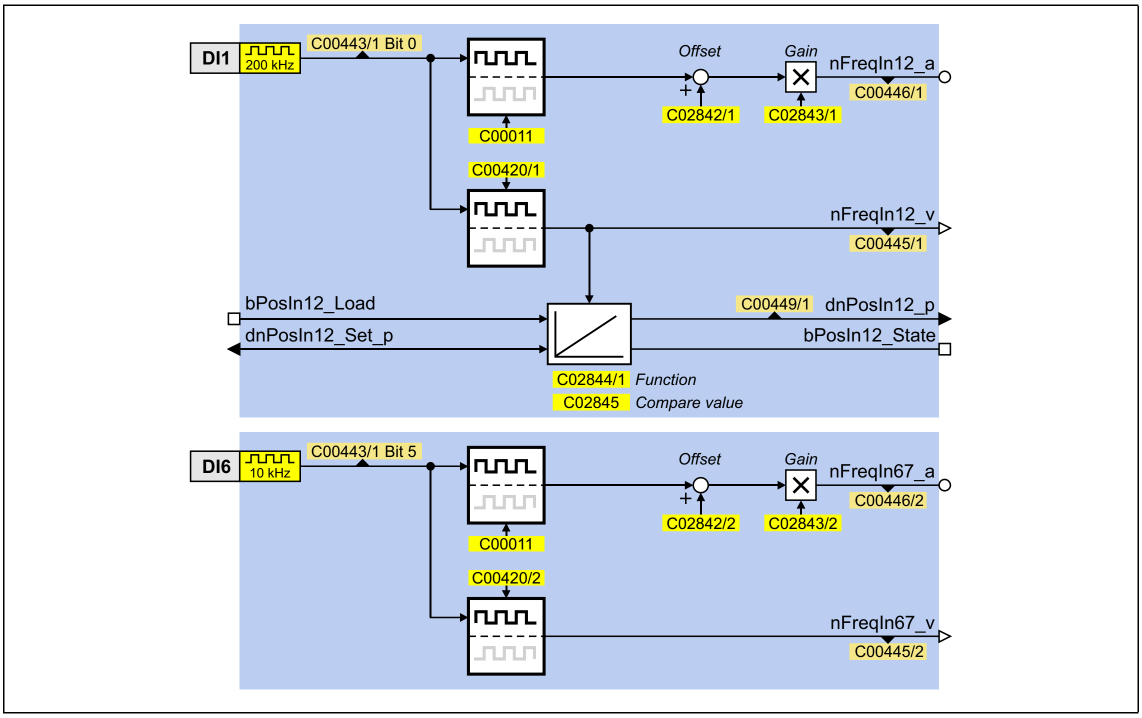
$$\text{Input frequency} = \frac{1500\ \text{rpm}}{60\ \text{s}} \times 512\ \text{pulses} = 12800\ \text{pulses/s} = 12.8\ \text{kHz}$$

Tip!

The LS_DigitalInput system block can also provide the encoder position. Detailed information on this topic is provided in chapter "Output of the encoder position of the DI1/DI2 frequency input". (417)

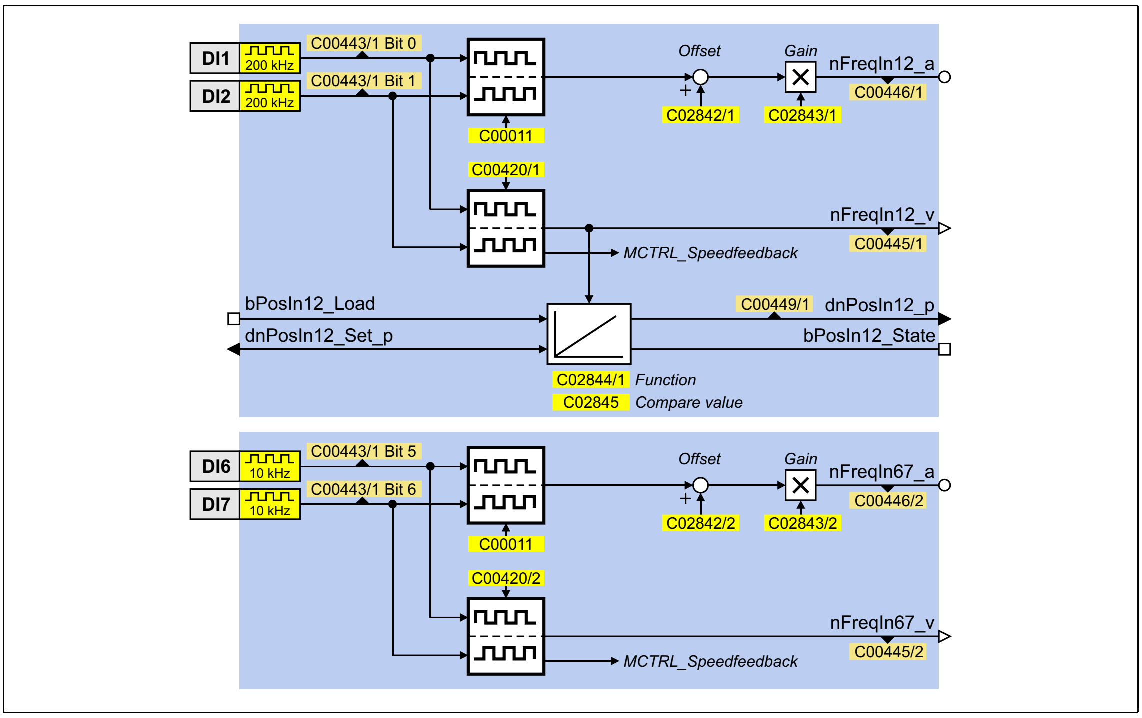
Function assignment 1: DI1(6)=FreqIn / DI2(7)=In

This setting in C00115 configures the input terminal DI1 or DI6 as frequency input. The input terminal DI2 or DI7 remains configured as "normal" digital input..



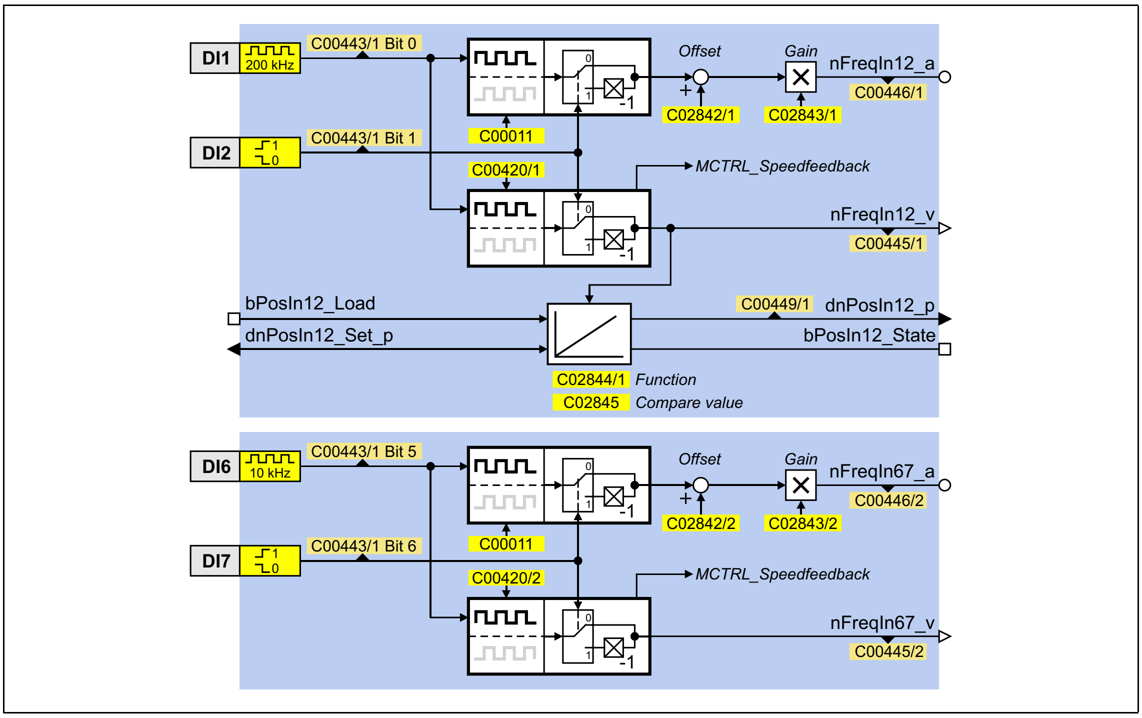
Function assignment 2: DI1(6)&DI2(7)=FreqIn (2-track)

This setting in C00115 can be used to connect a two-track encoder to the DI1/DI2 or DI6/DI7 terminals.



Function assignment 3: DI1(6)=FreqIn / DI2(7)=Direction

This setting in C00115 can be used to connect a single-track encoder to terminals DI1/DI2 or DI6/DI7. For this purpose, the rotation speed is evaluated via terminal DI1(6) and the direction of rotation of the encoder (LOW level ≡ CW direction of rotation) is evaluated via the DI(7) terminal.



Short overview of the parameters for the frequency inputs:

Parameters	Info	Lenze setting	
		Value	Unit
C00011	Appl.: Reference speed	1500	rpm
Frequency input DI1/DI2			
C00115/1	Fct. DI 1/2 200kHz	0: DI1(6)=In / DI2(7)=In	
C00420/1	Encoder increments at FreqIn12	128	Incr./rev.
C02842/1	FreqIn12: Offset	0.00	%
C02843/1	FreqIn12: Gain	100.00	%
C02844/1	PosIn12: Function	Loading with level	
C02845	PosIn12: Comparison value	0	
C00443/1	DIx: Terminal level	-	
C00445/1	FreqIn12_nOut_v	-	Incr/ms
C00446/1	FreqIn12_nOut_a	-	%
C00449/1	FreqIn12_dnOut_p	-	Incr
Frequency input DI6/DI7			
C00115/2	Fct. DI 6/7 100kHz	0: DI1(6)=In / DI2(7)=In	
C00420/2	Encoder increments at FreqIn67	128	Incr./rev.
C02842/2	FreqIn67: Offset	0.00	%
C02843/2	FreqIn67: Gain	100.00	%
C00443/1	DIx: Terminal level	-	
C00445/2	FreqIn67_nOut_v	-	Incr/ms
C00446/2	FreqIn67_nOut_a	-	%
Greyed out = display parameter			

Internal interfaces to the application

- Relevant inputs at the LS_DigitalInput system block:

Input Data type	Information/possible settings	
Frequency input DI1/DI2		
bPosIn12_Load BOOL	Load angle integrator with starting value and reset status signal	
	TRUE	Angle integrator is loaded with the value at *dnPosIn12_Set_p* and *bPosIn12_State* is reset to FALSE.
dnPosIn12_Set_p DINT	Starting value for angle integrator	

- Relevant outputs at the LS_DigitalInput system block:

Output Data type	Value/meaning	
Frequency input DI1/DI2		
nFreqIn12_a C00446/1 | INT	Output frequency as scaled analog signal in [%]	
nFreqIn12_v C00445/1 | INT	Output frequency as speed signal in [inc/ms]	
dnPosIn12_p DINT	Angle output signal • 65536 [incr.] ≡ 1 encoder revolution • Overflow is possible (display via *bPosIn12_State*)	
bPosIn12_State BOOL	Status signal "Overflow occurred/distance processed" • Status signal can be reset via *bPosIn12_Load*.	
	TRUE	Overflow has occurred or distance is processed.
Frequency input DI6/DI7		
nFreqIn67_a C00446/2 | INT	Output frequency as scaled analog signal in [%]	
nFreqIn67_v C00445/2 | INT	Output frequency as speed signal in [inc/ms]	

Related topics:

▸ Output of the encoder position of the DI1/DI2 frequency input (📖 417)
▸ Using DI1(6) and DI2(7) as digital inputs (📖 404)
▸ Using DI1(6) as counting input (📖 410)
▸ Internal interfaces | System block "LS_DigitalInput" (📖 414)

7.1.1.3 Using DI1(6) as counting input

General information on use as a counting input

The counting input is used for counting fast edges. A 32-bit counter counts from a parameterisable starting value up to a parameterisable comparison value and then outputs a corresponding status signal.

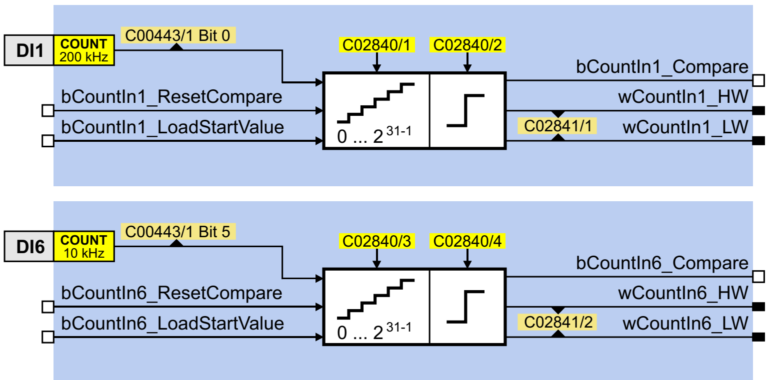
- Possible counting range: 0 ... 2^{31} - 1 (0 ... 2147483647)

Note!

- The starting value must be have been set so that it is smaller than the comparison value. Otherwise, the counter will be kept at the starting value because the condition "Count value ≥ Comparison value" has been satisfied.
- Note the maximum input frequency of the respective input terminal:
  - DI1: f_{max} = 200 kHz
  - DI6: f_{max} = 10 kHz

Function assignment 4: DI1(6)=CountIn / DI2(7)=In

This setting in C00115 configures the input terminal DI1 or DI6 as counting input. The input terminal DI2 or DI7 remains configured as "normal" digital input..



Short overview of parameters for the counting inputs:

Parameters	Info	Lenze setting Value	Unit
Counting input DI1			
C00115/1	Fct. DI 1/2 200kHz	0: DI1(6)=In / DI2(7)=In	
C00621/3	LS_DigitalInput: bCountIn1_Reset	0: Not connected	
C00621/4	LS_DigitalInput: bCountIn1_LoadStartValue	0: Not connected	
C02840/1	CountIn1: Starting value	0	incr
C02840/2	CountIn1: Comparison value	65535	incr
C02841/1	CountIn1: Counter content	-	incr
C00443/1	DIx: Terminal level	-	
Counting input DI6			
C00115/2	Fct. DI 6/7 100kHz	0: DI1(6)=In / DI2(7)=In	
C00621/97	LS_DigitalInput: bCountIn6_Reset	0: Not connected	
C00621/98	LS_DigitalInput: bCountIn6_LoadStartValue	0: Not connected	
C02840/3	CountIn6: Starting value	0	incr
C02840/4	CountIn6: Comparison value	65535	incr
C02841/2	CountIn6: Counter content	-	incr
C00443/1	DIx: Terminal level	-	
Greyed out = display parameter			

Internal interfaces to the application

- Relevant inputs at the LS_DigitalInput system block:

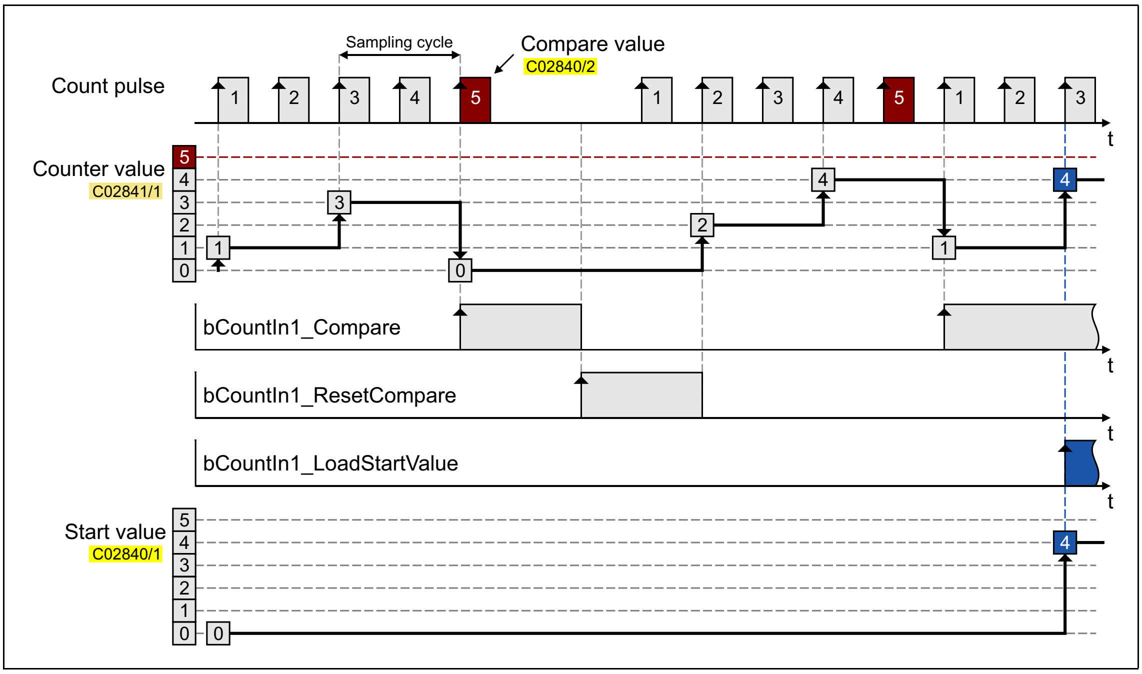
Designator DIS code | data type	Information/possible settings	
Counting input DI1		
bCountIn1_ResetCompare BOOL	Reset status signal "Comparison value reached"	
	FALSE↗TRUE	The output *bCountIn1_Compare* is reset to FALSE.
bCountIn1_LoadStartValue BOOL	Load starting value into counter	
	FALSE↗TRUE	The starting value set in C02840/1 is accepted as the current count value.
Counting input DI6		
bCountIn6_ResetCompare BOOL	Reset status signal "Comparison value reached"	
	FALSE↗TRUE	The output *bCountIn6_Compare* is reset to FALSE.
bCountIn6_LoadStartValue BOOL	Load starting value into counter	
	FALSE↗TRUE	The starting value set in C02840/3 is accepted as the current count value.

- Relevant outputs at the LS_DigitalInput system block:

Designator DIS code | data type	Value/meaning	
Counting input DI1		
bCountIn1_Compare BOOL	Status signal "Comparison value reached"	
	FALSE	Current count value < comparison value (C02840/2)
	TRUE	Current count value ≥ comparison value (C02840/2)
wCountIn1_HW wCountIn1_LW C02841/1 | WORD	Current count value • Output as High and Low word (without sign) • Possible counting range: 0 ... 2^{31} - 1	
Counting input DI6		
bCountIn6_Compare BOOL	Status signal "Comparison value reached"	
	FALSE	Current count value < comparison value (C02840/4)
	TRUE	Current count value ≥ comparison value (C02840/4)
wCountIn6_HW wCountIn6_LW C02841/2 | WORD	Current count value • Output as High and Low word (without sign) • Possible counting range: 0 ... 2^{31} - 1	

Counting behaviour

The following temporal characteristic shows the counting process depending on the signals of the interfaces described before:



[7-1] Transient characteristic of a quick counter block, sampling cycle = 1 ms

- The counter starts with the parameterised starting value.
- If the comparison value is reached or exceeded:
  - The counter jumps back to its starting value.
  - The output *bCount1(6)_Compare* is set to TRUE.
- If there is a FALSE-TRUE edge is at the input *bCountIn1(6)_ResetCompare*, the output *bCountIn1(6)_Compare* can be reset to FALSE.
- If there is a FALSE-TRUE edge at the input *bCountIn1(6)_LoadStartValue*, the current counter content can be reset to the parameterised starting value.

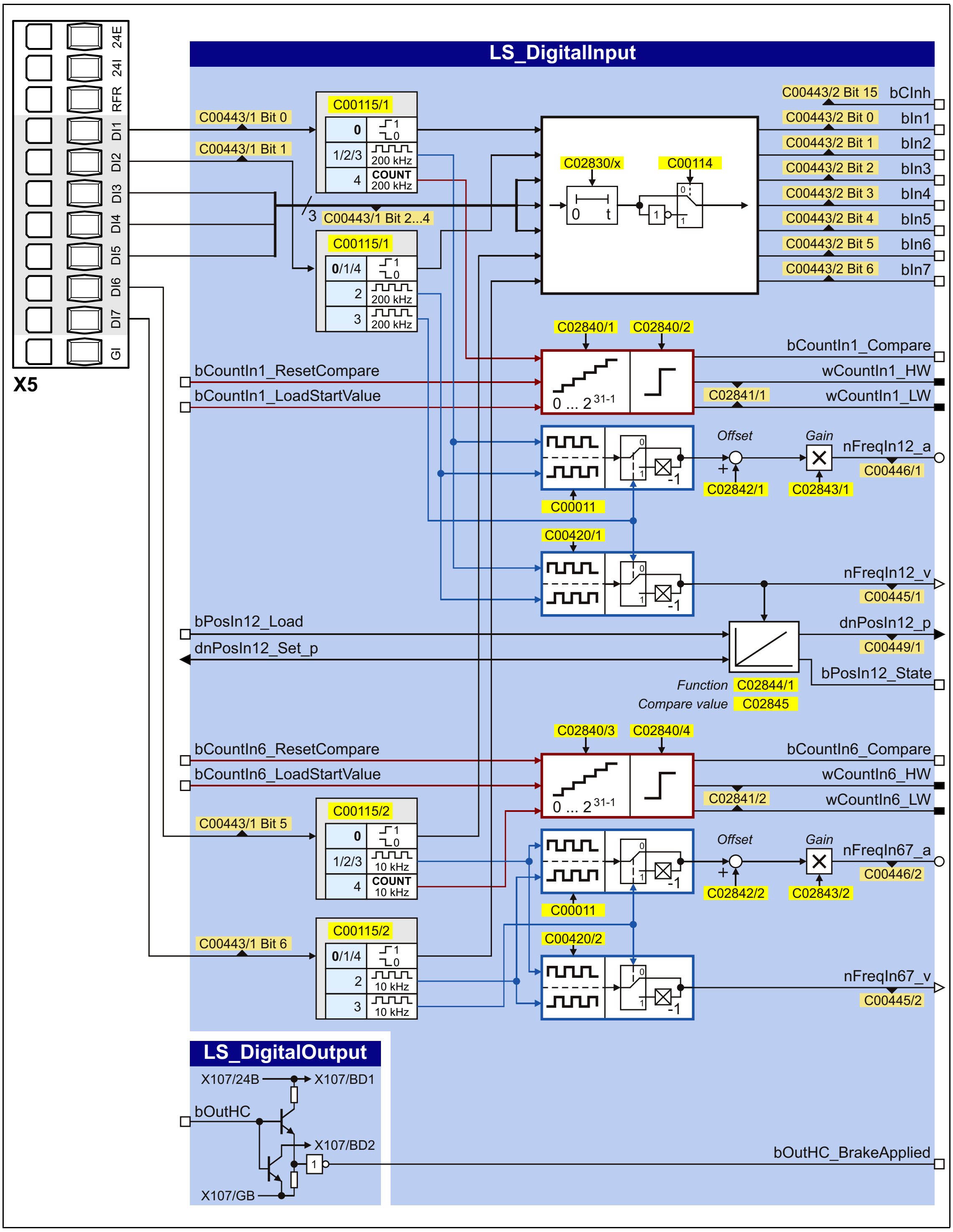
Related topics:

▶ Using DI1(6) and DI2(7) as digital inputs (📖 404)
▶ Using DI1(6) and DI2(7) as frequency inputs (📖 405)
▶ Internal interfaces | System block "LS_DigitalInput" (📖 414)

7.1.2 Internal interfaces | System block "LS_DigitalInput"

The system block **LS_DigitalInput** maps the digital input terminals in the FB editor.

- The internal processing function of the digital DI1/2 and DI6/7 input terminals can be reconfigured in C00115 if necessary. These input terminals can then be alternatively used as frequency inputs or counting inputs.
- The DI3 ... DI5 input terminals are basically designed as "normal" digital inputs.



inputs

Designator DIS code | data type	Information/possible settings	
Counting input DI1	▸ Using DI1(6) as counting input	
bCountIn1_ResetCompare BOOL	Reset status signal "Comparison value reached"	
	FALSE↗TRUE	The output *bCountIn1_Compare* is reset to FALSE.
bCountIn1_LoadStartValue BOOL	Load starting value into counter	
	FALSE↗TRUE	The starting value set in C02840/1 is accepted as the current count value.
Frequency input DI1/DI2	▸ Output of the encoder position of the DI1/DI2 frequency input	
bPosIn12_Load BOOL	Load angle integrator with starting value and reset status signal	
	TRUE	Angle integrator is loaded with the value at *dnPosIn12_Set_p* and *bPosIn12_State* is reset to FALSE.
dnPosIn12_Set_p DINT	Starting value for angle integrator	
Counting input DI6	▸ Using DI1(6) as counting input	
bCountIn6_ResetCompare BOOL	Reset status signal "Comparison value reached"	
	FALSE↗TRUE	The output *bCountIn6_Compare* is reset to FALSE.
bCountIn6_LoadStartValue BOOL	Load starting value into counter	
	FALSE↗TRUE	The starting value set in C02840/3 is accepted as the current count value.

outputs

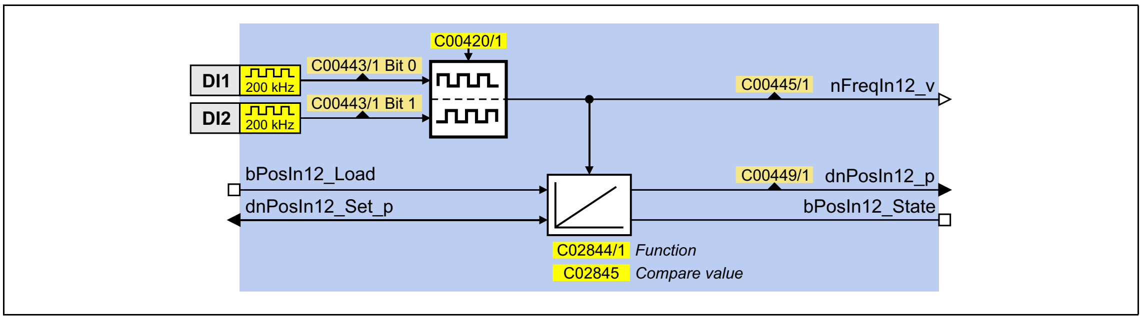
Designator DIS code | data type	Value/meaning	
bCInh C00443/2 | BOOL	RFR digital input (controller enable)	
Digital inputs DI1 ... DI7	▸ Using DI1(6) and DI2(7) as digital inputs	
bIn1 ... bIn7 C00443/2 | BOOL	Digital input DI1 ... DI7	
Counting input DI1	▸ Using DI1(6) as counting input	
bCountIn1_Compare BOOL	Status signal "Comparison value reached"	
	FALSE	Current count value < comparison value (C02840/2)
	TRUE	Current count value ≥ comparison value (C02840/2)
wCountIn1_HW wCountIn1_LW C02841/1 | WORD	Current count value • Output as High and Low word (without sign) • Possible counting range: 0 ... 2^{31} - 1	
Frequency input DI1/DI2	▸ Using DI1(6) and DI2(7) as frequency inputs	
nFreqIn12_a C00446/1 | INT	Output frequency as scaled analog signal in [%]	
nFreqIn12_v C00445/1 | INT	Output frequency as speed signal in [inc/ms]	
dnPosIn12_p DINT	Angle output signal • 65536 [incr.] ≡ 1 encoder revolution • Overflow is possible (display via *bPosIn12_State*)	
bPosIn12_State BOOL	Status signal "Overflow occurred/distance processed" • Status signal can be reset via *bPosIn12_Load*.	
	TRUE	Overflow has occurred or distance is processed.

Designator DIS code | data type	Value/meaning	
Counting input DI6	▶ Using DI1(6) as counting input	
bCountIn1_Compare BOOL	Status signal "Comparison value reached"	
	FALSE	Current count value < comparison value (C02840/4)
	TRUE	Current count value ≥ comparison value (C02840/4)
wCountIn6_HW wCountIn6_LW C02841/2 | WORD	Current count value • Output as High and Low word (without sign) • Possible counting range: 0 ... 2^{31} - 1	
Frequency input DI6/DI7	▶ Using DI1(6) and DI2(7) as frequency inputs	
nFreqIn67_a C00446/2 | INT	Output frequency as scaled analog signal in [%]	
nFreqIn67_v C00445/2 | INT	Output frequency as speed signal in [inc/ms]	
High current output	▶ Switching status of the motor holding brake at the high current output	
bOutHC_BrakeApplied BOOL (from version 15.00.00)	Switching status of a motor holding brake connected to the high current output • The inverted signal logic of this status signal corresponds to the evaluation of a real brake contact which provides the FALSE state in case of a released holding brake and the TRUE state in case of a closed holding brake.	
	FALSE	High current output is controlled if an external 24-V supply is available (holding brake is released).
	TRUE	High current output is not controlled (holding brake is closed) or external 24-V supply is not available.

7.1.2.1 Output of the encoder position of the DI1/DI2 frequency input

The LS_DigitalInput system block includes an integrator for providing the encoder position.

- The integrator can take max. ±32000 encoder revolutions.
- The starting position can be loaded via inputs.
- The internal function can be set via parameters.
- In addition to the encoder position, the "Overflow occurred/distance processed" status signal is provided.



inputs

Designator DIS code | data type	Information/possible settings	
bPosIn12_Load BOOL	Load angle integrator with starting value and reset status signal	
	TRUE	Angle integrator is loaded with the value at *dnPosIn12_Set_p* and *bPosIn12_State* is reset to FALSE.
dnPosIn12_Set_p DINT	Starting value for angle integrator	

outputs

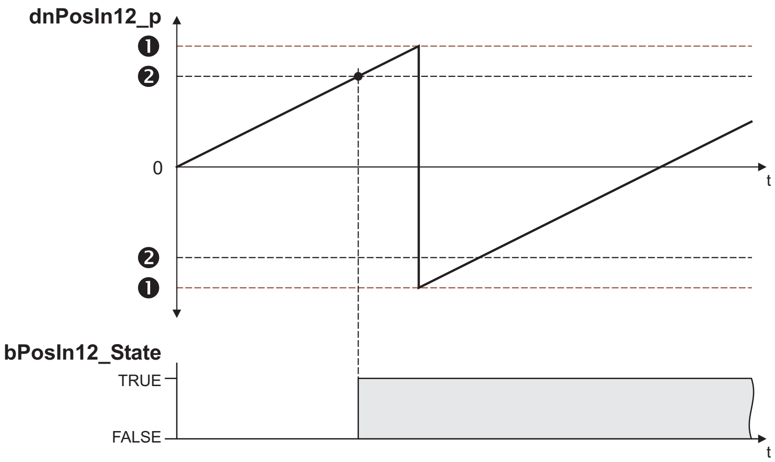
Designator DIS code | data type	Value/meaning	
dnPosIn12_p DINT	Angle output signal • 65536 [incr.] ≡ 1 encoder revolution • Overflow is possible (display via *bPosIn12_State*)	
bPosIn12_State BOOL	Status signal "Overflow occurred/distance processed" • Status signal can be reset via *bPosIn12_Load*.	
	TRUE	Overflow has occurred or distance is processed.

Parameters

Parameters	Possible settings				Information
C02844/1					Function
	0	Loading with level			Load integrator with TRUE level at the *bPosIn12_Load* input (Lenze setting).
	1	Loading with edge			Load integrator with FALSE/TRUE edge at the *bPosIn12_Load* input.
	2	Loading with level + reset			Load integrator when reaching the comparison value or with TRUE level at the *bPosIn12_Load* input.
C02845	0			2000000000	Comparison value • Is valid for both the positive and the negative value range. • Lenze setting: 0

Function at constant input value

Selection: C02844/1 = "0: Loading with level" or "1: Loading with edge"



❶ Overflow at ±2147483648 incr. (±32767 encoder revolutions)
❷ C02845: Comparison value (applies to both positive and negative value range)

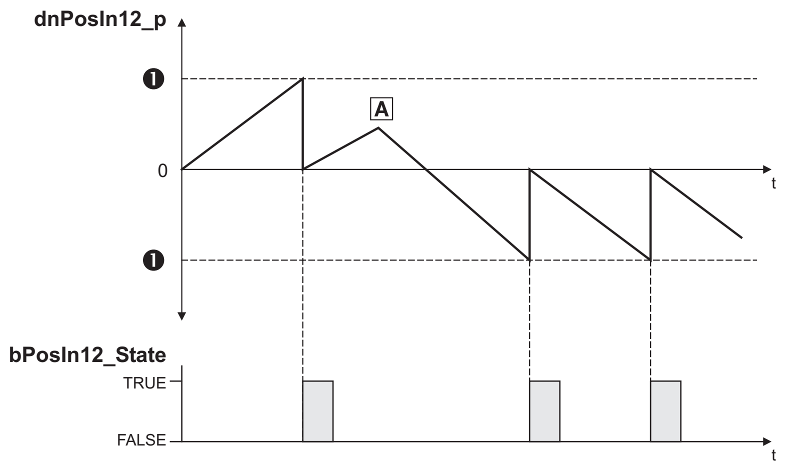
[7-2] Switching performance if the overflow is in the positive direction

- If "0: Loading with level" is selected in C02844/1, the *bPosIn12_Load* input is status-controlled: In case of a TRUE signal, the integrator is loaded with the value at *dnPosIn12_Set_p* and the *bPosIn12_State* output is set to FALSE.
- If "1: Loading with edge" is selected in C02844/1, the *bPosIn12_Load* input is edge-controlled: In case of a FALSE/TRUE edge, the integrator is loaded with the value at *dnPosIn12_Set_p* and then immediately continues to integrate, the *bPosIn12_State* output is set to FALSE.
- A positive *nFreqIn12_v* signal is incremented (the counter content is increased with every cycle).
- A negative *nFreqIn12_v* signal is decremented (the counter content is reduced with every cycle).

- *dnPosIn12_p* provides the counter content of the bipolar integrator.
  - If the counter content exceeds a value of +32767 encoder revolutions (corresponds to +2147483647 incr.), an overflow occurs and the counting process continues at a value of -32768 encoder revolutions.
  - If the counter content falls below a value of -32768 encoder revolutions (corresponds to -2147483648 incr.), an overflow occurs and the counting process starts at a value of +32767 encoder revolutions.
- *bPosIn12_State* is set to TRUE if the comparison value set in C02845 has been reached.

Function at input value with sign reversal

Selection: C02844/1 = "2: Loading with level + reset"



🄰 Sign reversal of the value at *nFreqIn12_v*
❶ C02845: Comparison value (applies to both positive and negative value range)

[7-3] Switching performance if the input signal changes signs

- If "2: Loading with level + reset" is selected in C02844/1, the *bPosIn12_Load* input is status-controlled: In case of a TRUE signal, the integrator is loaded with the value at *dnPosIn12_Set_p* and the *bPosIn12_State* output is set to FALSE.
- A positive *nFreqIn12_v* signal is incremented (the counter content is increased with every cycle).
- A negative *nFreqIn12_v* signal is decremented (the counter content is reduced with every cycle).
- *dnPosIn12_p* provides the counter content of the bipolar integrator.
  - If the positive counter content is higher than the comparison value set in C02845, the comparison value will be subtracted from the counter content, and *bPosIn12_State* will be set to TRUE for one task cycle.
  - If the negative counter content is lower than the comparison value set in C02845, the comparison value will be added to the counter content, and *bPosIn12_State* will be set to TRUE for one task cycle.

Calculation of the output signal

The output value at *dnPosIn12_p* is calculated as per the formula below:

$$\text{dnPosIn12\_p [incr.]} = \text{nFreqIn12\_v [rpm]} \cdot \text{t [s]} \cdot \text{65535 [incr./rev.]}$$

t = integration time
16384 ≡ 15000 rpm
1 ≡ 1 incr.

Example

You want to determine the counter content of the integrator at a certain speed at the input and a certain integration time **t**.

Given values:

- *nFreqIn12_v* = 1000 rpm ≈ integer value 1092
- Integration time t = 10 s
- Starting value of the integrator = 0

Solution:

- Conversion of the *nFreqIn12_v* input signal:

$$\text{1000 rpm} = \frac{\text{1000 rev.}}{\text{60 s}}$$

- Calculation of the output value:

$$\text{dnPosIn12\_p} = \frac{\text{1000 rev.}}{\text{60 s}} \cdot \text{10 s} \cdot \frac{\text{65535 incr.}}{\text{Rev.}} = \text{10922666 incr.}$$

7.1.2.2 Switching status of the motor holding brake at the high current output

This function extension is available from version 15.00.00!

For a simple monitoring of the switching status of a motor holding brake connected to the high current output, the *bOutHC_BrakeApplied* status signal can be used.

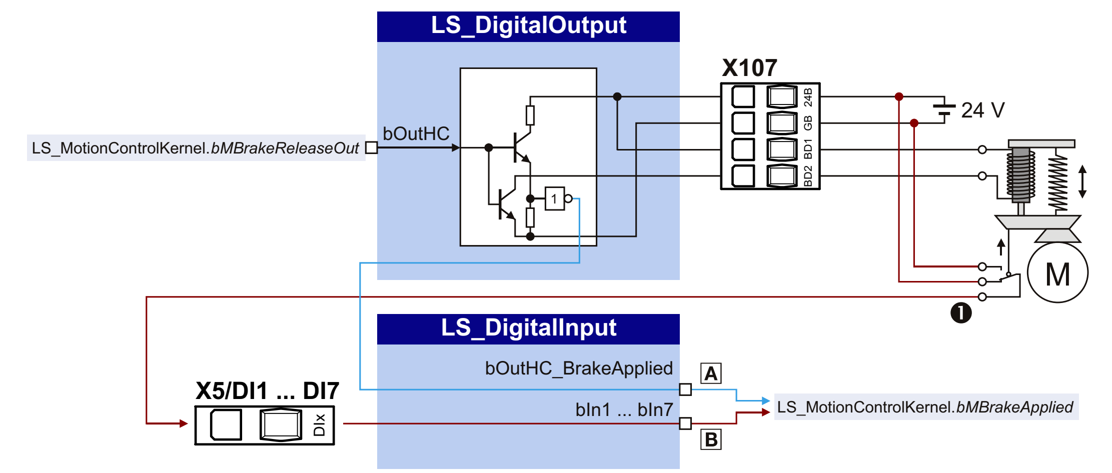
Designator Data type	Value/meaning	
bOutHC_BrakeApplied BOOL	Switching status of a motor holding brake connected to the high current output • The inverted signal logic of this status signal corresponds to the evaluation of a real brake contact which provides the FALSE state in case of a released holding brake and the TRUE state in case of a closed holding brake.	
	FALSE	High current output is controlled if an external 24-V supply is available (holding brake is released).
	TRUE	High current output is not controlled (holding brake is closed) or external 24-V supply is not available. In the closed state, an open-circuit monitoring is available so that a brake status error is created in case of an open circuit.

Required steps for a simple monitoring of the switching state:

1. Link the *bOutHC_BrakeApplied* status signal with the *bMBrakeApplied* input of the SB LS_MotionControlKernel.
2. Activate feedback monitoring of the holding brake control: Set bit 5 in C02582 to "1".

Connection principle

In the following, a connection principle for controlling a motor holding brake via the high current output is described with the two options for feedback monitoring:



Options for feedback monitoring:
A Use of the internal status signal *bOutHC_BrakeApplied*
B Evaluation of the switching contact of the holding via a free digital input
❶ Wire-break detection with a closed holding brake

[7-4] Connection principle motor holding brake

7.2 Digital output terminals

The inverter has

- three parameterisable output terminals (DO1 ... DO3) for outputting digital signals
- a relay output (terminal strip X101),
- a (high-current) output for controlling a brake (terminal strip X107).

Note!

Initialisation behaviour:

- After mains switching up to the start of the application, the digital outputs remain set to FALSE.

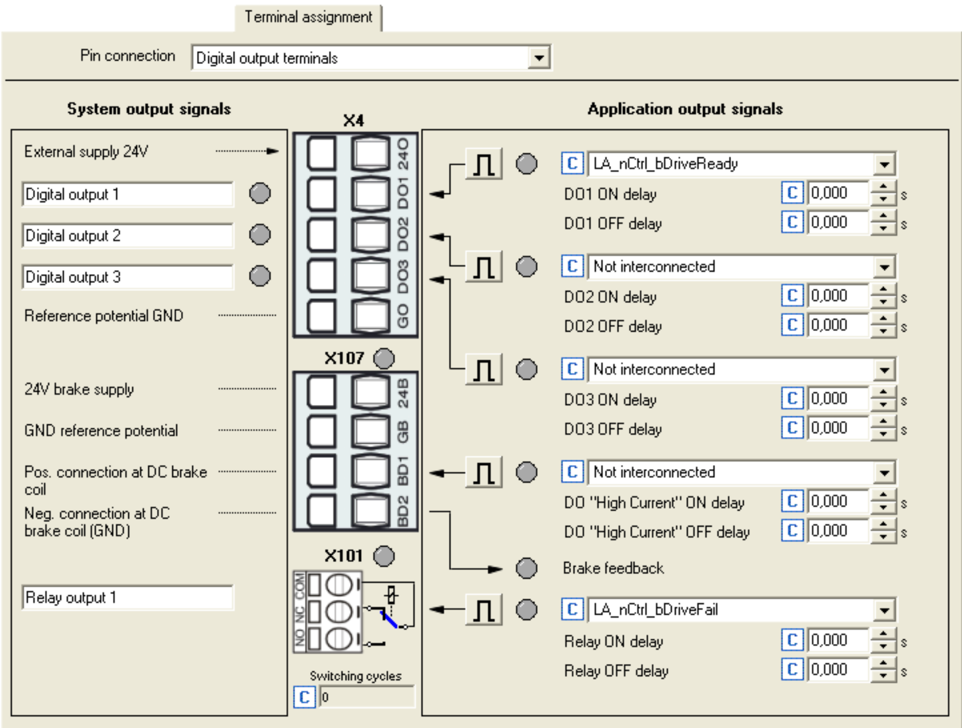
Exception handling:

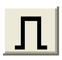
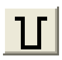
- In case of a critical exception in the application (e.g. reset), the digital outputs are set to FALSE considering the terminal polarity parameterised in C00118.

Switching cycle diagnostics of the relay:

- A reference for evaluating the wear limit can be obtained via the number of switching cycles of the relay displayed in C00177/2.

Parameterisation dialog in the »Engineer«:



Button	Function
	Indicates that the polarity of the output is HIGH active. The polarity can be changed from HIGH active to LOW active by clicking this button.
	Indicates that the polarity of the output is LOW active. The polarity can be changed from LOW active to HIGH active by clicking this button.

Short overview of parameters for the digital output terminals:

Parameters	Info	Lenze setting Value	Unit
Digital outputs DO1 ... DO3			
C00118	DigOutX: Inversion	Bit coded	
C00423/3	DO1 ON delay	0.000	s
C00423/4	DO1 OFF delay	0.000	s
C00423/5	DO2 ON delay	0.000	s
C00423/6	DO2 OFF delay	0.000	s
C00423/7	DO3 ON delay	0.000	s
C00423/8	DO3 OFF delay	0.000	s
C00444/1	DOx: Input level	-	
C00444/2	DOx: Terminal level	-	
High current output			
C00423/9	DO "High Current" ON delay	0.000	s
C00423/10	DO "High Current" OFF delay	0.000	s
C00117	Status of brake output BD	-	
Relay output			
C00423/1	Relay ON delay	0.000	s
C00423/2	Relay OFF delay	0.000	s
Digital outputs - terminal configuration			
C00621/1	LS_DigitalOutput:bRelay	1001: LA_nCtrl_bDriveFail	
C00621/2	LS_DigitalOutput:bOut1	1000: LA_nCtrl_bDriveReady	
C00621/99	LS_DigitalOutput: bOut2	0: Not connected	
C00621/100	LS_DigitalOutput: bOut3	0: Not connected	
C00621/101	LS_DigitalOutput: bOut HighCurrent	0: Not connected	
Greyed out = display parameter			

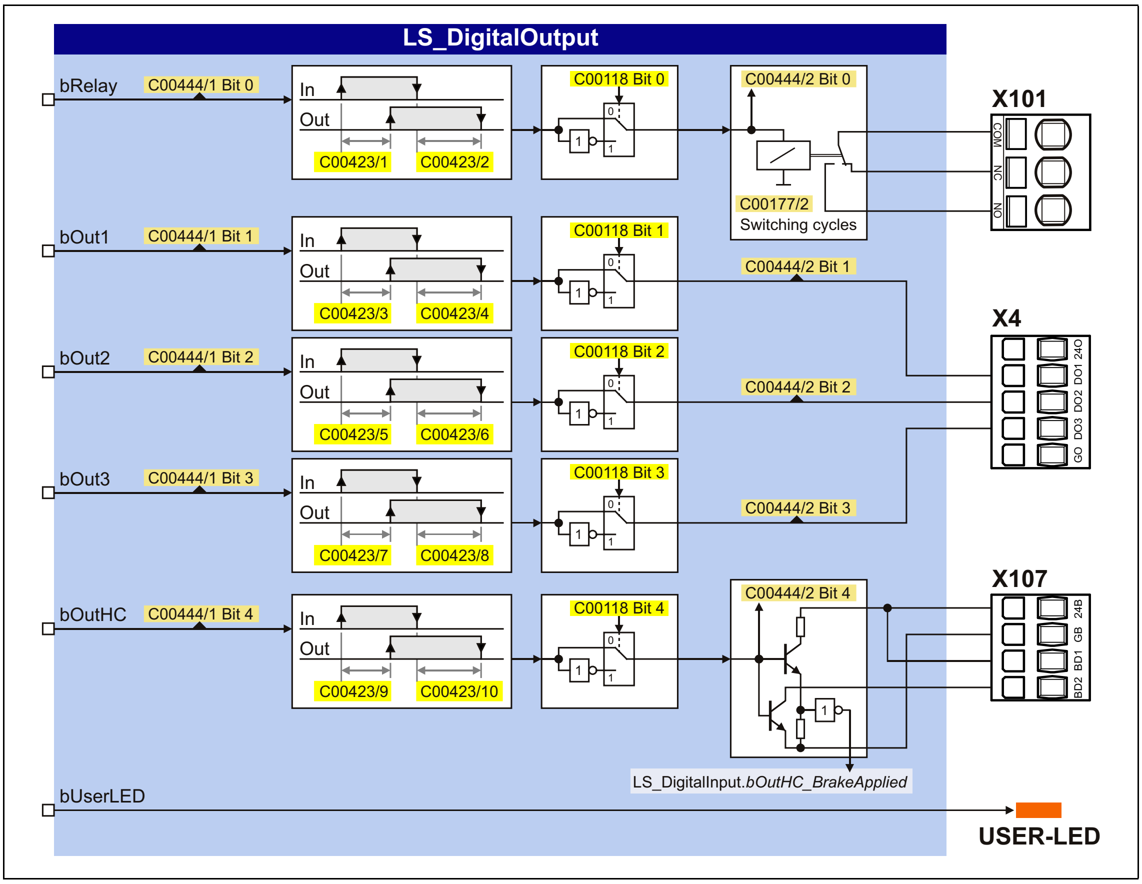
Related topics:

- Configuring exception handling of the output terminals (📖 444)
- User-defined terminal assignment (📖 445)

7.2.1 Internal interfaces | System block "LS_DigitalOutput"

The **LS_DigitalOutput** system block maps the digital output terminals in the FB editor.

- From version 12.00.00 onwards, the orange USER LED at the front of the inverter can be controlled with any digital process signal via the *bUserLED* input.



Input DIS code | data type	Information/possible settings	
bRelay C00444/1 | BOOL	Relay output, potential-free two-way switch	
bOut1 ... bOut3 C00444/1 | BOOL	Digital output DO1 ... DO3	
bOutHC C00444/1 | BOOL	Output for brake control	
bUserLED C00444/1 | BOOL (from version 12.00.00 onwards)	Control of the USER LED at the front of the inverter	
	FALSE	USER LED off
	TRUE	USER LED on

7.3 Analog terminals

The analog input terminals together with the analog output terminals are located on the X3 plug connector.

Analog input terminals

The inverter has four analog input terminals for detecting two current signals and two voltage signals:

- Voltage signals in the ± 10 V range
  The voltage signal can be e.g. an analog speed setpoint or the signal of an external sensor (temperature, pressure, etc.).
- Current signals in the 0/+ 4 ... + 20 mA range
  For open-circuit monitoring, the current signal can be evaluated with regard to "Life Zero" or "Dead Zero":
  - 0 ... 20 mA, without open-circuit monitoring
  - 4 ... 20 mA, with open-circuit monitoring

Note!

To avoid undefined states, free input terminals of the inverter must be assigned as well, e.g. by applying 0 V to the terminal.

Analog output terminals

The inverter has four analog output terminals.

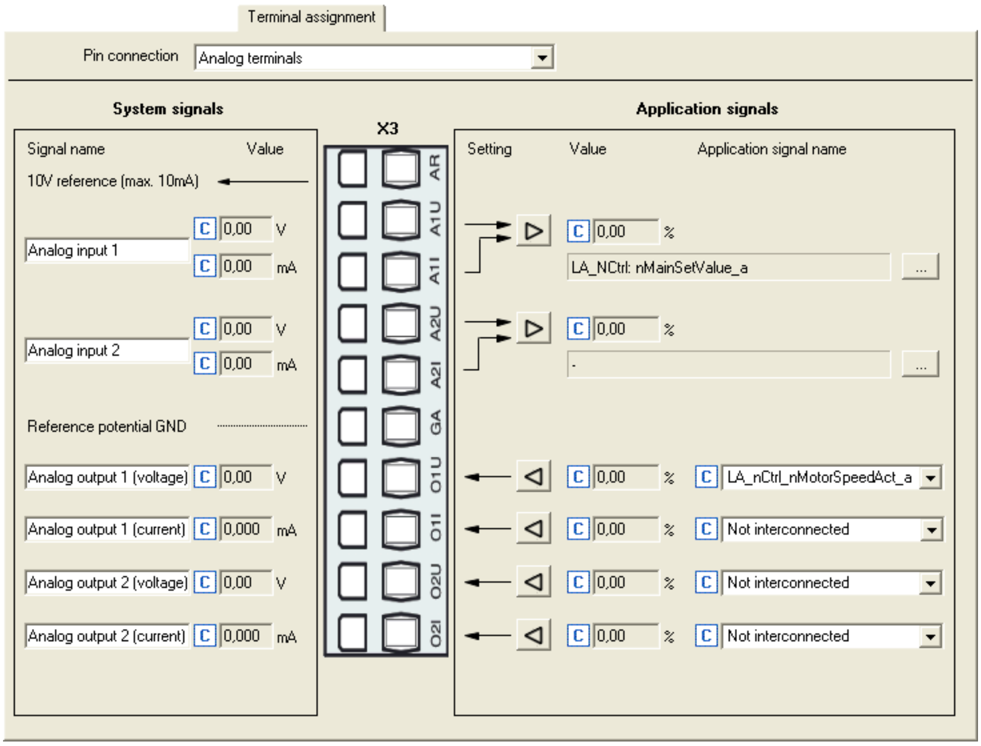
- Two output terminals output an analog current signal (O1I, O2I)
- Two output terminals output an analog voltage signal (O1U, O2U)

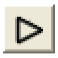
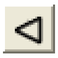
Note!

Initialisation behaviour:
- After mains switching up to the start of the application, the analog outputs remain set to 0 V.

Exception handling:
- In case of a critical exception in the application (e.g. reset), the analog outputs are set to 0 V.

Parameterisation dialog in the »Engineer«:



Button	Function
	Parameterising analog input (📖 428)
	Parameterising analog output (📖 432)
	Open the parameterising dialog for assigning application inputs to the analog input. ▸ Changing the terminal assignment with the »Engineer« (📖 449)

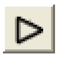
Short overview of parameters for the analog terminals:

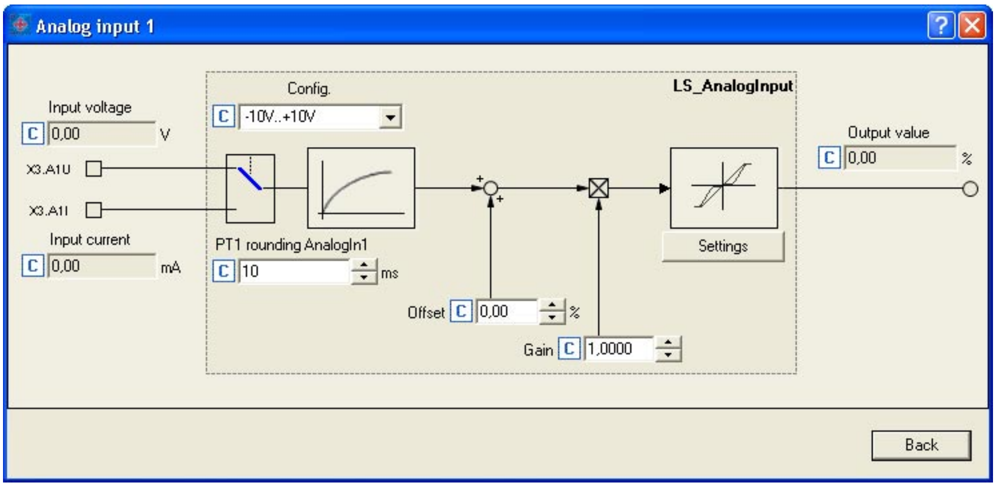
Parameters	Info	Lenze setting Value	Unit
Analog input 1			
C00028/1	AIN1: Input voltage	-	V
C00029/1	AIN1: Input current	-	mA
C00033/1	AIN1: Output value (to application)	-	%
Analog input 2			
C00028/2	AIN2: Input voltage	-	V
C00029/2	AIN2: Input current	-	mA
C00033/2	AIN2: Output value (to application)	-	%
Analog output 1			
C00439/1	O1U: Input value (from application)	-	%
C00439/3	O1I: Input value (from application)	-	%
C00436/1	O1U: Voltage	-	V
C00437/1	O1I: Current	-	mA
Analog output 2			
C00439/2	O2U: Input value (from application)	-	%
C00439/4	O2I: Input value (from application)	-	%
C00436/2	O2U: Voltage	-	V
C00437/2	O2I: Current	-	mA
Analog outputs - terminal assignment			
C00620/1	LS_AnalogOutput: nOut1_a (V)	1003: LA_nCtrl_nMotorSpeedAct_a	
C00620/39	LS_AnalogOutput: nOut1_a (I)	0: Not connected	
C00620/38	LS_AnalogOutput: nOut2_a (V)	0: Not connected	
C00620/40	LS_AnalogOutput: nOut2_a (I)	0: Not connected	
Greyed out = display parameter			

Related topics:

- Configuring exception handling of the output terminals (📖 444)
- User-defined terminal assignment (📖 445)

7.3.1 Parameterising analog input

By clicking on the  button on the **Terminal assignment** tab, you reach the parameterising dialog for the corresponding analog input:



Short overview of parameters for the analog inputs:

Parameters	Info	Lenze setting Value	Unit
Analog input 1			
C00034/1	AIN1: Config.	0: -10V..+10V	
C00026/1	AIN1: Offset	0.00	%
C00027/1	AIN1: Gain	1.0000	
C00028/1	AIN1: Input voltage	-	V
C00029/1	AIN1: Input current	-	mA
C00033/1	AIN1: Output value (to application)	-	%
C00440/1	PT1 rounding AnalogIn1	10	ms
C00598/1	Resp. to open circuit AIN1	3: TroubleQuickStop	
Analog input 2			
C00034/2	AIN2: Config.	0: -10V..+10V	
C00026/2	AIN2: Offset	0.00	%
C00027/2	AIN2: Gain	1.0000	
C00028/2	AIN2: Input voltage	-	V
C00029/2	AIN2: Input current	-	mA
C00033/2	AIN2: Output value (to application)	-	%
C00440/2	PT1 rounding AnalogIn2	10	ms
C00598/2	Resp. to open circuit AIN2	3: TroubleQuickStop	
Greyed out = display parameter			

Using current input A1I/A2I

In the Lenze setting, voltage signals in the range of ± 10 V are evaluated via the A1U and A2U input terminals. If current signals are detected via the A1I or A2I input terminals instead, the selection "1: 0...20mA" or "2: 4...20mA" is to be set in C00034.

Tip!

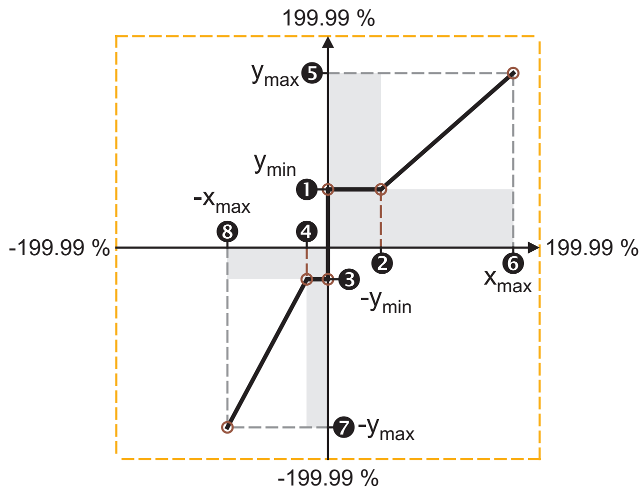
By selecting "2: 4...20mA", you can implement a 4 ...20 mA current loop, e.g. for stipulation of the speed setpoint.

Open-circuit monitoring

In the case of configuration as a 4 ... 20 mA current loop, the fault response set in C00598 takes place in the event of a wire breakage (Lenze setting: "TroubleQuickStop").

7.3.1.1 Signal adaptation by means of characteristic

According to the illustration below, an individual characteristic can be parameterised for the analog inputs via the subcodes of C00010 and C00020 to provide different slopes and a dead band. Here, the input signal corresponds to the X axis and the output signal corresponds to the Y axis:

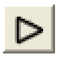


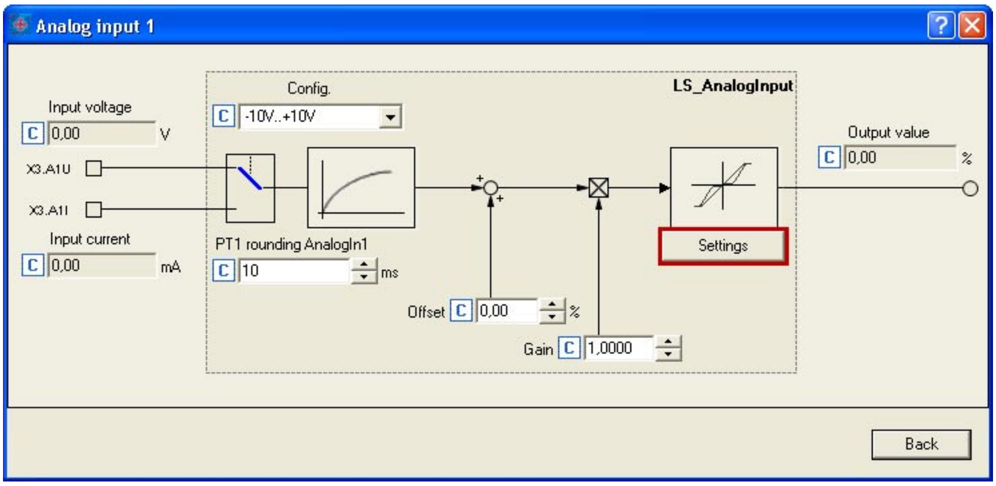
[7-5] Characteristic for analog inputs

Parameters	Info	Lenze setting	
		Value	Unit
C00010/1	❶ AIN1: (+y0) = min	0.00	%
C00010/2	❷ AIN1: (+x0) = Dead band	1.00	%
C00010/3	❸ AIN1: (-y0) = (-min)	0.00	%
C00010/4	❹ AIN1: (-x0) = (-Dead band)	1.00	%
C00010/5	❺ AIN1: (+ymax)	199.99	%
C00010/6	❻ AIN1: (+xmax)	199.99	%
C00010/7	❼ AIN1: (-ymax)	199.99	%
C00010/8	❽ AIN1: (-xmax)	199.99	%
C00020/1	❶ AIN2: (+y0) = min	0.00	%
C00020/2	❷ AIN2: (+x0) = Dead band	1.00	%
C00020/3	❸ AIN2: (-y0) = (-min)	0.00	%
C00020/4	❹ AIN2: (-x0) = (-Dead band)	1.00	%
C00020/5	❺ AIN2: (+ymax)	199.99	%
C00020/6	❻ AIN2: (+xmax)	199.99	%
C00020/7	❼ AIN2: (-ymax)	199.99	%
C00020/8	❽ AIN2: (-xmax)	199.99	%

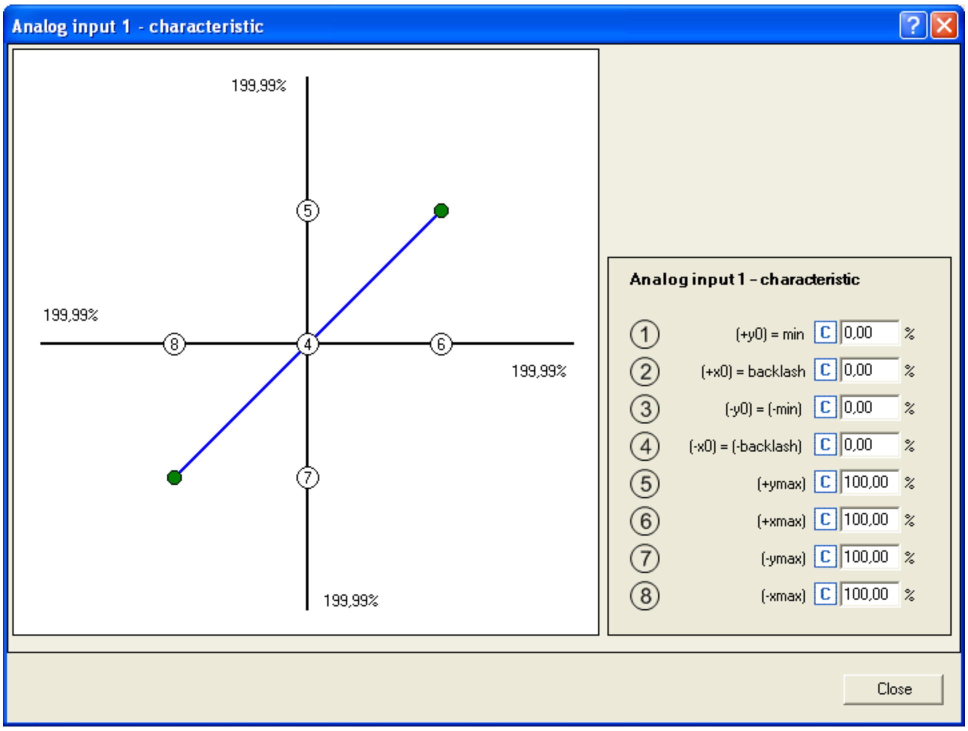
In the »Engineer«, there is a parameterising dialog for entering the characteristic. This dialog also displays the set characteristic graphically.

Proceed as follows to open the dialog for parameterising the characteristic:

1. Go to the **Terminal assignment** tab and select the "Analog terminals" entry in the **Control connections** list field.
2. Click on the  button for the analog input in order to open the *Analog input* dialog.

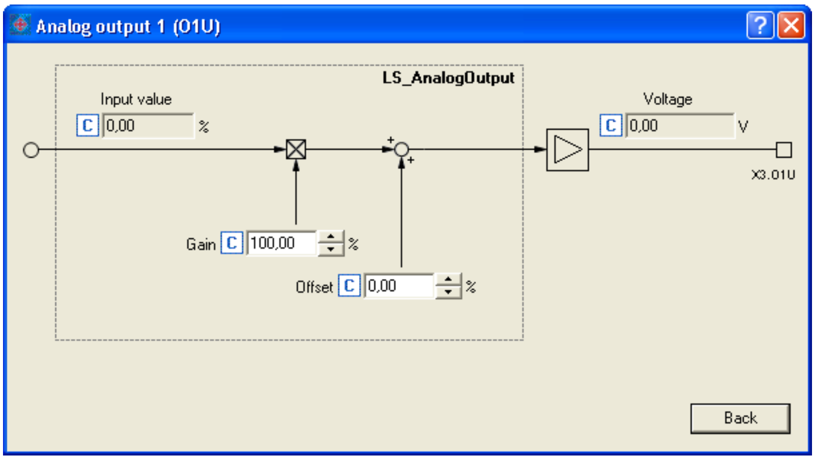


3. Click on the **Settings** button in order to open the *Analog input - Characteristic* dialog box:



7.3.2 Parameterising analog output

By clicking on the button on the **Terminal assignment** tab, you can open the parameterising dialog for the corresponding analog output (here: O1U):

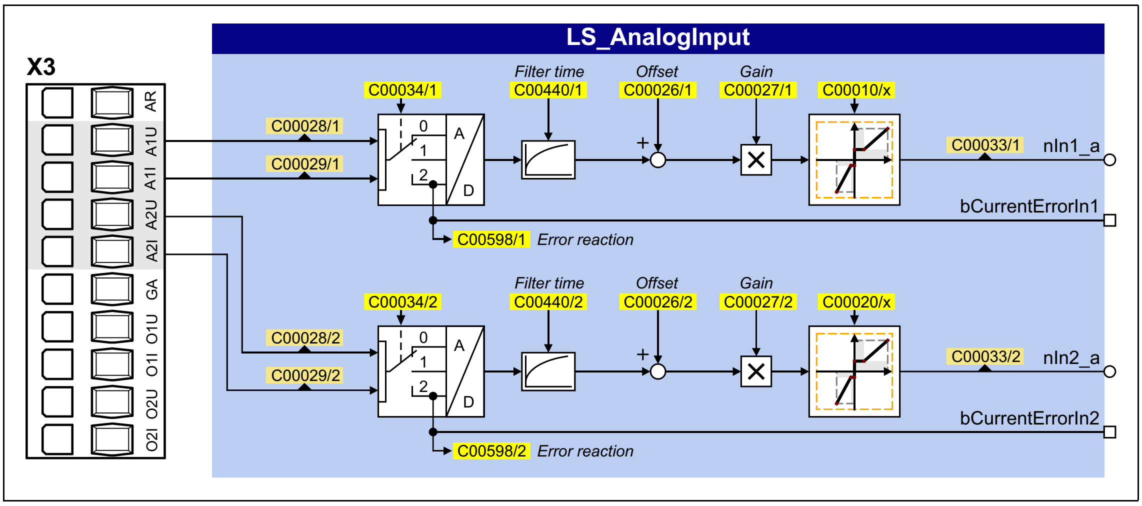


Short overview of parameters for the analog outputs:

Parameters	Info	Lenze setting Value	Unit
Analog output 1			
C00434/1	O1U: Gain	100.00	%
C00435/1	O1U: Offset	0.00	%
C00439/1	O1U: Input value (from application)	-	%
C00439/3	O1I: Input value (from application)	-	%
C00436/1	O1U: Voltage	-	V
C00437/1	O1I: Current	-	mA
Analog output 2			
C00434/2	O2U: Gain	100.00	%
C00435/2	O2U: Offset	0.00	%
C00439/2	O2U: Input value (from application)	-	%
C00439/4	O2I: Input value (from application)	-	%
C00436/2	O2U: Voltage	-	V
C00437/2	O2I: Current	-	mA
Greyed out = display parameter			

7.3.3 Internal interfaces | System block "LS_AnalogInput"

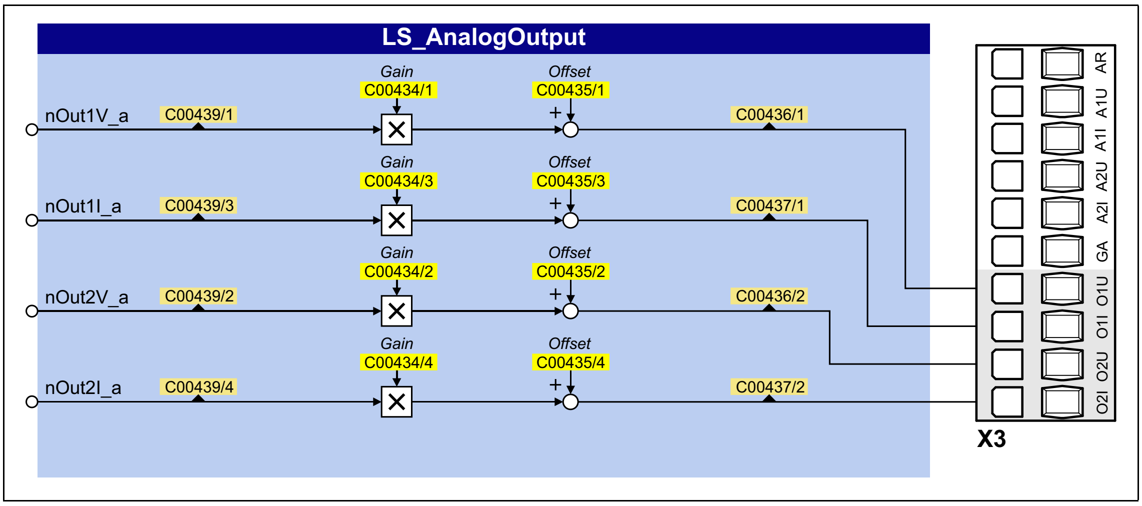
The **LS_AnalogInput** system block maps the analog inputs in the FB editor.



Output DIS code | data type	Value/meaning	
nIn1_a C00033/1 | INT	Analog input 1 • Scaling: $\pm 2^{14}$ ≙ ±10 V for use as voltage input $+2^{14}$ ≙ +20 mA for use as current input	
bCurrentErrorIn1 BOOL	Status signal "Current input error" • Only when analog input 1 is used as current input. • Application: Cable-breakage monitoring of the 4 ...20 mA circuit.	
	TRUE	$\lvert I_{AIN1} \rvert$ < 4 mA
nIn2_a C00033/2 | INT	Analog input 2 • Scaling: $\pm 2^{14}$ ≙ ±10 V for use as voltage input $+2^{14}$ ≙ +20 mA for use as current input	
bCurrentErrorIn2 BOOL	Status signal "Current input error" • Only when analog input 2 is used as current input. • Application: Cable-breakage monitoring of the 4 ...20 mA circuit.	
	TRUE	$\lvert I_{AIN2} \rvert$ < 4 mA

7.3.4 Internal interfaces | System block "LS_AnalogOutput"

The **LS_AnalogInput** system block maps the analog outputs in the FB editor.



Input DIS code | data type	Information/possible settings
nOut1V_a C00439/1 | INT	Analog output 1 (voltage) • Scaling: $2^{14} \equiv 16384 \equiv 10$ V
nOut1I_a C00439/3 | INT	Analog output 1 (current) • Scaling: $2^{14} \equiv 16384 \equiv 20$ mA
nOut2V_a C00439/2 | INT	Analog output 2 (voltage) • Scaling: $2^{14} \equiv 16384 \equiv 10$ V
nOut2I_a C00439/4 | INT	Analog output 2 (current) • Scaling: $2^{14} \equiv 16384 \equiv 20$ mA

7.4 Touch probe detection

"Touch probe" (TP) in general means quick detection of a position by a quick sensor if the pulse duration of the sensor signal is too short to be detected by a "normal" digital input (scanning time: 1 ms).

Moreover, touch probe requires exact detection of the position. Here, the difference in position between the 1 ms position detection and the touch probe signal is considered as correction value.

Applications:

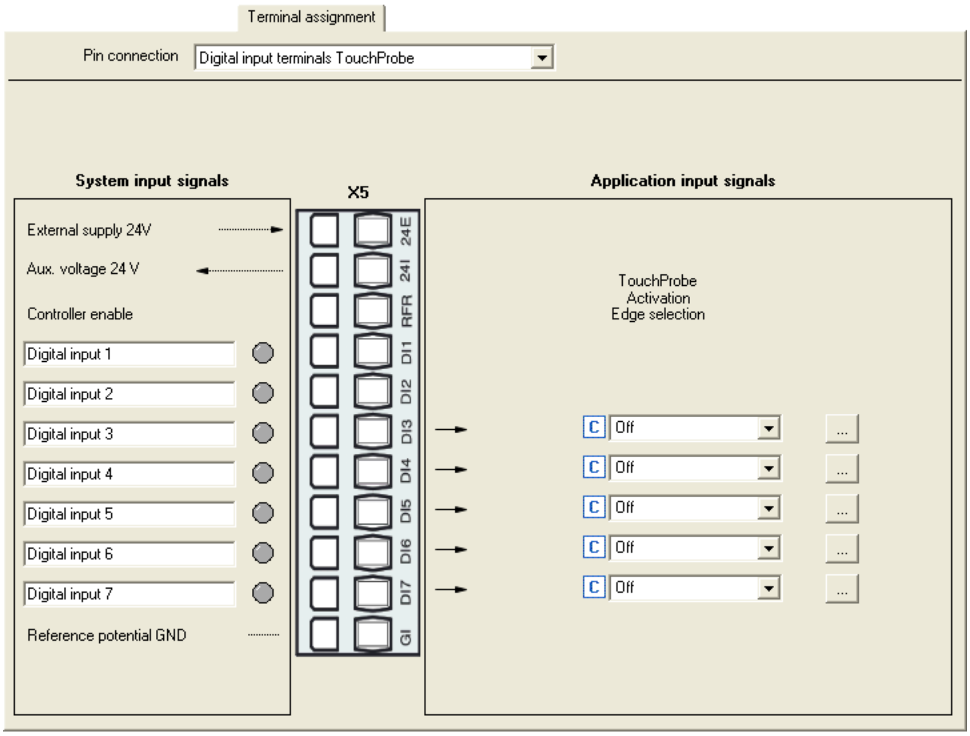
- Precise approach of a position at a previously detected signal mark
- High-precision traversing of a section starting at a previously detected signal mark
- Safe detection of signal marks with very short signal edges
- Measurement of a high-precision position at a previously detected signal mark
- Homing to touch probe (homing signal)
- Relative residual path positioning (traversing of a section starting at signal mark)
- Absolute positioning to target position, activated by a safely detected signal mark
- Position measurement of a distance between 2 signal marks

Overview of touch probe signal sources

The 8400 TopLine provides the following signal sources for touch probe detection which can be configured independently:

Signal source	Edge sensitivity	Applications
Digital input DI3 ... Digital input DI7	Rising edge, falling edge, rising and falling edge (parameterisable)	Homing Residual path positioning Position measurement Freely interconnectable
Z-track encoder (From version 12.00.00)		Homing
Z-track resolver (From version 12.00.00)		Homing

Parameterisation dialog in the »Engineer«:



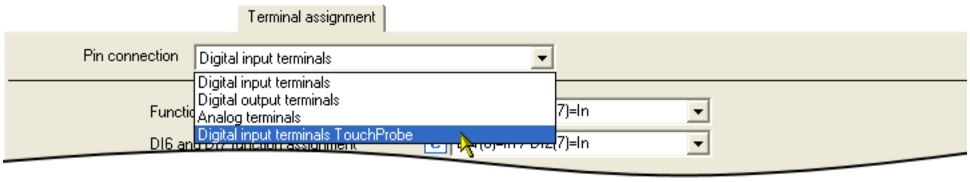
Button	Function
...	Open the parameterisation dialog for the selected TP signal source. ▸ Parameter setting (📖 437)

Parameters	Info	Lenze setting
C02810/3...7	TPDigIn3 ... TPDigIn7: Edge selection	0: Off

7.4.1 Parameter setting

Proceed as follows to open the parameterisation dialog for setting a TP signal source:

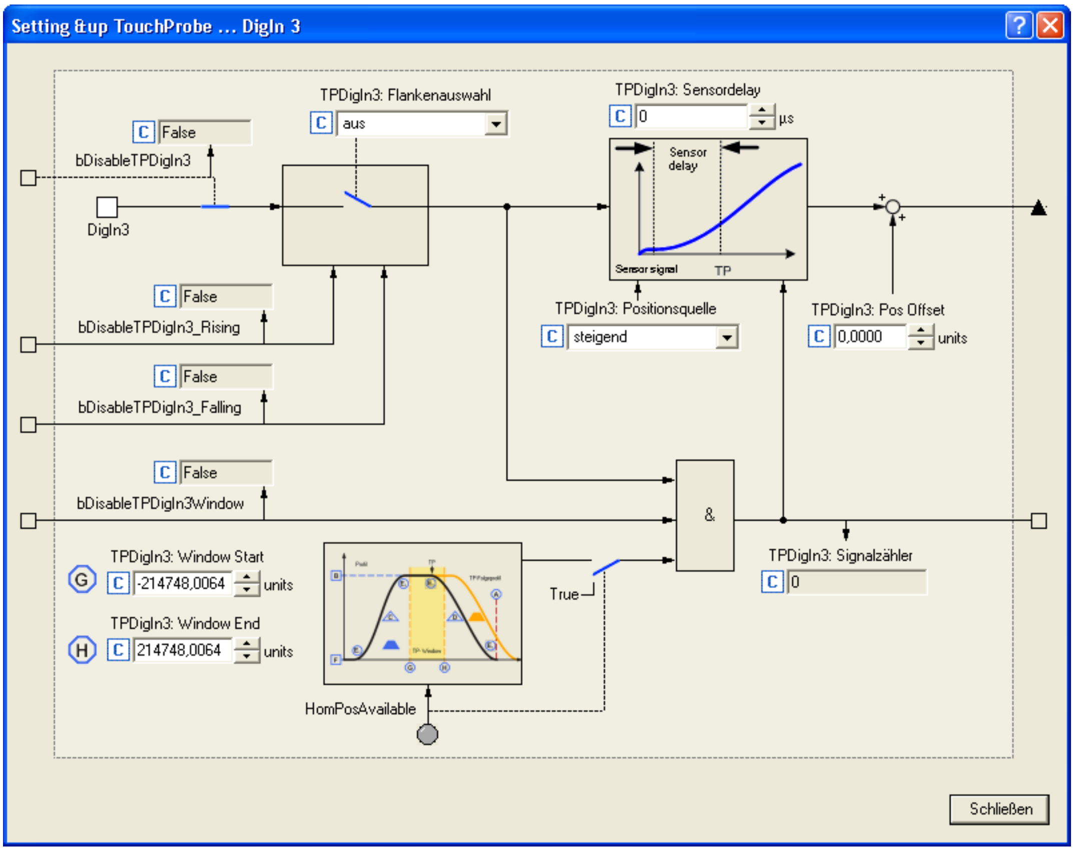
1. Go to the **Terminal Assignment** tab and select "Digital input terminals TouchProbe" in the **Control connections** list field:



2. Click the ... button for the digital input that is to be set for the touch probe detection.

Tip!

If the "Table positioning" technology application is used, the following parameterisation dialog can be opened for the selected TP signal source in the parameterisation dialogs for the Profile entry and the basic "Homing" function via the **Set up touch probe...** button:



Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value	Unit
C02810/3...7	TPDigIn3 ... TPDigIn7: Edge selection	0: Off	
C02811/3...7	TPDigIn3 ... TPDigIn7: Sensor delay	0	µs
C02812/3...7	TPDigIn3 ... TPDigIn7: Pos Offset	0.0000	units
C02813/1...3	TPDigIn3 ... TPDigIn5: Window start	-214748.3647	units
C02814/1...3	TPDigIn3 ... TPDigIn5: Window end	214748.3647	units
C02815/3...7	TPDigIn3 ... TPDigIn7: Position source	0: Position encoder actual value	
C02816/3...7	TPDigIn3 ... TPDigIn7: Signal counter	-	
C02817/3...7	TPDigIn3 ... TPDigIn7: TP position	-	units
Greyed out = display parameter			

Edge selection

Select which edge the corresponding input is to respond to.

- In the case of signal sources DI3 ... DI5, the edge sensitivity can also be dynamically changed via inputs at the LS_TouchProbe SB.

Sensor delay

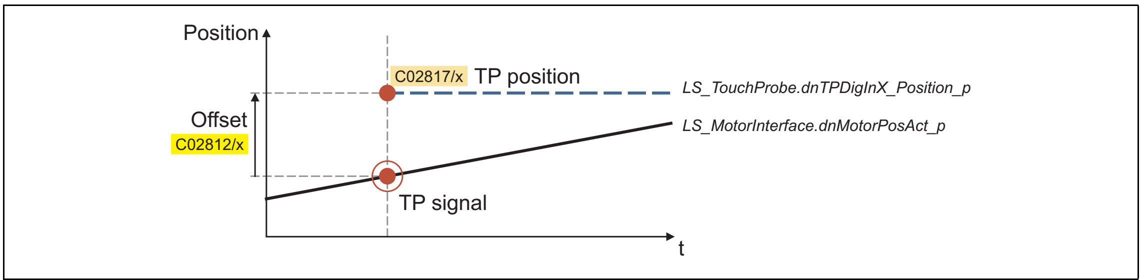
This setting serves to compensate for a sensor delay of the touch probe sensor, if any.

- Typical values for laser photoelectric barriers are e.g. 300 µs.
- Delay of the digital inputs for the 8400 device series:
  - 5 µs for a rising edge
  - 25 µs for a falling edge
- Internal automatic compensation for device-internal signal delay due to encoder zero pulse.

Pos Offset

Use this setting to add an offset to the position value measured by touch probe. This may be required if the touch probe sensor has assumed a disadvantageous position on the machine. By adding an offset the touch probe sensor can be moved to a position which is more convenient to the application.

The offset value given in [units] is added to the TP position value which depends on the position source selected in C02815/x. The offset value influences the display of the TP position (C02817/x), home position values internally derived from the touch probe and the respective output *dnTpDigInX_Position_p* at the SB LS_TouchProbe. For this purpose, the offset value is converted internally from [units] to [increments].



Window start / end

Via the two parameters Window start (C02813/x) and Window end (C02814/x), acceptance windows can be set for the DI3 ... DI5 signal sources in which the touch probe signal is accepted.

- The acceptance windows can be dynamically activated via the inputs at the LS_TouchProbe SB.
- If the actual position is outside the acceptance window, touch probe is automatically deactivated.
- If both window limits are set to "±214748.3647", the acceptance window does not have any effect.
- To properly use this function, the drive needs to know the home position (zero position).

Position source

Selection of the position signal source to be measured with touch probe. This usually is the actual position of the motor / encoder position encoder.

Note!

When the touch probe functionality is used in the operating modes "Positioning" and "Homing":

Make sure that the position source of the respective TP signal in C02815/x is set to "0: position encoder actual value". Otherwise, no TP correction will take place.

7.4.1.1 Z-track encoder as a touch probe signal source

From version 12.00.00 onwards, the Z-track of a connected encoder is also supported as a touch probe signal source.

Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value	Unit
C02810/8	TPEncoderZtrack: Edge selection	0: Off	
C02811/8	TPEncoderZTrack: Sensor delay	0	µs
C02812/8	TPEncoderZTrack: Pos offset	0.0000	units
C02813/4	TPEncoderZTrack: Window start	-214748.3647	units
C02814/4	TPEncoderZTrack: Window end	214748.3647	units
C02815/8	TPEncoderZTrack: Position source	0: Position encoder actual value	
C02816/8	TPEncoderZTrack: Signal counter	-	
C02817/8	TPEncoderZTrack: TP position	-	units
Greyed out = display parameter			

Related topics:

- Multi-Encoder (X8) (📖 341)

7.4.1.2 Z-track resolver as a touch probe signal source

From version 12.00.00 onwards, the Z-track of a connected resolver is also supported as a touch probe signal source.

Short overview of the relevant parameters:

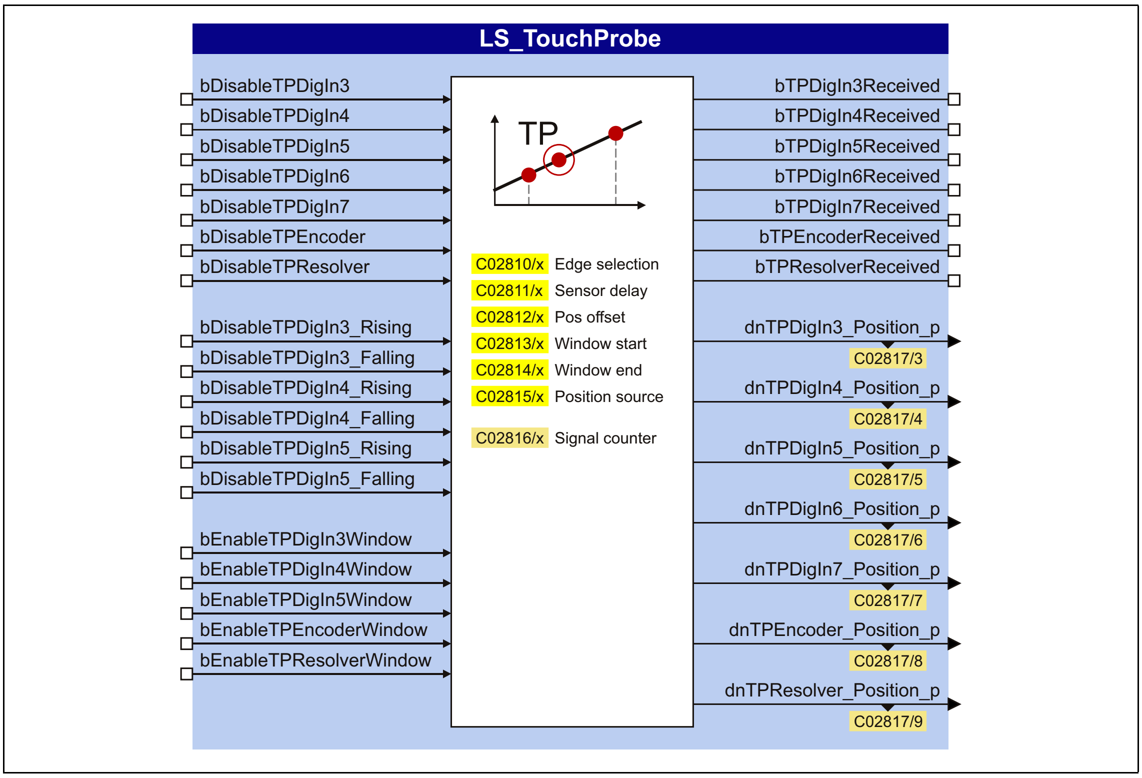
Parameters	Info	Lenze setting Value	Unit
C02810/9	TPResolverZtrack: Edge selection	0: Off	
C02811/9	TPResolverZTrack: Sensor delay	0	µs
C02812/9	TPResolverZTrack: Pos offset	0.0000	units
C02813/5	TPResolverZTrack: Window start	-214748.3647	units
C02814/5	TPResolverZTrack: Window end	214748.3647	units
C02815/9	TPResolverZTrack: Position source	0: Position encoder actual value	
C02816/9	TPResolverZTrack: Signal counter	-	
C02817/9	TPResolverZTrack: TP position	-	units
Greyed out = display parameter			

Related topics:

- Resolver (X7) (📖 335)

7.4.2 Internal interfaces | System block "LS_TouchProbe"

The **LS_TouchProbe** system block provides the internal interfaces for touch probe detection in the function block editor:



inputs

Input Data type	Information/possible settings	
bDisableTPDigIn3...7 BOOL	DI3 ... DI7: Dynamically deactivate TP function	
	TRUE	TP function is deactivated.
bDisableTPEncoder BOOL (From version 12.00.00)	Z-track of encoder: Dynamically deactivate TP function	
	TRUE	TP function is deactivated.
bDisableTPResolver BOOL (From version 12.00.00)	Z-track of resolver: Dynamically deactivate TP function	
	TRUE	TP function is deactivated.
bDisableTPDigIn3...5_Rising BOOL	DI3 ... DI5: Dynamically deactivate detection of rising edges	
	TRUE	Detection of rising edges is deactivated.
bDisableTPDigIn3...5_Falling BOOL	DI3 ... DI5: Dynamically deactivate detection of falling edges	
	TRUE	Detection of falling edges is deactivated.
bEnableTPDigIn3...5Window BOOL	DI3 ... DI5: Activate acceptance window	
	TRUE	Acceptance window function is active: • If the actual position is outside the acceptance window, whose starting position is set in C02813/1...3 and whose end position is set in C02814/1...3, touch probe is automatically deactivated.
bEnableTPEncoderWindow BOOL (From version 12.00.00)	Z-track of encoder: Activate acceptance window	
	TRUE	Acceptance window function is active: • If the actual position is outside the acceptance window, whose starting position is set in C02813/4 and whose end position is set in C02814/4, touch probe is automatically deactivated.

Input Data type	Information/possible settings	
bEnableTPResolverWindow BOOL (From version 12.00.00)	Z-track of resolver: Activate acceptance window	
	TRUE	Acceptance window function is active: • If the actual position is outside the acceptance window, whose starting position is set in C02813/5 and whose end position is set in C02814/5, touch probe is automatically deactivated.

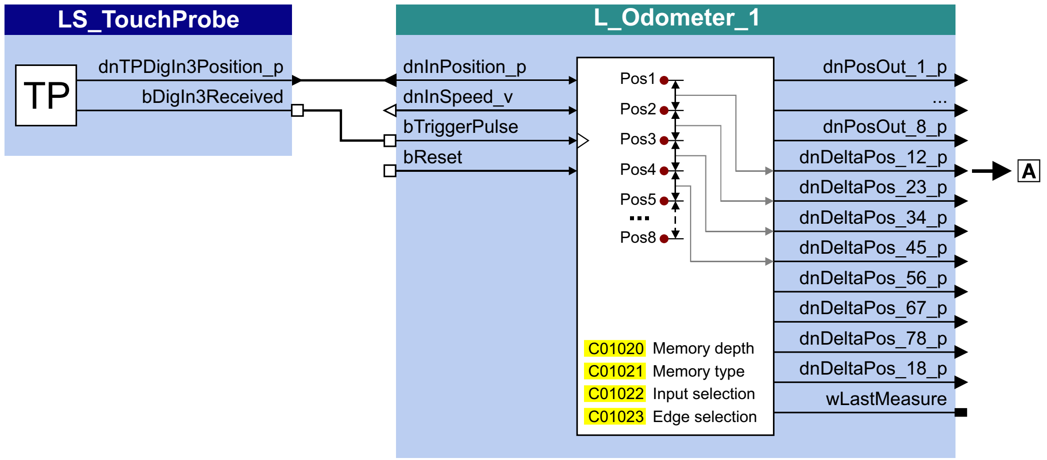
outputs

Output Data type	Value/meaning	
bTPDigIn3...7Received BOOL	TRUE	DI3 ... DI7: Touch probe received. • The signal is only pending for one task cycle (1 ms).
bTPEncoderReceived BOOL (From version 12.00.00)	TRUE	Z-track of encoder: Touch probe received. • The signal is only pending for one task cycle (1 ms).
bTPResolverReceived BOOL (From version 12.00.00)	TRUE	Z-track of resolver: Touch probe received. • The signal is only pending for one task cycle (1 ms).
dnTPDigIn3...7_Position_p DINT	DI3 ... DI7: Position measured by touch probe in [increments]	
dnTPEncoder_Position_p DINT (From version 12.00.00)	Z track of encoder: Position measured by touch probe in [increments]	
dnTPResolver_Position_p DINT (From version 12.00.00)	Z track of resolver: Position measured by touch probe in [increments]	

7.4.2.1 Application example: "Position measurement"

The touch probe function can be combined with the L_Odometer FB for a position measurement. This FB is able to save position signals to a ring buffer and detect differences between two position signals.

In the wiring below, digital input DI3 is used to connect the touch probe sensor. For the sake of clarity, irrelevant inputs and outputs of the LS_TouchProbe SB are masked out.



A Delta position between two touch probes

[7-6] Wiring for position measurement

7.5 Configuring exception handling of the output terminals

Exception handling for the analog and digital output terminals in the event of an error can be set via decoupling configuration and decoupling values.

- Bit coded selection is carried out in C00441 for the analog output terminals, defining the events that will trigger decoupling.
- Bit coded selection is carried out in C00447 for the digital output terminals, defining the events that will trigger decoupling.

Bit	Event
Bit 0 ☐	SafeTorqueOff
Bit 1 ☐	ReadyToSwitchOn
Bit 2 ☐	SwitchedOn
Bit 3 ☐	Reserved
Bit 4 ☐	Trouble
Bit 5 ☐	Fault
Bit 6 ☐	Reserved
Bit 7 ☐	Reserved
Bit 8 ☐	Reserved
Bit 9 ☐	Fail CAN_Management
Bit 10 ☐	Reserved
Bit 11 ☐	Reserved
Bit 12 ☐	Reserved
Bit 13 ☐	Reserved
Bit 14 ☐	Reserved
Bit 15 ☐	Reserved

Finally, the following parameters define the value/status that the output terminals are to have when they are decoupled:

Parameters	Info	Lenze setting	
		Value	Unit
C00442/1	AOut1_U: Decoupling value	0.00	%
C00442/2	AOut2_U: Decoupling value	0.00	%
C00442/3	AOut1_I: Decoupling value	0.00	%
C00442/4	AOut2_I: Decoupling value	0.00	%
C00448	DigOut decoupling value	Bit coded	

Related topics:

▸ Configuring exception handling of the CAN PDOs (📖 838)

7.6 User-defined terminal assignment

In order to individually adapt the preconfigured assignment of the input/output terminals to your application, you can choose one of the following procedures:

A. In the »Engineer«:
   - Change the terminal assignment on the **Terminal assignment** tab.
   - Change the signal assignment on the **Application Parameters** tab, on the dialog level *Overview* ➞ *Signal flow*.
   - Change the interconnections in the FB editor (on the I/O level).

B. In the »Engineer« or with the keypad:
   - Change the parameters for signal configuration in the parameters list.

Note!

If you change the preconfigured assignment of the input/output terminals, the terminal assignment will be a user-defined one. In C00007, control mode "0: Interconnection changed" will be shown.

Tip!

First of all, select a Lenze configuration useful for the purpose at hand by going to C00005 and selecting a technology application that matches your drive task and then going to C00007 and selecting an appropriate control mode. You will then have an application for which there is a signal flow, logical block links and terminal assignment.

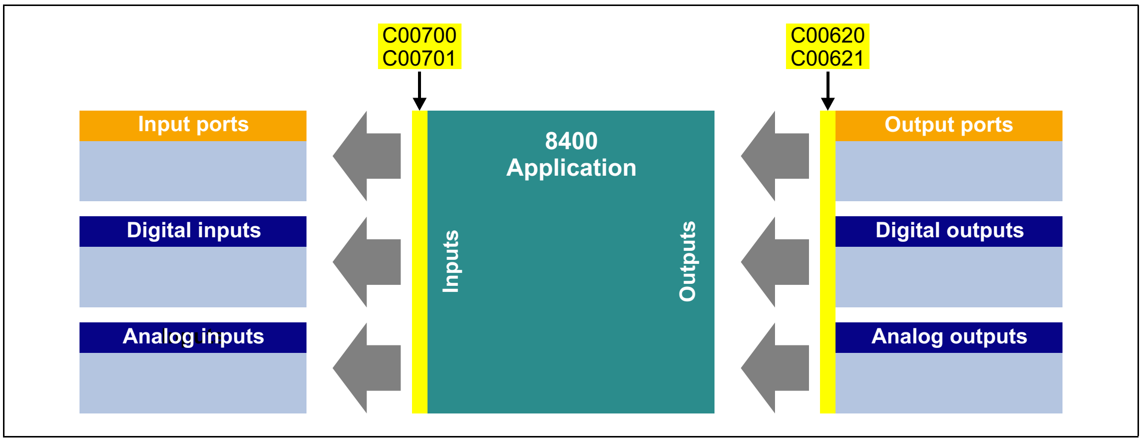
We recommend using the »Engineer« for the implementation of comprehensive user-defined drive solutions.

7.6.1 Source-destination principle

The I/O configuration of the input and output signals is carried out according to the source/destination principle:

- A connection always has a direction and therefore always has a source and a target.
- The inputs signals of the technology application are logically linked to the outputs of system blocks which represent the device input terminals.
- The inputs of system blocks that represent the device output terminals are logically linked to output signals of the technology application.

The following graphic illustrates the source/destination principle:



[7-7] Source-destination principle

Note the following:

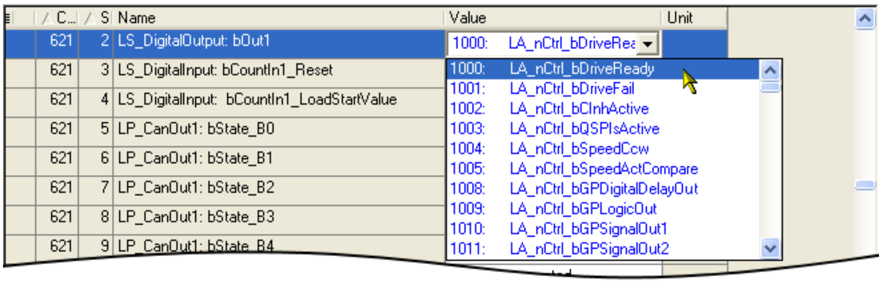
- An equipment input terminal can be logically linked to several inputs of the application block.
- Every input of the application block can only be logically linked to one input signal.
- An output of the application block can be logically linked to several device output terminals.

7.6.2 Changing the terminal assignment with the keypad

You can reconfigure the preconfigured terminal assignment with the keypad (and with the »Engineer«) by means of so-called configuration parameters.

- Each configuration parameter represents a signal input of a system block or application block.
- Each configuration parameter contains a selection list with output signals of the same type of data.
- Logical linking is thus carried out by selecting the output signal for the corresponding signal input.

In the following example, digital output 1 (**LS_DigitalOutput**.*bOut1* input) is logically linked to the status signal "Drive ready" (*LA_nCtrl_bDriveReady* output signal):



Configuration parameters for the analog and digital output terminals

The preconfigured assignment of the analog and digital output terminals can be altered by means of the subcodes of C00620 and C00621:

Parameters	Info	Lenze setting
Analog outputs - terminal assignment		
C00620/1	LS_AnalogOutput: nOut1_a (V)	1003: LA_nCtrl_nMotorSpeedAct_a
C00620/39	LS_AnalogOutput: nOut1_a (I)	0: Not connected
C00620/38	LS_AnalogOutput: nOut2_a (V)	0: Not connected
C00620/40	LS_AnalogOutput: nOut2_a (I)	0: Not connected
Digital outputs - terminal assignment		
C00621/1	LS_DigitalOutput:bRelay	1001: LA_nCtrl_bDriveFail
C00621/2	LS_DigitalOutput:bOut1	1000: LA_nCtrl_bDriveReady
C00621/99	LS_DigitalOutput: bOut2	0: Not connected
C00621/100	LS_DigitalOutput: bOut3	0: Not connected
C00621/101	LS_DigitalOutput: bOut HighCurrent	0: Not connected

Other subcodes (not shown here) allow the configuration of input signals of different system blocks and port blocks.

Configuration parameters for the inputs of the technology application

The following parameters can be used to change the preconfigured assignment of the application inputs:

Parameters	Information
TA "Actuating drive speed": Configuration parameters (📖 478)	
C00700/x	Analog connection list
C00701/x	Digital connection list
TA "Table positioning": Configuration parameters (📖 541)	
C00710/x	Analog connection list
C00711/x	Digital connection list
TA "Switch-off positioning": Configuration parameters (📖 568)	
C00760/x	Analog connection list
C00761/x	Digital connection list

Example

Task: Starting from the preset technology application "Actuating drive speed" and the "Terminals 0" control mode, the DI2 digital input is to be used for choosing an alternative acceleration/deceleration time for the main setpoint instead of for choosing the fixed setpoint 2/3. To do this, the DI2 digital input is not to be linked to the *bJogSpeed2* input but to the *bJogRamp1* input of the application module.

Procedure:

1. Use the keypad to go to the menu level **Applications → Actuating drive speed (conf.)**. This menu level contains all the configuration parameters of the "Actuating drive speed" technology application". ▸ Configuration parameters (📖 478)
2. Navigate to the configuration parameter LA_NCtrl: bJogSpeed2 (C00701/10) which represents the logical signal link of the application input *bJogSpeed2*.
3. Change the setting of C00701/10:
   Change selection "16001: DigIn_bIn2" in selection "0: Not interconnected".
4. Navigate to the configuration parameter LA_NCtrl: bJogRamp1 (C00701/13) which represents the logical signal link of the application input *bJogRamp1*.
5. Change the setting of C00701/13:
   Change selection "0: Not interconnected" in selection "16001: DigIn_bIn2".

Tip!

The example shows that, for each input of the application block, the associated configuration parameter (C00700/x or C00701/x) is only allowed to contain one source that you enter.

7.6.3 Changing the terminal assignment with the »Engineer«

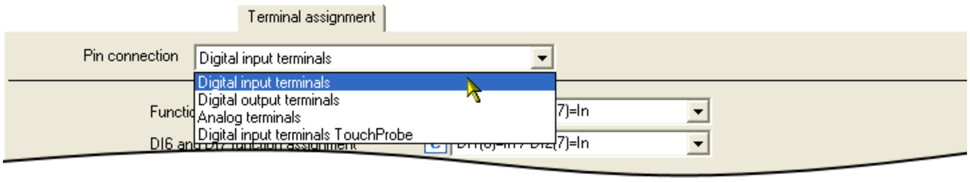
Whereas the configuration parameters referred to have to be parameterised with the keypad, implementation in the »Engineer« is much easier due to the availability of the corresponding dialogs. The following task illustrates the respective procedure.

Task: Starting from the preset technology application "Actuating drive speed" and the "Terminals 0" control mode, the DI2 digital input is to be used for choosing an alternative acceleration/deceleration time for the main setpoint instead of for choosing the fixed setpoint 2/3. To do this, the DI2 digital input is not to be linked to the *bJogSpeed2* input but to the *bJogRamp1* input of the application module.

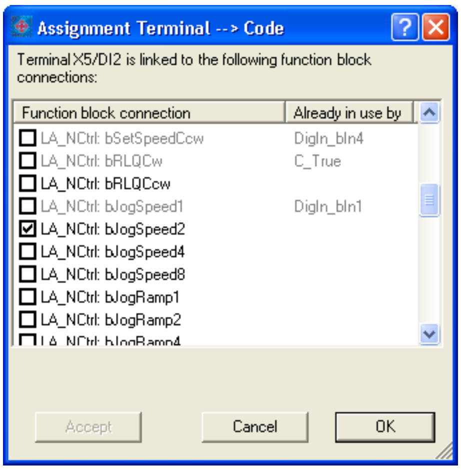
Possibility 1: Change terminal assignment by means of the Terminal Assignment tab

Procedure:

1. Go to the **Terminal Assignment** tab and select "Digital input terminals" in the **Control connections** list field:



2. Click on the ... button for the DI2 terminal in order to open the dialog box *Assignment Terminal --> Function block*.
   - In the list field, all block inputs that are currently logically linked to digital input DI2 are marked with a checkmark:

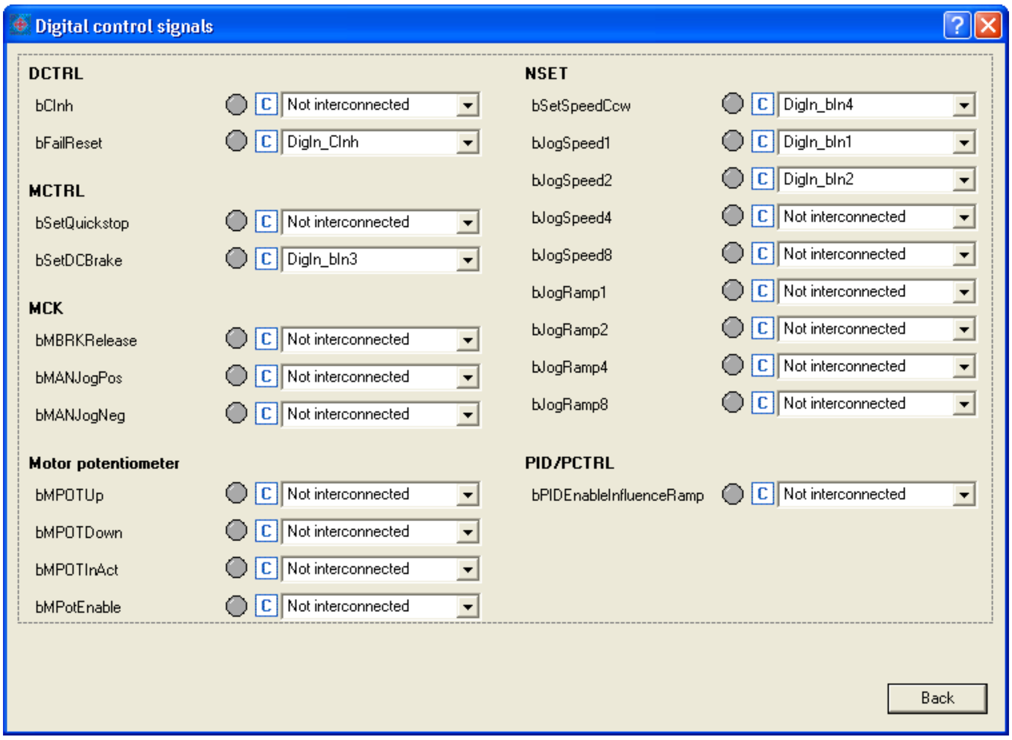


3. Remove checkmark for the connection **LA_NCtrl: bJogSpeed2** in order to cancel the existing logical link.
4. Set checkmark for connection **LA_NCtrl: bJogRamp1** in order to logically link this application input to digital input DI2.

Possibility 2: Change terminal assignment by means of the signal flow shown

Procedure:

1. Go to the **Application parameters** tab.
2. Go to the **Application Parameters** tab and click on the **Signal flow** button in order to change to the dialog level *Overview* ➞ *Signal flow*.
3. On the dialog level *Overview* ➞ *Signal flow*, click on the **Digital control signals** button in order to open the *Digital control signals* dialog box:

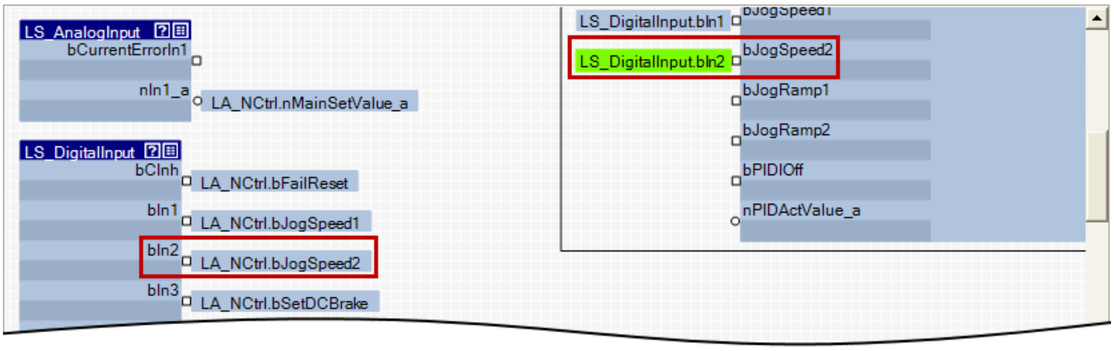


4. In the **bJogSpeed2** list field, set the selection "0: Not interconnected".
5. In the **bJogRamp1** list field, set the selection "16001: DigIn_bIn2".
6. Click on the **Back** button in order to close the dialog box again.

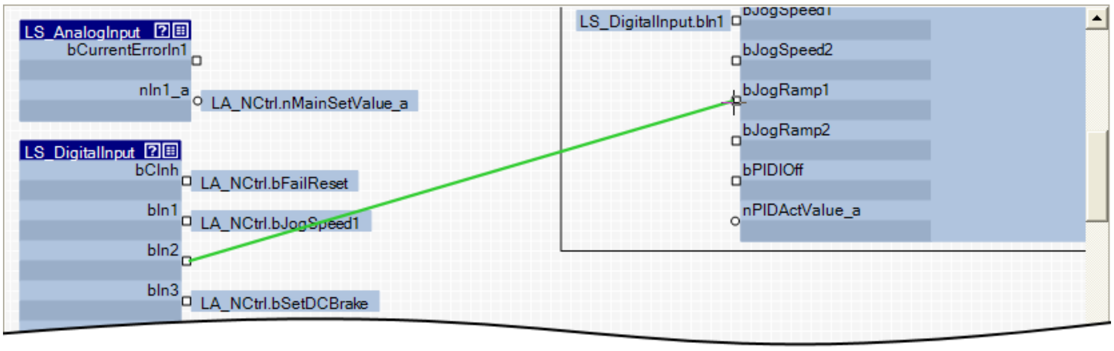
Possibility 3: Change terminal assignment with the FB editor

Procedure:

1. Go to the **FB Editor** tab.
2. Delete the existing interconnection from **LS_DigitalInput**.*bIn2* to **LA_NCtrl**.*bJogSpeed2*:



3. Establish a new interconnection from **LS_DigitalInput**.*bIn2* to **LA_NCtrl**.*bJogRamp1*:



Tip!

You can find detailed information on how to use the FB editor of the »Engineer« in the main chapter entitled "Working with the FB Editor". (📖 1416)

8 Technology applications

This chapter describes the handling and the functional range of the technology applications available for the 8400 TopLine inverter.

Technology application "Actuating drive speed"

This technology application preset in C00005 serves to solve speed-controlled drive tasks, e.g. conveyor drives (interconnected), extruders, test benches, vibrators, travelling drives, presses, machine tools, dosing systems.

▸ TA "Actuating drive speed" (📖 454)

Technology application "actuating drive speed (AC Drive profile)"

This technology application available from version 13.00.00 provides a speed and torque control by means of "AC Drive Profile". The fieldbuses EtherNet/IP™ and system bus (CANopen) are supported.

▸ TA "actuating drive speed (AC Drive Profile)" (📖 481)

Technology application "Table positioning"

This technology application serves to solve position-controlled drive tasks which are normally controlled by a higher-level control via a fieldbus, e.g. transport facilities, rotary tables, storage and retrieval units, feed drives, metering units, hoists.

Note: This TA requires an external sequence control!
From version 12.00.00, the FB L_Sequencer_1 can also be used as internal sequence control. This FB processes a positioning program based on a sequence table.

▸ TA "Table positioning" (📖 514)

"Switch-off positioning" technology application

This technology application is used to solve speed-controlled drive tasks which require a pre-switch off or stopping at certain positions, e.g. roller conveyors and conveying belts. The pre-switch off is implemented by connecting switch-off sensors.

▸ TA "Switch-off positioning" (📖 544)

Note!

Please note that the "StateLine", "HighLine" and "TopLine" device types differ with regard to the number, functional range, and flexibility of the technology applications offered.

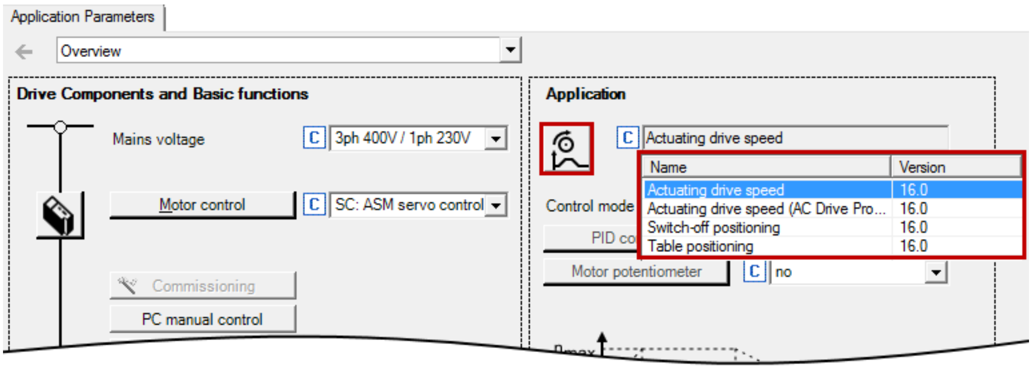
Related topics:

▸ Integrated technology applications (📖 36)
▸ Commissioning (📖 58)

8.1 Selection of the technology application and the control mode

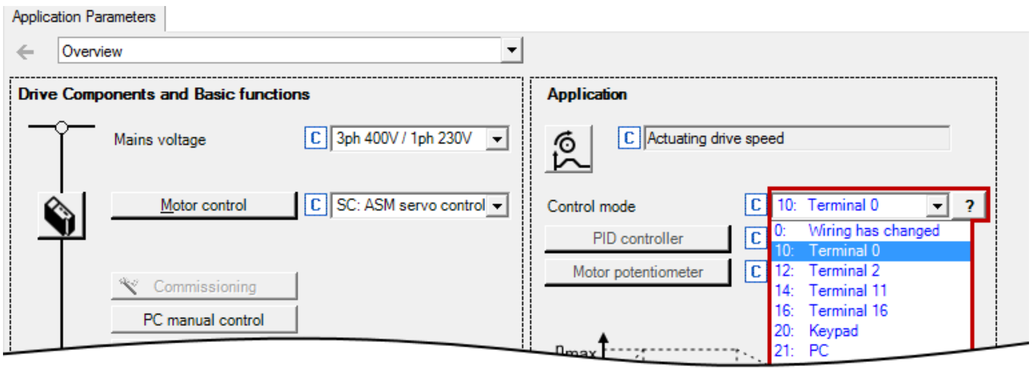
The technology application to be used is selected in C00005.

- You can select the technology application in the »Engineer« on the **Application parameter** tab via the **Application** list field:



Different control modes can be selected for every application in C00007. By selecting the control mode you set the way by which the technology application should be controlled, e.g. via terminals or via a fieldbus. The interconnection of the input/output terminals and ports shown in the FB editor in the I/O level changes accordingly.

- You can select the control mode in the »Engineer« on the **Application parameter** tab via the **Control mode** list field:



Tip!

You can infer the pre-configured assignment of the input/output terminals and ports for each control mode from the description of the corresponding technology application:

TA "Actuating drive speed": Terminal assignment of the control modes (📖 465)

TA "Table positioning": Terminal assignment of the control modes (📖 529)

TA "Switch-off positioning": Terminal assignment of the control modes (📖 555)

Detailed information on the individual configuration of the input/output terminals can be found in the description of the I/O terminals in the subchapter"User-defined terminal assignment". (📖 445)

8.2 TA "Actuating drive speed"

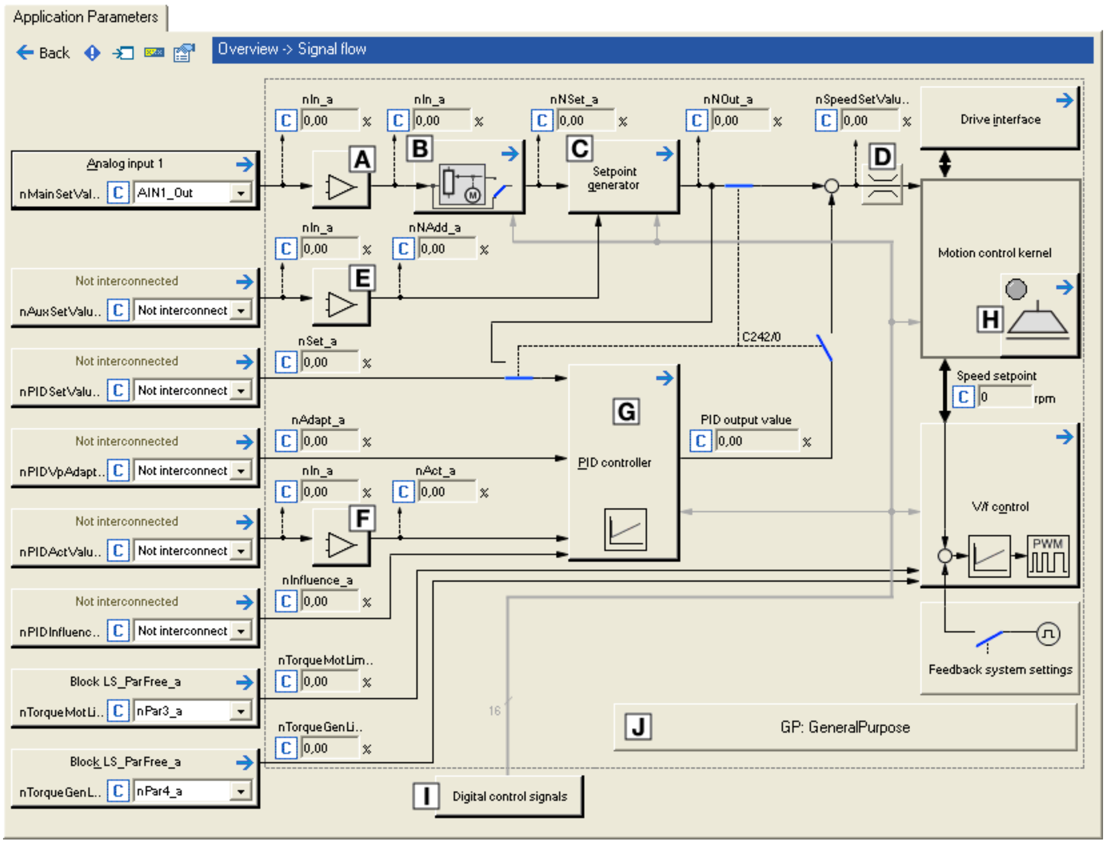
Product features

- Pre-configured control modes for terminals and bus control (with predefined process data connection to the fieldbus)
- Free configuration of input and output signals
- Offset, gain, and negation of main setpoint, additional setpoint, actual process controller value
- Up to 15 fixed setpoints for speed and ramp time
- Adjustable setpoint ramp times
- Freely selectable, variable ramp shape
- Automatic holding brake control
- Quick stop (QSP) with adjustable ramp time
- Motor potentiometer function
- Process controller
- Load monitoring *(in preparation)*
- Integrated, freely available "GeneralPurpose" functions:
  Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-flipflop
- Interface to the safety module (optional)
- Integration of encoder feedback

Related topics:

▶ Commissioning of the "Actuating drive speed" technology application (📖 65)

8.2.1 Basic signal flow



[8-1] Signal flow of the technology application "Actuating drive speed"

- A Main speed setpoint offset and gain (L_OffsetGainP_1)
- B Motor potentiometer function (L_MPot_1)
- C Setpoint generator (L_NSet_1)
- D Speed setpoint input limitation
- E Additional speed setpoint offset and gain (L_OffsetGainP_2)
- F Actual speed/sensor value offset and gain (L_OffsetGainP_3)
- G Process controller (L_PCTRL_1)
- H Holding brake control
- I Terminal assignment & display of digital control signals
- J Integrated disposable "GeneralPurpose" functions: Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-flipflop

Selection of the main speed setpoint

The main speed setpoint is selected in the Lenze setting via the analog input 1.

- Offset and gain of this input signal can be set in C00696 and C00670 for a simple signal adjustment of a setpoint encoder.
- Scaling: 16384 ≡ 100 % reference speed (C00011)
- The main setpoint is transformed to a speed setpoint in the setpoint encoder via a ramp function generator with linear or S-shaped ramps.
- Upstream to the ramp function generator, a blocking speed masking function and a setpoint MinMax limitation are effective.
- For a detailed functional description see the L_NSet FB.

Motor potentiometer function

Alternatively, the main speed setpoint can be generated via a motor potentiometer function.

- In the Lenze setting, the motor potentiometer function is deactivated.
- Activation is possible via C00806 or via the *bMPotEnable* input.
- The behaviour of the motor potentiometer during switch-on of the drive system can be selected in C00805.
- For a detailed functional description see the L_MPot FB.

Optional selection of an additional speed setpoint

You can optionally select an additional speed setpoint (e.g. as a correcting signal).

- The additional speed setpoint can be linked arithmetically with the main speed setpoint behind the ramp function generator.
- You must set the setpoint arithmetic to "1: NOut = NSet + NAdd" in C00190 in order to activate the additional speed setpoint.
- Offset and gain of this input signal can be set in C00697 and C00671 for a simple signal adjustment of a setpoint encoder.
- Scaling: 16384 ≡ 100 % reference speed (C00011)
- The acceleration and deceleration time for the additional speed setpoint can be set in C00220 and C00221.
- For a detailed functional description see the L_NSet FB.

Tip!

In the case of a grinding machine, the additional speed setpoint can, for instance, be used to control a constant circumferential speed while the grinding disk diameter is reduced.

8.2.2 Internal interfaces | application block "LA_NCtrl"

Note!

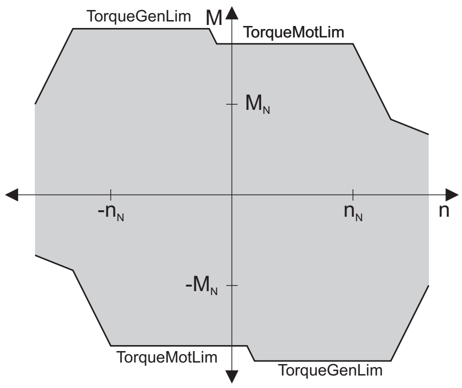
The connectors greyed out in the following table are hidden in the function block editor in the Lenze setting.

- These connections can be shown via the **Connector visibilities** command in the *Context menu* of the application block.

inputs

Designator Data type	Information/possible settings	
wCANDriveControl WORD	Control word via system bus (CAN) for device control • See the "wCANControl/wMCIControl control words" subchapter of the chapter on device control for a detailed description of the individual control bits.	
wMCIDriveControl WORD	Control word via communication module (e.g. PROFIBUS) for device control • See the "wCANControl/wMCIControl control words" subchapter of the chapter on device control for a detailed description of the individual control bits.	
wSMControl WORD	Interface to the optional safety system. • Setting control bit 0 ("SafeStop1") in this control word causes e.g. the automatic deceleration of the drive to standstill within this application (in the **Motion Control Kernel**). • See the subchapter "Interface to safety system" of the chapter on basic drive functions for a detailed description of the individual control bits.	
bCInh BOOL	Enable/inhibit inverter	
	FALSE	Enable inverter: The inverter switches to the "OperationEnabled" device status if no other source for controller inhibit is active. • C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.
	TRUE	Inhibit inverter (controller inhibit): The inverter switches to the "SwitchedOn" device status.
bFailReset BOOL	Reset error message In the Lenze setting this input is connected to the digital input controller enable so that a possibly existing error message is reset together with the controller enable (if the cause for the fault is eliminated).	
	TRUE	The current fault is reset, if the cause for the fault is eliminated. • If the fault still exists, the error status remains unchanged.
bSetQuickstop BOOL	Activate quick stop (QSP) • Also see device command "Activate/deactivate quick stop".	
	TRUE	Activate quick stop • Motor control is decoupled from the setpoint selection and, within the deceleration time parameterised in C00105, the motor is brought to a standstill ($n_{act} = 0$). • The motor is kept at a standstill during closed-loop operation. • A pulse inhibit is set if the auto-DCB function has been activated via C00019.
	FALSE	Deactivate quick stop • The quick stop is deactivated if no other source for the quick stop is active. • C00159 displays a bit code of active sources/causes for the quick stop.

Designator Data type	Information/possible settings	
bSetDCBrake BOOL	Manual DC-injection braking (DCB) • Detailed information on DC-injection braking is provided in the motor control chapter, subchapter "DC-injection braking".	
	Note! Holding braking is not possible when this braking mode is used! Use the basic "Holding brake control" function for controlling the holding brake with a low rate of wear.	
	FALSE	Deactivate DC-injection braking.
	TRUE	Activate DC-injection braking, i.e. the drive is brought to a standstill by means of DC-injection braking. • The braking effect stops when the rotor is at standstill. • After the hold time (C00107) has expired, the controller sets the pulse inhibit.
bRFG_Stop BOOL	Ramp function generator: Maintain the current value of the main setpoint integrator • The speed, for instance, of a running ramp process is immediately kept constant when *bRFG_Stop* is activated. At the same time, the acceleration/deceleration jumps to the value "0". • For a detailed functional description see the L_NSet FB.	
	TRUE	The current value of the main setpoint integrator is held.
bRFG_0 BOOL	Ramp function generator: Lead the main setpoint integrator to "0" within the current Ti times • For a detailed functional description see the L_NSet FB.	
	TRUE	The current value of the main setpoint integrator is led to "0" within the Ti time set.
nVoltageAdd_a INT	Additive voltage impression • An additional setpoint for the motor voltage can be specified via this process input. • If there are, for instance, different loads at the motor output end, it is possible to apply a voltage boost at the starting time. • If the value is negative, the voltage is reduced. • Scaling: 16384 ≡ 1000 V	
	Stop! Values selected too high may cause the motor to heat up due to the resulting current!	
nBoost_a INT	Additional setpoint for the motor voltage at speed = 0 • The entire voltage-frequency characteristic is provided with an offset. • Scaling: 16384 ≡ 1000 V	
	Stop! Values selected too high may cause the motor to heat up due to the resulting current!	
nPWMAngleOffset INT	Additional offset for the electrical angle of rotation • If a torque is connected, e.g. dynamic acceleration processes can be generated. • Scaling: ±32767 ≡ ±180 ° angle of rotation	

Designator Data type	Information/possible settings
nTorqueMotLim_a nTorqueGenLim_a INT	Torque limitation in motor mode and in generator mode • These input signals are directly transferred to the motor control to limit the inverter's maximum torque in motor and generator mode. • The drive cannot output a higher torque in motor/generator mode than set here. • The applied values (any polarity) are internally interpreted as absolute values. • If V/f characteristic control (VFCplus) is selected, limitation is indirectly performed via a so-called I_{max} controller. • If sensorless vector control (SLVC) or servo control (SC) is selected, limitation has a direct effect on the torque-producing current component. • Scaling: 16384 ≡ 100 % M_{max} (C00057)
	Torque limits in motor and generator mode: 
bSetSpeedCcw BOOL	Change of direction of rotation • For instance if a motor or gearbox is fixed laterally reversed to a machine part, but the setpoint selection should still be executed for the positive direction of rotation. FALSE: Clockwise rotation (Cw) TRUE: Direction of rotation to the left (Ccw)
bRLQCw BOOL	Activate clockwise rotation (fail-safe) • For a detailed functional description see the L_RLQ FB. FALSE: Quick stop TRUE: CW rotation
bRLQCcw BOOL	Activate counter-clockwise rotation (fail-safe) • For a detailed functional description see the L_RLQ FB. FALSE: Quick stop TRUE: CCW rotation
nMainSetValue_a INT	Main speed setpoint • Offset and gain of this input signal can be set in C00696 and C00670 for a simple signal adjustment of a setpoint encoder. • Scaling: 16384 ≡ 100 % reference speed (C00011) • The main setpoint is transformed to a speed setpoint in the setpoint encoder via a ramp function generator with linear or S-shaped ramps. • Upstream to the ramp function generator, a blocking speed masking function and a setpoint MinMax limitation are effective. • For a detailed functional description see the L_NSet FB.

Designator Data type	Information/possible settings
nAuxSetValue_a INT	Additional speed setpoint • The additional speed setpoint can be linked arithmetically with the main speed setpoint behind the ramp function generator. • You must set the setpoint arithmetic to "1: NOut = NSet + NAdd" in C00190 in order to activate the additional speed setpoint. • Offset and gain of this input signal can be set in C00697 and C00671 for a simple signal adjustment of a setpoint encoder. • Scaling: 16384 ≡ 100 % reference speed (C00011) • The acceleration and deceleration time for the additional speed setpoint can be set in C00220 and C00221. • For a detailed functional description see the L_NSet FB.
bJogSpeed1 bJogSpeed2 BOOL bJogSpeed4 bJogSpeed8 BOOL	Inputs for overriding fixed setpoints (JOG setpoints) for the main setpoint • A fixed setpoint for the setpoint generator can be activated instead of the main setpoint via these selection inputs. • The four selection inputs are binary coded, therefore 15 fixed setpoints can be selected. • In the case of binary coded selection "0" (all inputs = FALSE or not assigned), main setpoint *nMainSetValue_a* is active. • The selection of the fixed setpoints is carried out in C00039/1...15 in [%] based on the reference speed (C00011). • For a detailed functional description see the L_NSet FB.
bJogRamp1 bJogRamp2 BOOL bJogRamp4 bJogRamp8 BOOL	Selection inputs for alternative acceleration/deceleration times for the main setpoint • The four selection inputs are binary coded, therefore 15 alternative acceleration/deceleration times can be selected. • For main setpoint *nMainSetValue_a*, the set acceleration time (C00012) and deceleration time (C00013) are active in the case of the binary coded selection "0" (all inputs = FALSE or not assigned). • Alternative acceleration times are selected in C00101/1...15. • The selection of the alternative deceleration times is carried out in C00103/1...15. • For a detailed functional description see the L_NSet FB.
Motor potentiometer Alternatively to the input signal *nMainSetValue_a*, the main setpoint can also be generated by a motor potentiometer function. • In the Lenze setting, the motor potentiometer function is deactivated. • Activation is possible via C00806 or via the *bMPotEnable* input. • The behaviour of the motor potentiometer during switch-on of the drive system can be selected in C00805. • For a detailed functional description see the L_MPot FB.	
bMPotEnable BOOL	Activating the motor potentiometer function • This input and C00806 are OR'd. TRUE: The motor potentiometer function is active; the speed setpoint can be changed via the *bMPotUp* and *bMPotDown* control inputs.
bMPotUp BOOL	Increasing the speed setpoint TRUE: Approach the upper speed limit value set in C00800 with the acceleration time set in C00802.
bMPotInAct BOOL	Activating the inactive function TRUE: The speed setpoint behaves according to the inactive function set in C00804. • In the Lenze setting, the speed setpoint is maintained.
bMPotDown BOOL	Decreasing the speed setpoint TRUE: Approach the lower speed limit value set In C00801 with the deceleration time set in C00803.

Designator Data type	Information/possible settings	
Process controller • In the Lenze setting, the process controller is deactivated. • The activation is executed by selecting the operating mode in C00242. • For a detailed functional description see FB L_PCTRL.		
bPIDEnableInfluenceRamp BOOL	Activate ramp for influencing factor	
	FALSE	Influencing factor of the PID controller is ramped down to "0".
	TRUE	Influencing factor of the PID controller is ramped up to the value *nPIDInfluence_a*.
bPIDIOff BOOL	Switch off the I-component of the process controller • In conjunction with the operating mode set in C00242 (Lenze setting: "Off").	
	TRUE	I-component of the process controller is switched off.
nPIDVpAdapt_a INT	Adaptation of gain Vp set in C00222 in percent • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 % • Changes can be done online.	
nPIDSetValue_a INT	Sensor and process setpoint for operating modes 2, 4 and 5 • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 %	
nPIDActValue_a INT	Speed or actual sensor value (actual process value) • Offset and gain for this input signal can be set in C00698 and C00672. • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 %	
nPIDInfluence_a INT	Limitation of the influencing factor in percent • The influence factor of the PID controller can be limited to a certain value (-199.99% ... + 199.99%) via *nPIDInfluence_a*. • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 %	
MCK basic functions		
bMBrakeRelease BOOL	Holding brake control: Release/apply brake • In conjunction with the operating mode selected in C02580 (Lenze setting: "Brake control off").	
	FALSE	Apply brake. • During automatic operation, the internal brake logic controls the brake.
	TRUE	Release brake manually (forced release). • **Note!** The brake can also be released when the controller is inhibited! • During automatic operation, the internal brake logic is deactivated and the brake is released (supervisor operation). If a controller inhibit has been set by the brake control, it will be deactivated. • In semi-automatic operation, the brake is released including feedforward control.

Designator	Data type	Information/possible settings
GP: GeneralPurpose The following inputs are interconnected with logic/arithmetic functions on application level for free usage. "GeneralPurpose" functions		
bGPFree1 ... bGPFree2 (from version 12.00.00)	BOOL	Free inputs for digital signals • Digital signals can be transferred from the I/O level to the application level via these inputs.
nGPAnalogSwitchIn1_a nGPAnalogSwitchIn2_a	INT	Analog switch: Input signals • The input signal selected via the selection input *bGPAnalogSwitchSet* is output at output *nGPAnalogSwitchOut_a*.
bGPAnalogSwitchSet	BOOL	Analog switch: Selection input FALSE: *nGPAnalogSwitchOut_a = nGPAnalogSwitchIn1_a* TRUE: *nGPAnalogSwitchOut_a = nGPAnalogSwitchIn2_a*
nGPArithmetikIn1_a nGPArithmetikIn2_a	INT	Arithmetic: Input signals • The arithmetic function is selected in C00338. • The result is output at output *nGPArithmetikOut_a*.
nGPMulDivIn_a	INT	Multiplication/Division: Input signal • The factor for the multiplication can be set in C00699/1 (numerator) and C00699/2 (denominator). • The result is output at output *nGPMulDivOut_a*.
bGPDigitalDelayIn	BOOL	Binary delay element: Input signal • The on-delay can be set in C00720/1. • The off-delay can be set in C00720/2. • The time-delayed input signal is output at output *bGPDigitalDelayOut*.
bGPLogicIn1 bGPLogicIn2 bGPLogicIn3	BOOL	Binary logic: Input signals • The logic operation is selected in C00820. • The result is output at output *bGPLogicOut*.
nGPCompareIn1_a nGPCompareIn2_a	INT	Analog comparison: Input signals • The comparison operation is selected in C00680. • Hysteresis and window size can be set in C00680 and C00682. • If the comparison statement is true, the output *bGPCompareOut* will be set to TRUE.
bGPDFlipFlop_InD bGPDFlipFlop_InClk bGPDFlipFlop_InClr	BOOL	D-FlipFlop: Input signals • Data, clock and reset input
Free inputs The following inputs can freely be interconnected on the application level. The signals can be transferred from the I/O level to the application level via these inputs.		
bFreeIn1 ... bFreeIn8	BOOL	Free inputs for digital signals
wFreeIn1 ... wFreeIn4	WORD	Free inputs for 16-bit signals

outputs

Designator / Data type	Value/meaning	
wDriveControlStatus WORD	Status word of the inverter • The status word contains information on the currents status of the inverter. • See the "wDeviceStatusWord status word" subchapter of the chapter on device control for a detailed description of the bit assignment.	
wStateDetermFailNoLow WORD	Display of the status determining error (LOW word)	
wStateDetermFailNoHigh WORD	Display of the status determining error (HIGH word)	
bDriveFail BOOL	TRUE	Inverter in error status. • "Fault" device status is active.
bDriveReady BOOL	TRUE	Inverter is ready for operation • "SwitchedOn" device status is active. • The drive is in this device status if the DC bus voltage is applied and the inverter is still inhibited by the user (controller inhibit).
bCInhActive BOOL	TRUE	Controller inhibit is active.
bQSPIsActive BOOL	TRUE	Quick stop is active.
bSpeedCcw BOOL	Current direction of rotation	
	FALSE	Clockwise rotation (Cw)
	TRUE	Direction of rotation to the left (Ccw)
bSpeedActCompare BOOL	Result of the speed comparison (detection of speed=0)	
	TRUE	During open-loop operation: Speed setpoint < Comparison value (C00024)
		During closed-loop operation: Actual speed value < Comparison value (C00024)
bOverLoadActive BOOL	In preparation (output is not interconnected on the application level)	
bUnderLoadActive BOOL	In preparation (output is not interconnected on the application level)	
bImaxActive BOOL	"Current setpoint inside the limitation" status signal	
	TRUE	The current setpoint is internally limited (the inverter operates at the maximum current limit).
bSpeedSetReached BOOL	Status signal "setpoint = 0"	
	TRUE	Speed setpoint from the ramp function generator = 0
bSpeedActEqSet BOOL	TRUE	Actual speed value = speed setpoint
nMotorCurrent_a INT	Current stator current/effective motor current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	
nMotorSpeedSet_a INT	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)	
nMotorSpeedAct_a INT	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)	
nMotorTorqueAct_a INT	Actual torque • In the "VFC (+encoder)" operating mode of the motor control, this value is determined from the current motor current and corresponds to the actual torque only by approximation. • Scaling: 16384 ≡ 100 % M_{max} (C00057)	

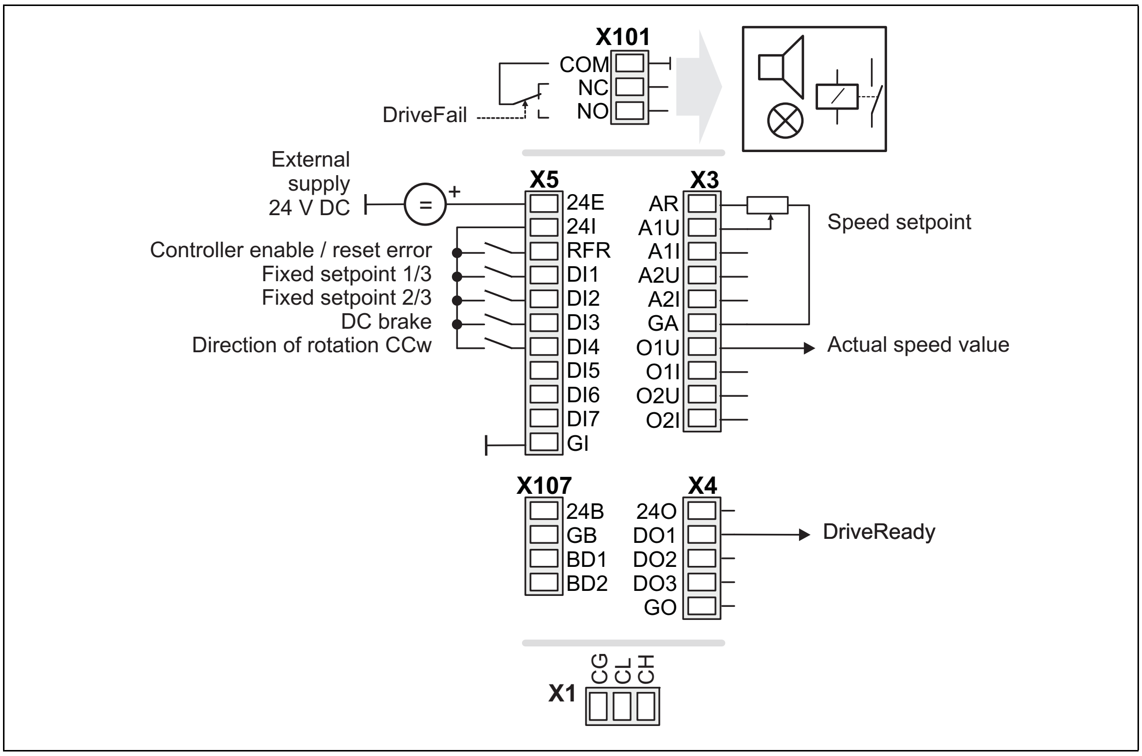
<table>
<tr><th>Designator
Data type</th><th colspan="2">Value/meaning</th></tr>
<tr><td>nDCVoltage_a
INT</td><td colspan="2">Current DC-bus voltage
• Scaling: 16384 ≡ 1000 V</td></tr>
<tr><td>nMotorVoltage_a
INT</td><td colspan="2">Current motor voltage/inverter output voltage
• Scaling: 16384 ≡ 1000 V</td></tr>
<tr><td colspan="3">MCK basic functions</td></tr>
<tr><td rowspan="3">bMBrakeReleaseOut
BOOL</td><td colspan="2">Holding brake control: Trigger signal for the holding brake control switching element via a digital output
• Use bit 0 in C02582 to activate inverted switching element triggering.</td></tr>
<tr><td>FALSE</td><td>Apply brake.</td></tr>
<tr><td>TRUE</td><td>Release brake.</td></tr>
<tr><td rowspan="2">bMBrakeReleased
BOOL</td><td colspan="2">Holding brake control: "Brake released" considering the brake release time
• When the holding brake is triggered to close, bMBrakeReleased is immediately set to FALSE even if the brake closing time has not yet elapsed!</td></tr>
<tr><td>TRUE</td><td>Brake released (after the brake release time has expired).</td></tr>
<tr><td colspan="3">GP: GeneralPurpose
The following outputs are interconnected with logic/arithmetic functions on application level for free usage.
"GeneralPurpose" functions</td></tr>
<tr><td>nGPAnalogSwitchInOut_a
INT</td><td colspan="2">Analog switch: Output signal</td></tr>
<tr><td>nGPArithmetikOut_a
INT</td><td colspan="2">Arithmetic: Output signal</td></tr>
<tr><td>nGPMulDivOut_a
INT</td><td colspan="2">Multiplication/Division: Output signal</td></tr>
<tr><td>bGPDigitalDelayOut
BOOL</td><td colspan="2">Binary delay element: Output signal</td></tr>
<tr><td>bGPLogicOut
BOOL</td><td colspan="2">Binary logic: Output signal</td></tr>
<tr><td>bGPCompareOut
BOOL</td><td colspan="2">Analog comparison: Output signal</td></tr>
<tr><td>bGPSignalOut1
...
bGPSignalOut4
BOOL</td><td colspan="2">Binary signal monitor: Output signals
• The signal sources to be output are selected in C00411/1...4.
• A bit coded inversion of the output signals can be parameterised in C00412.</td></tr>
<tr><td>nGPSignalOut1_a
...
nGPSignalOut4_a
BOOL</td><td colspan="2">Analog signal monitor: Output signals
• The signal sources to be output are selected in C00410/1...4.
• Gain and offset for each output signal can be parameterised in C00413/1...8.</td></tr>
<tr><td>bGPDFlipFlop_Out
BOOL</td><td colspan="2">D-FlipFlop: Output signal</td></tr>
<tr><td>bGPDFlipFlop_NegOut
BOOL</td><td colspan="2">D-FlipFlop: Negated output signal</td></tr>
<tr><td colspan="3">Free outputs
The following outputs can freely be interconnected on the application level.
The signals from the application level can be transferred to the I/O level via these outputs.</td></tr>
<tr><td>bFreeOut1 ... bFreeOut8
BOOL</td><td colspan="2">Free outputs for digital signals</td></tr>
<tr><td>wFreeOut1 ... wFreeOut4
WORD</td><td colspan="2">Free outputs for 16-bit signals</td></tr>
</table>

8.2.3 Terminal assignment of the control modes

The following comparison provides information about which inputs/outputs of the application block **LA_NCtrl** are interconnected to the digital and analog input/output terminals of the inverter in the different control modes.

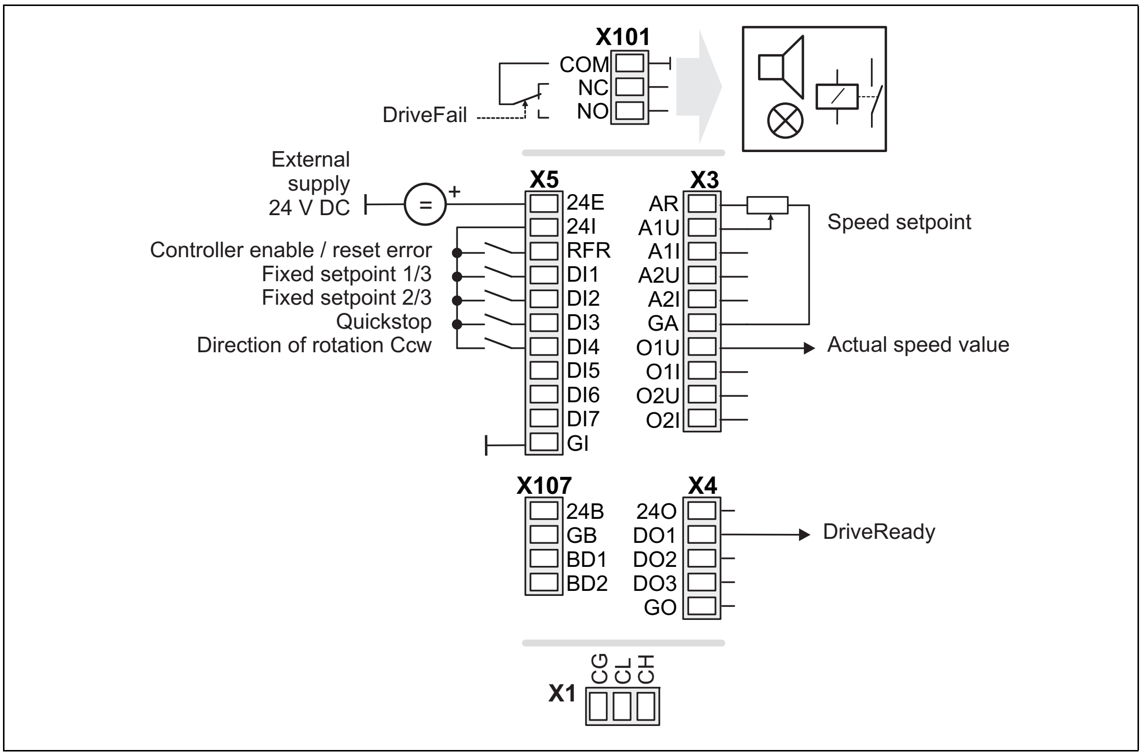
	Control mode (C00007)							
	10: Terminals 0	12: Terminals 2	14: Terminals 11	16: Terminal 16	20: Keypad	21: PC	30: CAN	40: MCI
Digital input terminals								
X5/RFR	**Controller enable / Reset of error message** bFailReset							
X5/DI1	**Fixed setpoint 1/3** bJogSpeed1		**Change of direction of rotation** bSetSpeedCcw	**Fixed setpoint 1/3** bJogSpeed1	-	-	**Quick stop** bSetQuickstop	
X5/DI2	**Fixed setpoint 2/3** bJogSpeed2		**Activate manual DC-injection braking (DCB)** bSetDCBrake	**Fixed setpoint 2/3** bJogSpeed2	-	-	-	-
X5/DI3	**Activate manual DC-injection braking (DCB)** bSetDCBrake	**Quick stop** bSetQuickstop	**Motor potentiometer: Increase speed** bMPotUp	**CW rotation quick stop** bRLQCw	-	-	-	-
X5/DI4	**Change of direction of rotation** bSetSpeedCcw		**Motor potentiometer: Decrease speed** bMPotDown	**CCW rotation quick stop** bRLQCcw	-	-	-	-
X5/DI5 ... DI7	-	-	-	-	-	-	-	-
Analog input terminals								
X3/A1U, A1I	**Main speed setpoint** nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)				-	-	**Additional speed setpoint** nAuxSetValue_a 10 V ≡ 100 % reference speed (C00011)	
X3/A2U, A2I	-	-	-	-	-	-	-	-
Digital output terminals								
X4/DO1	**Status "Drive is ready"** bDriveReady							
X4/DO2 ... DO3	-	-	-	-	-	-	-	-
X107/BD1, BD2	-	-	-	-	-	-	-	-
X101/COM, NO	**Status "Error is pending"** bDriveFail							
Analog output terminals								
X3/O1U	**Actual speed value** nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)							
X3/O1I	-	-	-	-	-	-	-	-
X3/O2U								
X3/O2I								

8.2.3.1 Terminals 0



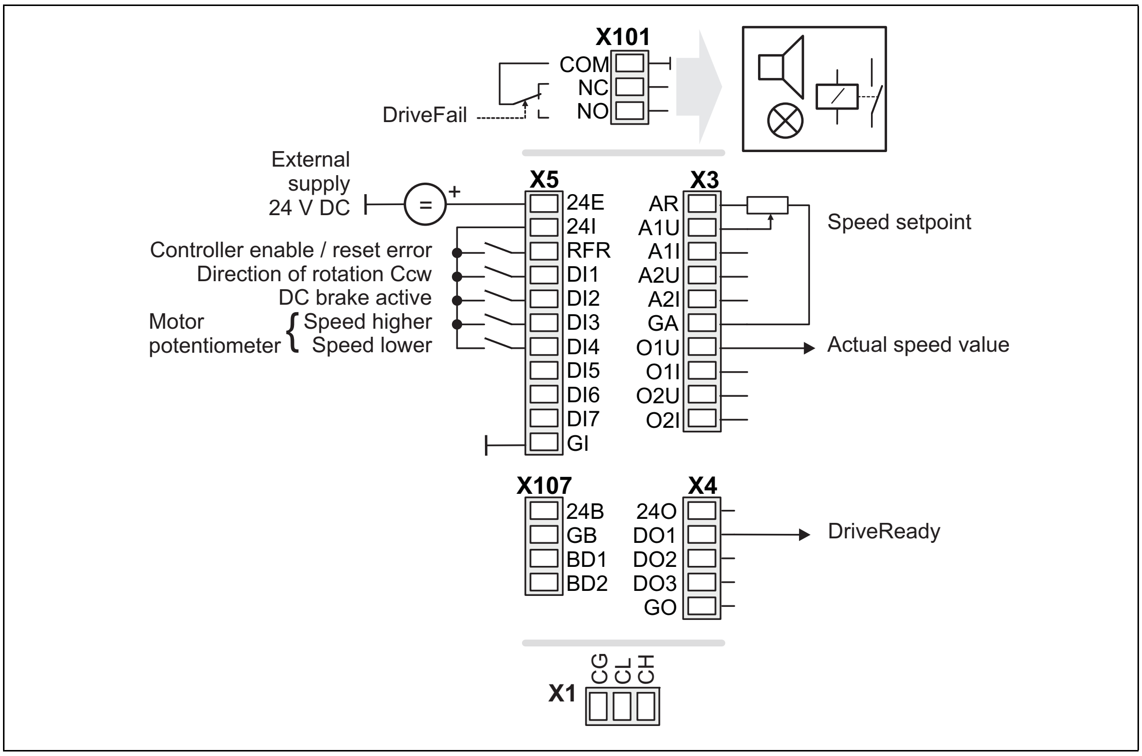
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nMainSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl.bSetDCBrake		**X3/A2I**	-
X5/DI4	LA_NCtrl.bSetSpeedCcw		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.2.3.2 Terminals 2



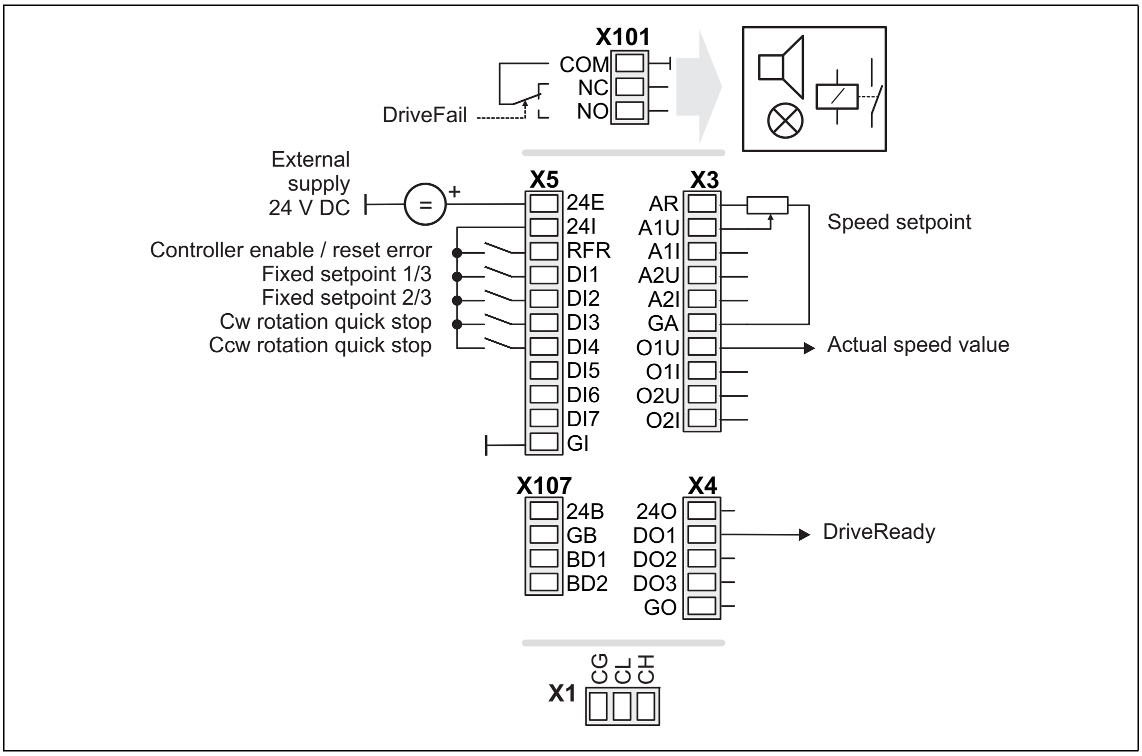
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nMainSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl.bSetQuickstop		**X3/A2I**	-
X5/DI4	LA_NCtrl.bSetSpeedCcw		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.2.3.3 Terminals 11



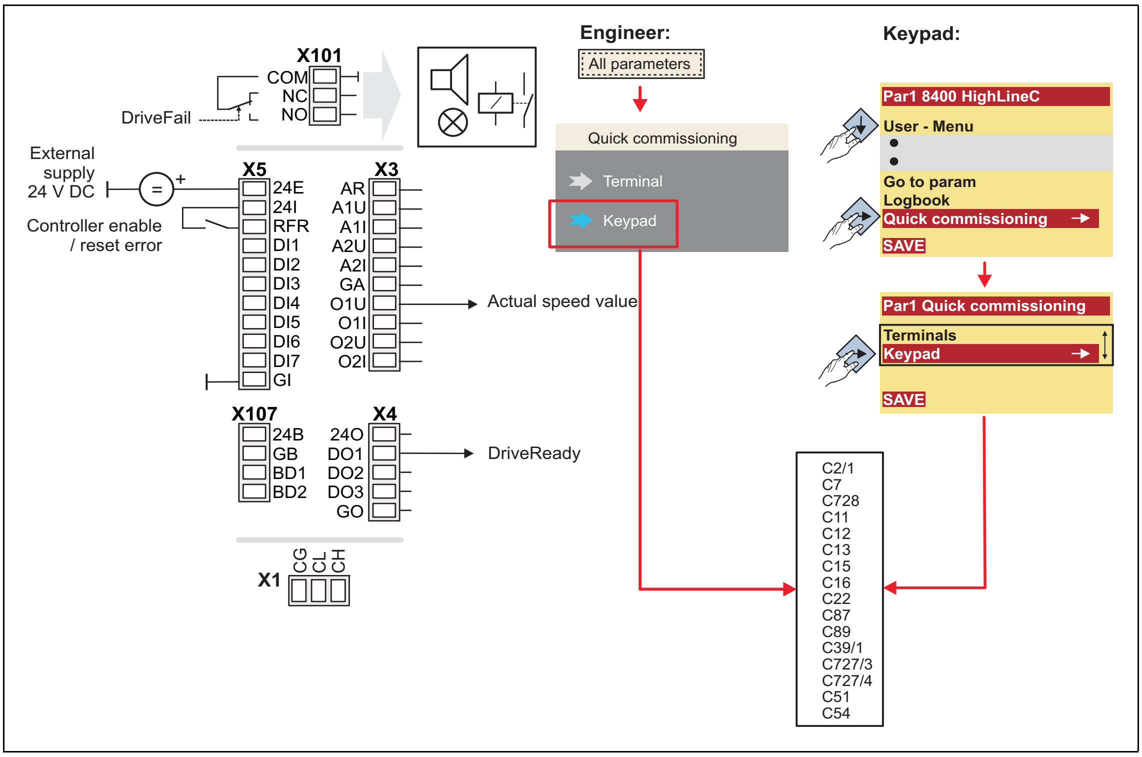
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nMainSetValue_a *
X5/DI1	LA_NCtrl.bSetSpeedCcw		**X3/A1I**	-
X5/DI2	LA_NCtrl.bSetDCBrake		**X3/A2U**	-
X5/DI3	LA_NCtrl.bMPotUp		**X3/A2I**	-
X5/DI4	LA_NCtrl.bMPotDown		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.2.3.4 Terminal 16



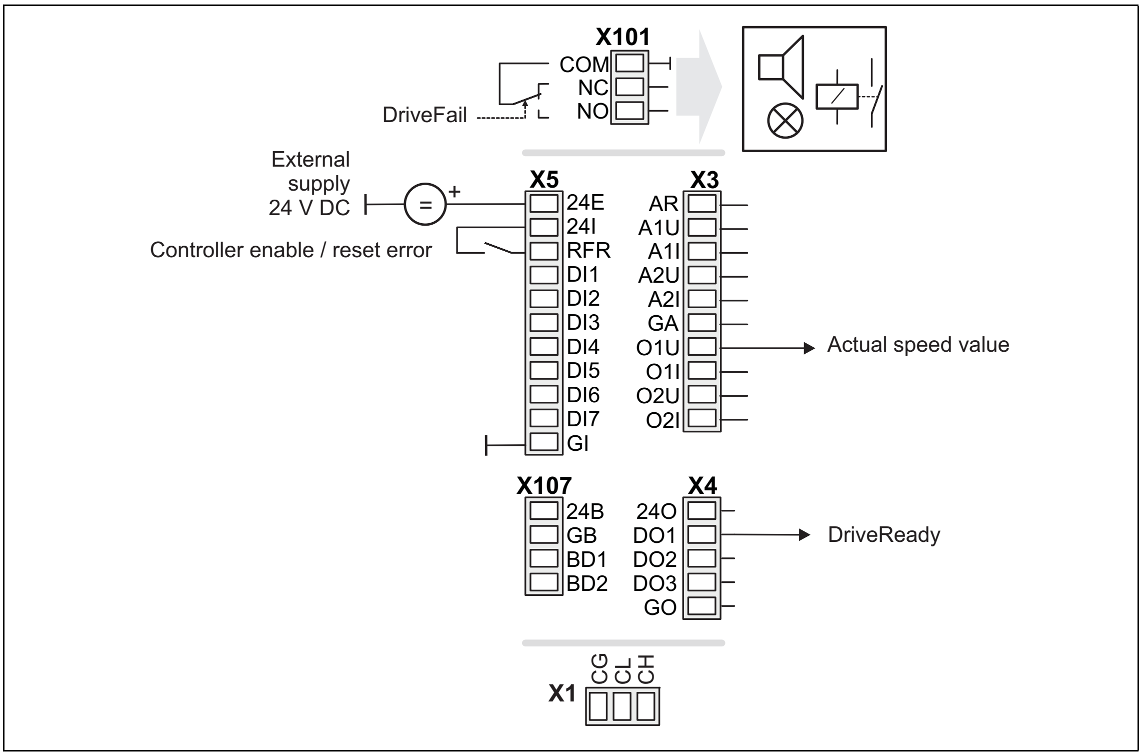
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nMainSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl. bRLQCw		**X3/A2I**	-
X5/DI4	LA_NCtrl. bRLQCcw		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.2.3.5 Keypad



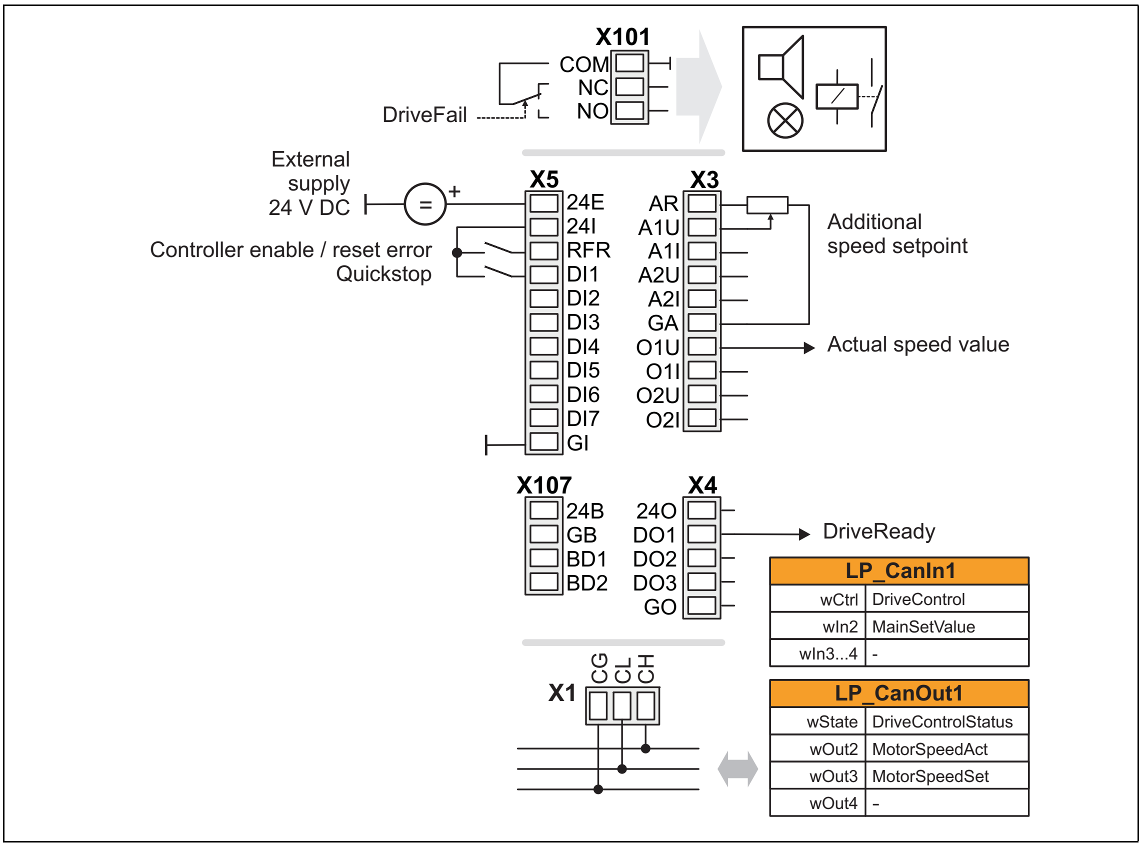
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		X3/A1U	-
X5/DI1	-		X3/A1I	-
X5/DI2	-		X3/A2U	-
X5/DI3	-		X3/A2I	-
X5/DI4	-		X3/O1U	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		X3/O1I	-
X5/DI6	-		X3/O2U	-
X5/DI7	-		X3/O2I	-
			* 10 V ≡ 100 % reference speed (C00011)	
X107/BD1	-		X4/DO1	LA_NCtrl.bDriveReady
X107/BD2	-		X4/DO2	-
			X4/DO3	-

8.2.3.6 PC



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		X3/A1U	-
X5/DI1	-		X3/A1I	-
X5/DI2	-		X3/A2U	-
X5/DI3	-		X3/A2I	-
X5/DI4	-		X3/O1U	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		X3/O1I	-
X5/DI6	-		X3/O2U	-
X5/DI7	-		X3/O2I	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		X4/DO1	LA_NCtrl.bDriveReady
X107/BD2	-		X4/DO2	-
			X4/DO3	-

8.2.3.7 CAN



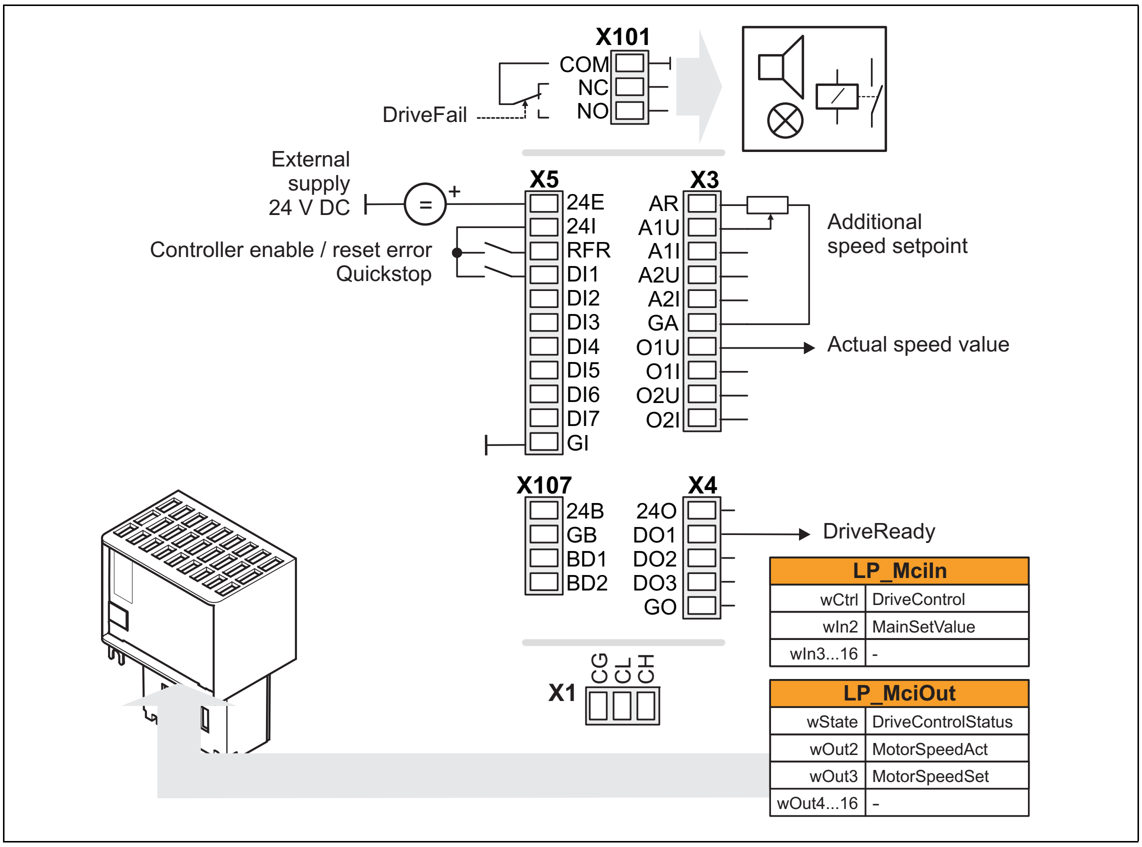
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bSetQuickStop		**X3/A1I**	-
X5/DI2	-		**X3/A2U**	-
X5/DI3	-		**X3/A2I**	-
X5/DI4	-		**X3/O1U**	LA_NCtrl.NMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

▸ Process data assignment for fieldbus communication (📖 474)

Note!

- You must set the setpoint arithmetic in C00190 to "1: NOut = NSet + NAdd" so that the additional speed setpoint selected via the analog input A1U has an additive effect.
- The "manual jog" function via digital terminals is being prepared!

8.2.3.8 MCI



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		X3/A1U	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bSetQuickStop		X3/A1I	-
X5/DI2	-		X3/A2U	-
X5/DI3	-		X3/A2I	-
X5/DI4	-		X3/O1U	LA_NCtrl.NMotorSpeedAct_a *
X5/DI5	-		X3/O1I	-
X5/DI6	-		X3/O2U	-
X5/DI7	-		X3/O2I	-
			* 10 V ≡ 100 % reference speed (C00011)	
X107/BD1	-		X4/DO1	LA_NCtrl.bDriveReady
X107/BD2	-		X4/DO2	-
			X4/DO3	-

▸ Process data assignment for fieldbus communication (📖 474)

Note!

- You must set the setpoint arithmetic in C00190 to "1: NOut = NSet + NAdd" so that the additional speed setpoint selected via the analog input A1U has an additive effect.
- The "manual jog" function via digital terminals is being prepared!

8.2.4 Process data assignment for fieldbus communication

The fieldbus communication is connected (preconfigured) to the previously selected technology application by selecting the corresponding control mode in C00007:

- "30: CAN" for the connection to the system bus (CAN)
- "40: MCI" for the connection to a plugged-on communication module (e.g. PROFIBUS)

The assignment of the process data words depends only on the application, not on the bus system used:

Input words	Name	Assignment
Word 1	DriveControl	Control word • For bit assignment see the table below.
Word 2	MainSetValue	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	-	Not preconfigured
Word 4	-	Not preconfigured
Words 5 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Control word	Name	Function
Bit 0	SwitchOn	1 ≡ Change to the "SwitchedOn" device status • This bit must be set in the CAN/MCI control word to ensure that the device changes to the "SwitchedOn" device status after mains connection without the need for a master control specifying this bit via fieldbus. • If control via a bus system is not wanted (e.g. in the case of control via terminals), the *wDriveCtrl* output signal of the LS_ParFix system block can be connected to the control word inputs.
Bit 1	DisableVoltage	1 ≡ Inhibit inverter control (pulse inhibit)
Bit 2	SetQuickStop	1 ≡ Activate quick stop (QSP). ▸ Activate/deactivate quick stop (📖 114)
Bit 3	EnableOperation	1 ≡ Enable inverter (RFR) • If control via terminals is performed, this bit must be set both in the CAN control word and in the MCI control word. Otherwise, the controller is inhibited. ▸ Enable/inhibit inverter (📖 113)
Bit 4	ModeSpecific_1	Reserved (currently not assigned)
Bit 5	ModeSpecific_2	
Bit 6	ModeSpecific_3	
Bit 7	ResetFault	1 ≡ Reset fault (trip reset) • Acknowledge error message (if the error cause has been eliminated). ▸ Reset error (📖 117)
Bit 8	SetHalt	1 ≡ Activate stop function • Stop drive via stopping ramp (in preparation).
Bit 9	reserved_1	Reserved (currently not assigned)
Bit 10	reserved_2	
Bit 11	SetDCBrake	1 ≡ Activate DC-injection braking ▸ Manual DC-injection braking (DCB) (📖 284)
Bit 12	JogSpeed1	Activation of fixed speed 1 ... 3
Bit 13	JogSpeed2	

Control word	Name	Function
Bit 14	SetFail	1 ≡ Set error (trip set)
Bit 15	SetSpeedCcw	0 ≡ Direction of rotation to the right (Cw) 1 ≡ Direction of rotation to the left (Ccw)

Output words	Name	Assignment
Word 1	DriveControlStatus	Status word • For bit assignment see the table below.
Word 2	MotorSpeedAct	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	MotorSpeedSet	Resulting overall setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 4	-	Not preconfigured
Words 5 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Status word	Name	Status
Bit 0	FreeStatusBit0	Free status bit 0 (not assigned, freely assignable)
Bit 1	PowerDisabled	1 ≡ Inverter control inhibited (pulse inhibit is active)
Bit 2	FreeStatusBit2	Free status bit 2 (not assigned, freely assignable)
Bit 3	FreeStatusBit3	Free status bit 3 (not assigned, freely assignable)
Bit 4	FreeStatusBit4	Free status bit 4 (not assigned, freely assignable)
Bit 5	FreeStatusBit5	Free status bit 5 (not assigned, freely assignable)
Bit 6	ActSpeedIsZero	During open-loop operation: 1 ≡ Speed setpoint < Comparison value (C00024) During closed-loop operation: 1 ≡ Actual speed value < Comparison value (C00024)
Bit 7	ControllerInhibit	1 ≡ Inverter is inhibited (controller inhibit is active)
Bit 8	StatusCodeBit0	Bit coded display of the active device status ▶ Device state machine and device states (see table [4-1])
Bit 9	StatusCodeBit1	
Bit 10	StatusCodeBit2	
Bit 11	StatusCodeBit3	
Bit 12	Warning	1 ≡ a warning is indicated
Bit 13	Trouble	1 ≡ Inverter is in the "Trouble" device status • E.g. if an overvoltage has occurred.
Bit 14	FreeStatusBit14	Free status bit 14 (not assigned, freely assignable)
Bit 15	FreeStatusBit15	Free status bit 15 (not assigned, freely assignable)

8.2.5 Setting parameters (short overview)

Parameters	Info	Lenze setting Value	Unit
C00012	Accel. time - main setpoint	2.000	s
C00013	Decel. time - main setpoint	2.000	s
C00019	Auto-DCB: Threshold	3	rpm
C00024	LS_DriveInterface: bNActCompare	0.00	%
C00036	DCB braking: Current	50.00	%
C00039/1	Preset setpoint 1	40.00	%
C00039/2	Preset setpoint 2	60.00	%
C00039/3	Preset setpoint 3	80.00	%
C00039/4...15	Fixed setpoint 4 ... 15	0.00	%
C00101/1...15	Add. accel. time 1 ... 15	0.000	s
C00103/1...15	Add. decel. time 1 ... 15	0.000	s
C00105	Decel. time - quick stop	2.000	s
C00106	Auto-DCB: Hold time	0.500	s
C00107	DCB braking: Hold time	999.000	s
C00134	L_NSet_1: Ramp smoothing	0: Off	
C00182	L_NSet_1: S-ramp time PT1	20.00	s
C00190	L_NSet_1: Setpoint arithmetic	0: Out = Set	
C00220	L_NSet_1: Acceleration time - add. setpoint	0.000	s
C00221	L_NSet_1: Deceleration time - add. setpoint	0.000	s
C00222	L_PCTRL_1: Vp	1.0	
C00223	L_PCTRL_1: Tn	400	ms
C00224	L_PCTRL_1: Kd	0.0	
C00225	L_PCTRL_1: MaxLimit	199.99	%
C00226	L_PCTRL_1: MinLimit	-199.99	%
C00227	L_PCTRL_1: Acceleration time	0.010	s
C00228	L_PCTRL_1: Deceleration time	0.010	s
C00233	L_PCTRL_1: Root function	0: Off	
C00241	L_NSet_1: Hyst. NSet reached	0.50	%
C00242	Operating mode of process controller	0: Off	
C00243	L_PCTRL_1: Accel. time influence	5.000	s
C00244	L_PCTRL_1: Deceleration time influence	5.000	s
C00632/1	L_NSet_1: Blocking speed 1 max	0.00	%
C00632/2	L_NSet_1: Blocking speed 2 max	0.00	%
C00632/3	L_NSet_1: Blocking speed 3 max	0.00	%
C00633/1	L_NSet_1: Blocking speed 1 min	0.00	%
C00633/2	L_NSet_1: Blocking speed 2 min	0.00	%
C00633/3	L_NSet_1: Blocking speed 3 min	0.00	%
C00635	L_NSet_1: nMaxLimit	199.99	%
C00636	L_NSet_1: nMinLimit	-199.99	%
C00670	L_OffsetGainP_1: Gain	1.0000	
C00671	L_OffsetGainP_2: Gain	1.0000	
C00672	L_OffsetGainP_3: Gain	1.0000	

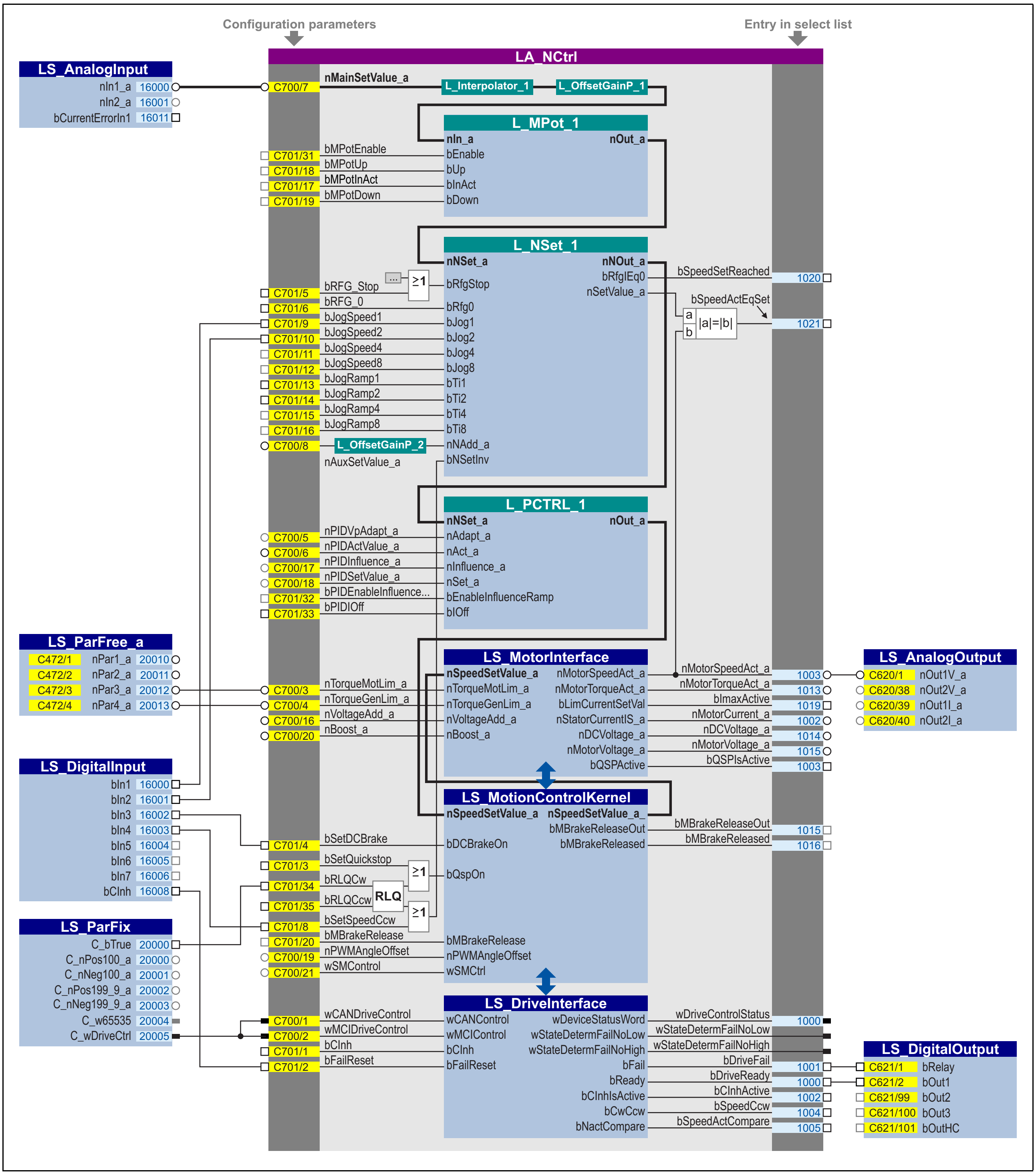
Parameters	Info	Lenze setting	
		Value	Unit
C00696	L_OffsetGainP_1: Offset	0.00	%
C00697	L_OffsetGainP_2: Offset	0.00	%
C00698	L_OffsetGainP_3: Offset	0.00	%
C00800	L_MPot_1: Upper limit	100.00	%
C00801	L_MPot_1: Lower limit	-100.00	%
C00802	L_MPot_1: Acceleration time	10.0	s
C00803	L_MPot_1: Deceleration time	10.0	s
C00804	L_MPot_1: Inactive fct.	0: Retain value	
C00805	L_MPot_1: Init fct.	0: Load last value	
C00806	Use of motor potentiometer	0: No	

Related topics:

- "GeneralPurpose" functions (📖 571)

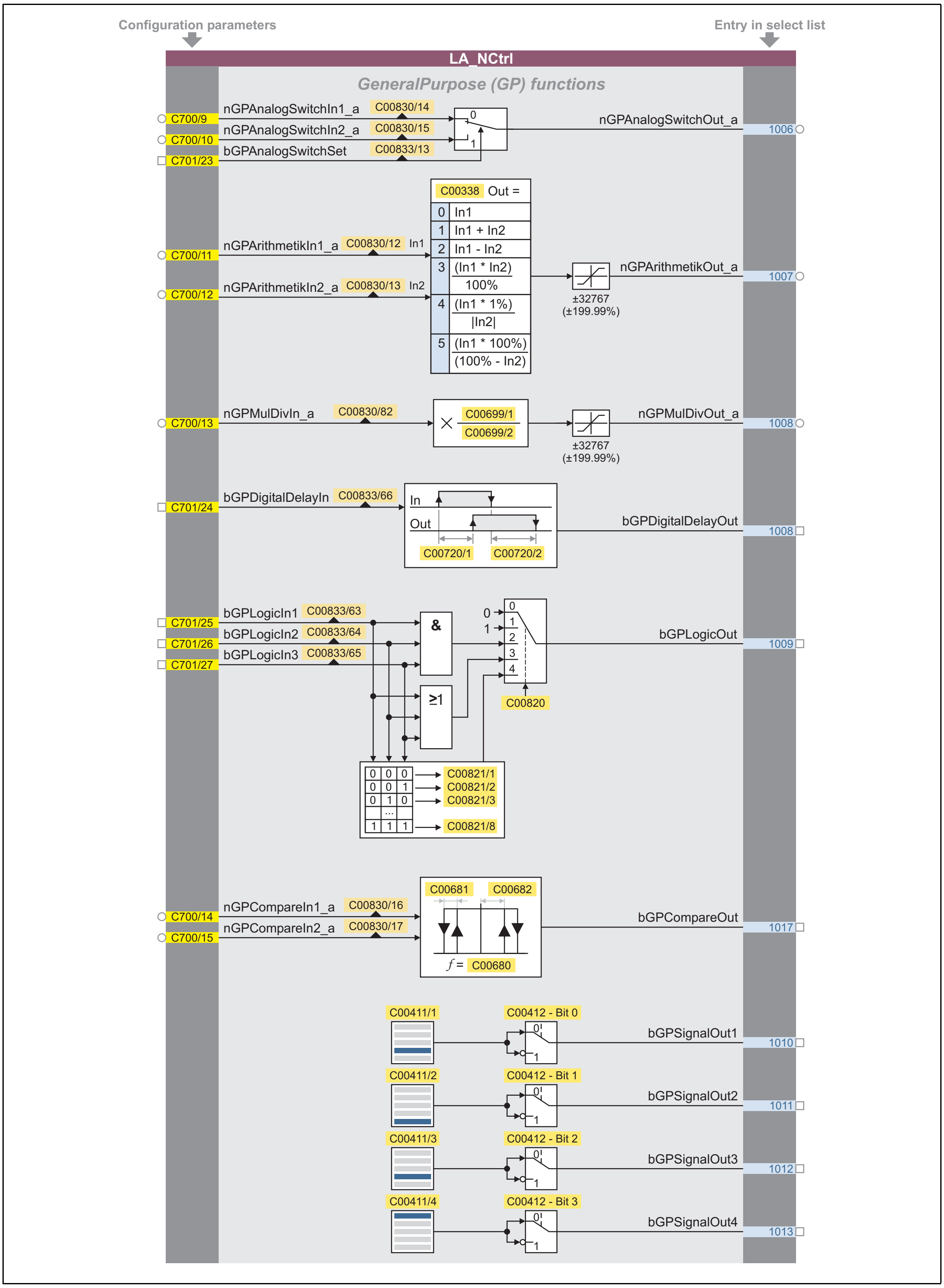
8.2.6 Configuration parameters

If required, the subcodes of C00700 and C00701 serve to change the pre-configured assignment of the application inputs:

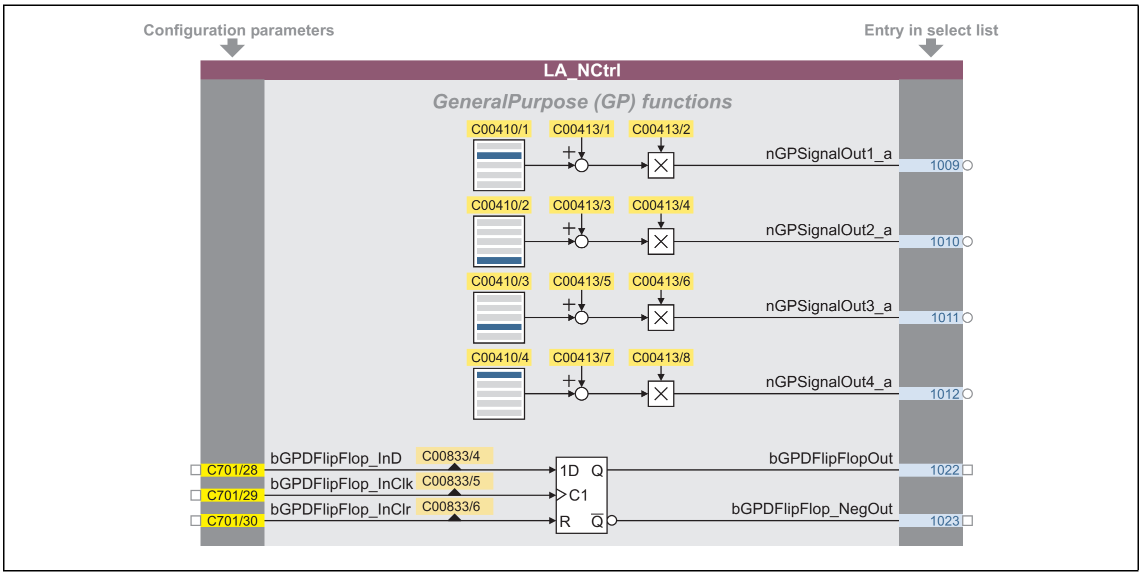


[8-2] Pre-assignment of the "Actuating drive speed" application in the "Terminals 0" control mode

Configuration parameters for "GeneralPurpose" functions



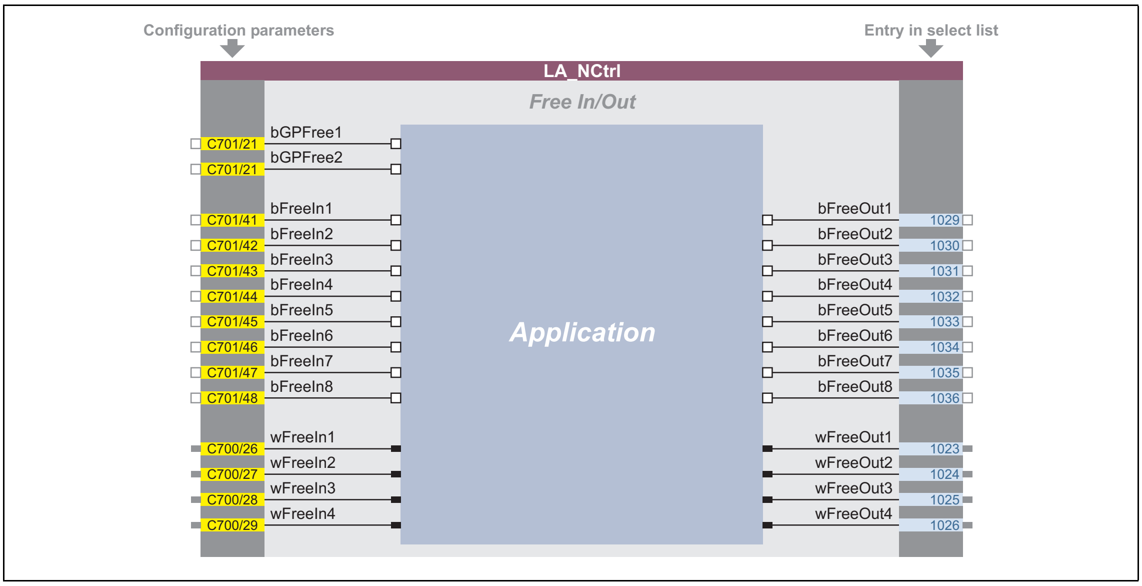
[8-3] "GeneralPurpose" functions



[8-4] "GeneralPurpose" functions (continuation)

Free inputs and outputs

These inputs can be freely interconnected in the application level. They can be used to transfer signals from the I/O level to the application level and vice versa.



[8-5] Free inputs/outputs

Related topics:

- User-defined terminal assignment (📖 445)
- "GeneralPurpose" functions (📖 571)

8.3 TA "actuating drive speed (AC Drive Profile)"

This technology application is available from version 13.00.00!

The EtherNet/IP™ communication module supports the "AC Drive Profile".

When the inverter is provided with an EtherNet/IP™ communication module and the control is to be carried out by means of "AC Drive Profile" via EtherNet/IP™, make the following settings:

1. Set the application "1100: Actuating drive speed (AC Drive Profile)" in C00005.
2. Set the "40: MCI" control mode in C00007.
   - The process data word received by the master control is then interpreted by the application as "AC Drive Profile" control word.
   - When the control is carried out via the system bus (CANopen), set the "30: CAN" control mode instead in C00007.

Product features

- Pre-configured control modes for terminals and bus control (with predefined process data connection to the fieldbus)
- Free configuration of input and output signals
- adjustable offset, gain and negation of the speed setpoint
- Up to 15 fixed setpoints for speed and ramp time
- Adjustable setpoint ramp times
- Freely selectable, variable ramp shape
- Automatic holding brake control
- Quick stop (QSP) with adjustable ramp time
- Motor potentiometer function (optional)
- Process controller (optional)
- Integrated, freely available "GeneralPurpose" functions:
  Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-flipflop
- Interface to the safety module (optional)
- Integration of encoder feedback

Note!

In contrast to the "actuating drive speed" standard application, this application makes use of the *nAuxSetValue_a* input to determine a local speed setpoint (when NetRef=0). For this reason, the *nNAdd_a* input at the L_NSet_1 setpoint generator for determining an additional speed setpoint is not connected in the Lenze setting.

8.3.1 I/O assemblies

For the data exchange, the technology application supports the assembly output object instance 23 (0x17) and assembly input object instance 73 (0x49) defined by the ODVA (Open DeviceNet Vendor Association).

Instance 23 (0x17): Extended Speed and Torque Control Output								
Bytes	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0		Net Ref	NetCtrl			Fault Reset	Run Rev	Run Fwd
1								
2	Speed Reference (Low Byte)							
3	Speed Reference (High Byte)							
4	Torque Reference (Low Byte)							
5	Torque Reference (High Byte)							

Instance 73 (0x49): Extended Speed and Torque Control Input								
Bytes	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
0	At Reference	RefFrom Net	CtrlFrom Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
1	Drive State							
2	Speed Actual (Low Byte)							
3	Speed Actual (High Byte)							
4	Torque Actual (Low Byte)							
5	Torque Actual (High Byte)							

Detailed information on the data transfer and "AC Drive Profile" can be found in the E84AYCEO communication manual (EtherNet/IP™).

Tip!

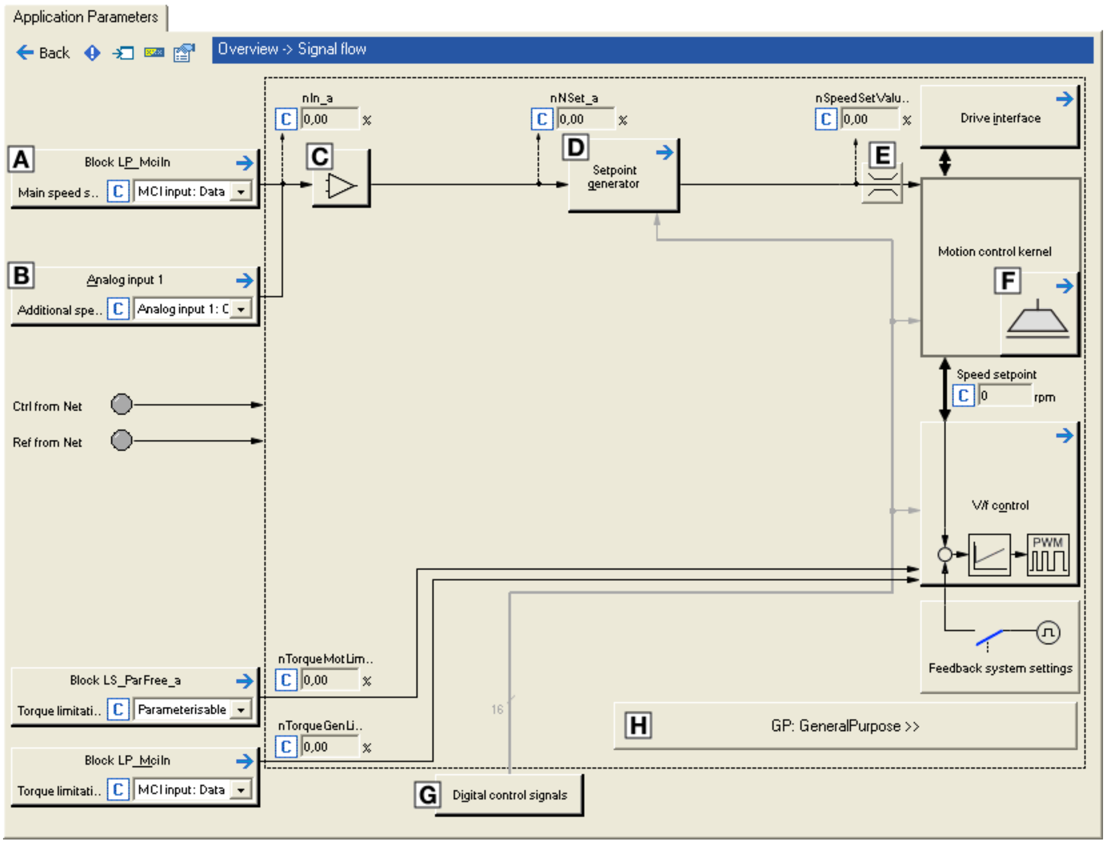
Detailed information on EtherNet/IP™ can be found at the web page of the ODVA (Open DeviceNet Vendor Association) user organisation:

http://www.odva.org

Related topics:

- Process data assignment for fieldbus communication
- Run/Stop event
- Scaling of speed and torque values

8.3.2 Basic signal flow



[8-6] Signal flow of the "actuating drive speed (AC Drive Profile)" technology application

A Configuration of the signal source for setpoint selection via fieldbus (NetRef=1)

B Configuration of the signal source for local setpoint selection (NetRef=0)

C Offset and gain for speed setpoint (L_OffsetGainP_1)

D Setpoint generator (L_NSet_1)

E Speed setpoint input limitation

F Holding brake control

G Terminal assignment & display of digital control signals

H Integrated disposable "GeneralPurpose" functions: Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-flipflop

Definition of the speed setpoint

The speed setpoint is usually defined as process date via the fieldbus. If set accordingly, the speed setpoint can also be determined locally (e.g. via the analog input 1). Depending on the selection, either only the *nMainSetValue_a* application input or only the application input *nAuxSetValue_a* is effective. The following table shows the relationships:

Fieldbus used	Control mode (C00007)	Speed reference	AC Drive Profile control word (*wMCIDriveControl*)	effective input (at LA_NCtrl)
EtherNet/IP	40: MCI	via fieldbus (data word 2)	Bit 6 ("NetRef") = 1	*nMainSetValue_a*
		local*	Bit 6 ("NetRef") = 0	*nAuxSetValue_a*
CANopen	30: CAN	via fieldbus (data word 2)	Bit 6 ("NetRef") = 1	*nMainSetValue_a*
		local*	Bit 6 ("NetRef") = 0	*nAuxSetValue_a*
-	10: Terminals 0 12: Terminals 2 14: Terminals 11 16: Terminals 16	local*	9 ≡ 0x0009 • Bit 0, SwitchOn = TRUE • Bit 3, EnableOperation = TRUE • All others: FALSE	*nAuxSetValue_a*
	20: Keypad	C00728/3		
	21: PC	C00472/1		

* The local setpoint is selected in the Lenze setting via the analog input 1

Scaling of the speed setpoint

If the setpoint is determined via fieldbus (NetRef=1), the setpoint applied to the *nMainSetValue_a* application input is evaluated with the speed scaling set in C01353/1 (AC Drive Attribute 22). ▶ Scaling of speed and torque values.

If the setpoint is determined locally (e.g. via the analog input 1), the setpoint applied to the *nAuxSetValue_a* application input is scaled as follows:

16384 ≡ 100 % reference speed (C00011).

Offset & gain

Offset and gain of the speed signal can be set in C00696 and C00670 for a simple signal adjustment of a setpoint encoder.

Ramp function generator

The setpoint is transformed to a speed setpoint in the setpoint encoder via a ramp function generator with linear or S-shaped ramps.

- Upstream to the ramp function generator, a blocking speed masking function and a setpoint MinMax limitation are effective.
- For a detailed functional description see the L_NSet FB.

Selection of the direction of rotation

- In case of control via fieldbus via bit 0 "Run Forward" and bit 1 "Run Backward" of the AC Drive Profile control word.
- In case of local control via the *bSetSpeedCcw* application input or by selecting negative fixed setpoints.

Selection of the torque setpoint in torque mode

In torque mode", the *nTorqueGenLim_a* application input has the function of the torque setpoint selection.

The torque setpoint is usually defined as process date via the fieldbus. If set accordingly, the torque setpoint can also be determined locally (e.g. via the analog input 1).

For the local selection of the torque setpoint, the same application input (*nAuxSetValue_a*) is used as for the local selection of the speed setpoint. In "torque mode" however, the *nAuxSetValue_a* input is internally connected to the *nTorqueGenLim_a* input. In this case, the speed setpoint is internally permanently set to "100 %".

The relationships are shown in the following table:

Fieldbus used	Control mode (C00007)	Torque reference	AC Drive Profile control word (*wMCIDriveControl*)	effective input (at LA_NCtrl)
EtherNet/IP	40: MCI	via fieldbus (data word 3)	Bit 6 ("NetRef") = 1	*nTorqueGenLim_a*
		local*	Bit 6 ("NetRef") = 0	*nAuxSetValue_a*
CANopen	30: CAN	via fieldbus (data word 3)	Bit 6 ("NetRef") = 1	*nTorqueGenLim_a*
		local*	Bit 6 ("NetRef") = 0	*nAuxSetValue_a*
-	10: Terminals 0 12: Terminals 2 14: Terminals 11 16: Terminals 16	local*	9 ≡ 0x0009 • Bit 0, SwitchOn = TRUE • Bit 3, EnableOperation = TRUE • All others: FALSE	*nAuxSetValue_a*
	20: Keypad	C00728/3		
	21: PC	C00472/1		

* The local setpoint is selected in the Lenze setting via the analog input 1

8.3.3 Internal interfaces | application block "LA_NCtrl"

Note!

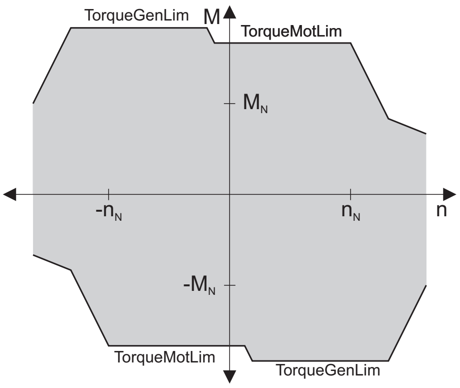
The connectors greyed out in the following table are hidden in the function block editor in the Lenze setting.

- These connections can be shown via the **Connector visibilities** command in the *Context menu* of the application block.

inputs

Designator Data type	Information/possible settings	
wCANDriveControl WORD	Input for CAN control word • Is not used in this configuration and is thus set to the permanent value "9" in the Lenze setting (SwitchOn = TRUE and EnableOperation = TRUE). • For the evaluation of the AC Drive Profile control word received via fieldbus, the *wMCIDriveControl* input is used in the control modes 30: CAN" and "40: MCI".	
wMCIDriveControl WORD	Input for the AC Drive Profile control word received via fieldbus • The AC Drive Profile control word operates the inverter in compliance with the assembly output object instances 20 ... 23. For this purpose, the control bits are evaluated and lead to a corresponding modification of the control signals *bFailReset*, *bRFG_0* and *bSetSpeedCcw* which results in the AC Drive-specific behaviour. • See the "Process data assignment for fieldbus communication" subchapter for a detailed description of the individual control bits. • Display parameter: C01351/1	
wSMControl WORD	Interface to the optional safety system. • Setting control bit 0 ("SafeStop1") in this control word causes e.g. the automatic deceleration of the drive to standstill within this application (in the **Motion Control Kernel**). • See the subchapter "Interface to safety system" of the chapter on basic drive functions for a detailed description of the individual control bits.	
bCInh BOOL	Enable/inhibit inverter	
	FALSE	Enable inverter: The inverter switches to the "OperationEnabled" device status if no other source for controller inhibit is active. • C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.
	TRUE	Inhibit inverter (controller inhibit): The inverter switches to the "SwitchedOn" device status.
bFailReset BOOL	Reset error message • In the Lenze setting this input is connected to the digital input controller enable so that a possibly existing error message is reset together with the controller enable (if the cause for the fault is eliminated). • **In case of control via fieldbus (NetCtrl=1):** This input is OR'd with bit 2 ("fault reset") of the AC Drive Profile control word.	
	TRUE	The current fault is reset, if the cause for the fault is eliminated. • If the fault still exists, the error status remains unchanged.

Designator (Data type)		Information/possible settings
bSetQuickstop (BOOL)		Activate quick stop (QSP) • Also see device command "Activate/deactivate quick stop".
	TRUE	Activate quick stop • Motor control is decoupled from the setpoint selection and, within the deceleration time parameterised in C00105, the motor is brought to a standstill (n_{act} = 0). • The motor is kept at a standstill during closed-loop operation. • A pulse inhibit is set if the auto-DCB function has been activated via C00019.
	FALSE	Deactivate quick stop • The quick stop is deactivated if no other source for the quick stop is active. • C00159 displays a bit code of active sources/causes for the quick stop.
bSetDCBrake (BOOL)		Manual DC-injection braking (DCB) • Detailed information on DC-injection braking is provided in the motor control chapter, subchapter "DC-injection braking".
		Note! Holding braking is not possible when this braking mode is used! Use the basic "Holding brake control" function for controlling the holding brake with a low rate of wear.
	FALSE	Deactivate DC-injection braking.
	TRUE	Activate DC-injection braking, i.e. the drive is brought to a standstill by means of DC-injection braking. • The braking effect stops when the rotor is at standstill. • After the hold time (C00107) has expired, the controller sets the pulse inhibit.
bRFG_Stop (BOOL)		Ramp function generator: Maintain the current value of the main setpoint integrator • The speed, for instance, of a running ramp process is immediately kept constant when *bRFG_Stop* is activated. At the same time, the acceleration/deceleration jumps to the value "0". • For a detailed functional description see the L_NSet FB.
	TRUE	The current value of the main setpoint integrator is held.
bRFG_0 (BOOL)		Ramp function generator: Lead the main setpoint integrator to "0" within the current Ti times • **In case of control via fieldbus (NetCtrl=1):** If a stop is triggered via bit 0 ("Run Forward") and bit 1 ("Run Backward") of the AC Drive Profile control word, this signal is internally set to TRUE and thus the drive is braked to standstill. • For a detailed functional description see the L_NSet FB.
	TRUE	The current value of the main setpoint integrator is led to "0" within the Ti time set.
nVoltageAdd_a (INT)		Additive voltage impression • An additional setpoint for the motor voltage can be specified via this process input. • If there are, for instance, different loads at the motor output end, it is possible to apply a voltage boost at the starting time. • If the value is negative, the voltage is reduced. • Scaling: 16384 ≡ 1000 V
		Stop! Values selected too high may cause the motor to heat up due to the resulting current!

Designator Data type	Information/possible settings	
nBoost_a INT	Additional setpoint for the motor voltage at speed = 0 • The entire voltage-frequency characteristic is provided with an offset. • Scaling: 16384 ≡ 1000 V	
	Stop! Values selected too high may cause the motor to heat up due to the resulting current!	
nPWMAngleOffset INT	Additional offset for the electrical angle of rotation • If a torque is connected, e.g. dynamic acceleration processes can be generated. • Scaling: ±32767 ≡ ±180 ° angle of rotation	
nTorqueMotLim_a nTorqueGenLim_a INT	Torque limitation in motor mode and in generator mode (speed mode) or torque setpoint selection (torque mode)	
	When "speed mode" is set, the following applies: • The torque limitation in motor mode and generator mode are determined via *nTorqueMotLim_a*. • The *nTorqueGenLim_a* input is not effective.	
	When "torque mode" is set, the following applies: • The torque setpoint is determined via *nTorqueGenLim_a*. • The *nTorqueMotLim_a* input is not effective.	
	More notes for both inputs: • These input signals are directly transferred to the motor control to limit the inverter's maximum torque in motor and generator mode. • The drive cannot output a higher torque in motor/generator mode than set here. • The applied values (any polarity) are internally interpreted as absolute values. • If V/f characteristic control (VFCplus) is selected, limitation is indirectly performed via a so-called I_{max} controller. • If sensorless vector control (SLVC) or servo control (SC) is selected, limitation has a direct effect on the torque-producing current component. • Scaling: 16384 ≡ 100 % M_{max} (C00057)	
	Torque limits in motor and generator mode: 	
bSetSpeedCcw BOOL	Change of direction of rotation • For instance if a motor or gearbox is fixed laterally reversed to a machine part, but the setpoint selection should still be executed for the positive direction of rotation. • **Input only effective in case of local control (NetCtrl=0)**	
	FALSE	Clockwise rotation (Cw)
	TRUE	Direction of rotation to the left (Ccw)
bRLQCw BOOL	Activate clockwise rotation (fail-safe) • For a detailed functional description see the L_RLQ FB.	
	FALSE	Quick stop
	TRUE	CW rotation

Designator	Data type	Information/possible settings
bRLQCcw	BOOL	Activate counter-clockwise rotation (fail-safe) • **Input only effective in case of local control (NetCtrl=0)** • For a detailed functional description see the L_RLQ FB.
		FALSE: Quick stop
		TRUE: CCW rotation
nMainSetValue_a	INT	Speed setpoint via fieldbus in [rpm] • **Input only effective in case of setpoint selection via fieldbus (NetRef=1)** • This input is evaluated with the speed scaling set in C01353/1 (AC Drive Attribute 22). ▸ Scaling of speed and torque values • An absolute value is created internally (the sign has no meaning). • Offset and gain of this input signal can be set in C00696 and C00670 for a simple signal adjustment of a setpoint encoder. • The setpoint is transformed to a speed setpoint in the setpoint encoder via a ramp function generator with linear or S-shaped ramps. • Upstream to the ramp function generator, a blocking speed masking function and a setpoint MinMax limitation are effective. • For a detailed functional description see the L_NSet FB.
nAuxSetValue_a	INT	Local speed setpoint (speed mode) or torque setpoint (torque mode) • **Input only effective in case of local setpoint selection (NetRef=0)**
		When "speed mode" is set, the following applies: • The input value is interpreted as speed setpoint. • Scaling: 16384 ≡ 100 % reference speed (C00011) • Offset and gain of this input signal can be set in C00696 and C00670 for a simple signal adjustment of a setpoint encoder. • The setpoint is transformed to a speed setpoint in the setpoint encoder via a ramp function generator with linear or S-shaped ramps. • Upstream to the ramp function generator, a blocking speed masking function and a setpoint MinMax limitation are effective. • For a detailed functional description see the L_NSet FB.
		When "torque mode" is set, the following applies: • The input value is interpreted as torque setpoint. (Input *nAuxSetValue_a* is internally connected to input *nTorqueGenLim_a*). • Scaling: 16384 ≡ 100 % M_{max} (C00057) • The speed setpoint is internally permanently set to "100 %".
bJogSpeed1 bJogSpeed2	BOOL	Selection inputs for override fixed setpoints (JOG setpoints) • **Inputs only effective in case of local setpoint selection (NetRef=0)** • These selection inputs can be used to activate a fixed setpoint for the setpoint generator instead of the setpoint applied to the *nAuxSetValue_a* input. • The four selection inputs are binary coded, therefore 15 fixed setpoints can be selected. • In case of binary coded selection "0" (all inputs = FALSE or not assigned), the main setpoint applied to the *nAuxSetValue_a* input is active. • The selection of the fixed setpoints is carried out in C00039/1...15 in [%] based on the reference speed (C00011). • For a detailed functional description see the L_NSet FB.
bJogSpeed4 bJogSpeed8	BOOL	
bJogRamp1 bJogRamp2	BOOL	Selection inputs for alternative acceleration/deceleration times • The four selection inputs are binary coded, therefore 15 alternative acceleration/deceleration times can be selected. • In case of binary selection "0" (all inputs = FALSE or not assigned), the acceleration time (C00012) and deceleration time (C00013) set for the main setpoint are active. • Alternative acceleration times are selected in C00101/1...15. • The selection of the alternative deceleration times is carried out in C00103/1...15. • For a detailed functional description see the L_NSet FB.
bJogRamp4 bJogRamp8	BOOL	

Designator Data type	Information/possible settings	
Motor potentiometer Alternatively to the input signal *nMainSetValue_a* (or *nAuxSetValue_a* in case of local setpoint selection), the speed setpoint can also be generated via a motor potentiometer function. • In the Lenze setting, the motor potentiometer function is deactivated. • Activation is possible via C00806 or via the *bMPotEnable* input. • The behaviour of the motor potentiometer during switch-on of the drive system can be selected in C00805. • For a detailed functional description see the L_MPot FB.		
bMPotEnable BOOL	Activating the motor potentiometer function • This input and C00806 are OR'd.	
	TRUE	The motor potentiometer function is active; the speed setpoint can be changed via the *bMPotUp* and *bMPotDown* control inputs.
bMPotUp BOOL	Increasing the speed setpoint	
	TRUE	Approach the upper speed limit value set in C00800 with the acceleration time set in C00802.
bMPotInAct BOOL	Activating the inactive function	
	TRUE	The speed setpoint behaves according to the inactive function set in C00804. • In the Lenze setting, the speed setpoint is maintained.
bMPotDown BOOL	Decreasing the speed setpoint	
	TRUE	Approach the lower speed limit value set In C00801 with the deceleration time set in C00803.
Process controller • In the Lenze setting, the process controller is deactivated. • The activation is executed by selecting the operating mode in C00242. • For a detailed functional description see FB L_PCTRL.		
bPIDEnableInfluenceRamp BOOL	Activate ramp for influencing factor	
	FALSE	Influencing factor of the PID controller is ramped down to "0".
	TRUE	Influencing factor of the PID controller is ramped up to the value *nPIDInfluence_a*.
bPIDIOff BOOL	Switch off the I-component of the process controller • In conjunction with the operating mode set in C00242 (Lenze setting: "Off").	
	TRUE	I-component of the process controller is switched off.
nPIDVpAdapt_a INT	Adaptation of gain Vp set in C00222 in percent • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 % • Changes can be done online.	
nPIDSetValue_a INT	Sensor and process setpoint for operating modes 2, 4 and 5 • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 %	
nPIDActValue_a INT	Speed or actual sensor value (actual process value) • Offset and gain for this input signal can be set in C00698 and C00672. • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 %	
nPIDInfluence_a INT	Limitation of the influencing factor in percent • The influence factor of the PID controller can be limited to a certain value (-199.99% ... + 199.99%) via *nPIDInfluence_a*. • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 %	

Designator Data type	Information/possible settings
MCK basic functions	
bMBrakeRelease BOOL	Holding brake control: Release/apply brake • In conjunction with the operating mode selected in C02580 (Lenze setting: "Brake control off"). FALSE: Apply brake. • During automatic operation, the internal brake logic controls the brake. TRUE: Release brake manually (forced release). • **Note!** The brake can also be released when the controller is inhibited! • During automatic operation, the internal brake logic is deactivated and the brake is released (supervisor operation). If a controller inhibit has been set by the brake control, it will be deactivated. • In semi-automatic operation, the brake is released including feedforward control.
GP: GeneralPurpose The following inputs are interconnected with logic/arithmetic functions on application level for free usage. ▸ "GeneralPurpose" functions	
bGPFree1 ... bGPFree2 BOOL	Free inputs for digital signals • Digital signals can be transferred from the I/O level to the application level via these inputs.
nGPAnalogSwitchIn1_a nGPAnalogSwitchIn2_a INT	Analog switch: Input signals • The input signal selected via the selection input *bGPAnalogSwitchSet* is output at output *nGPAnalogSwitchOut_a*.
bGPAnalogSwitchSet BOOL	Analog switch: Selection input FALSE: *nGPAnalogSwitchOut_a = nGPAnalogSwitchIn1_a* TRUE: *nGPAnalogSwitchOut_a = nGPAnalogSwitchIn2_a*
nGPArithmetikIn1_a nGPArithmetikIn2_a INT	Arithmetic: Input signals • The arithmetic function is selected in C00338. • The result is output at output *nGPArithmetikOut_a*.
nGPMulDivIn_a INT	Multiplication/Division: Input signal • The factor for the multiplication can be set in C00699/1 (numerator) and C00699/2 (denominator). • The result is output at output *nGPMulDivOut_a*.
bGPDigitalDelayIn BOOL	Binary delay element: Input signal • The on-delay can be set in C00720/1. • The off-delay can be set in C00720/2. • The time-delayed input signal is output at output *bGPDigitalDelayOut*.
bGPLogicIn1 bGPLogicIn2 bGPLogicIn3 BOOL	Binary logic: Input signals • The logic operation is selected in C00820. • The result is output at output *bGPLogicOut*.
nGPCompareIn1_a nGPCompareIn2_a INT	Analog comparison: Input signals • The comparison operation is selected in C00680. • Hysteresis and window size can be set in C00680 and C00682. • If the comparison statement is true, the output *bGPCompareOut* will be set to TRUE.
bGPDFlipFlop_InD bGPDFlipFlop_InClk bGPDFlipFlop_InClr BOOL	D-FlipFlop: Input signals • Data, clock and reset input

Designator Data type	Information/possible settings
Free inputs The following inputs can freely be interconnected on the application level. The signals can be transferred from the I/O level to the application level via these inputs.	
bFreeIn1 ... bFreeIn8 BOOL	Free inputs for digital signals
wFreeIn1 ... wFreeIn4 WORD	Free inputs for 16-bit signals

outputs

Designator Data type	Value/meaning	
wDriveControlStatus WORD	AC Drive Profile status word • The status word contains information on the currents status of the inverter. • For a detailed description of the individual status bits, see subchapter entitled "Process data assignment for fieldbus communication. • Display parameter: C01352/1	
wStateDetermFailNoLow WORD	Display of the status determining error (LOW word)	
wStateDetermFailNoHigh WORD	Display of the status determining error (HIGH word)	
bDriveFail BOOL	TRUE	Inverter in error status. • "Fault" device status is active.
bDriveReady BOOL	TRUE	Inverter is ready for operation • "SwitchedOn" device status is active. • The drive is in this device status if the DC bus voltage is applied and the inverter is still inhibited by the user (controller inhibit).
bCInhActive BOOL	TRUE	Controller inhibit is active.
bQSPIsActive BOOL	TRUE	Quick stop is active.
bSpeedCcw BOOL	Current direction of rotation	
	FALSE	Clockwise rotation (Cw)
	TRUE	Direction of rotation to the left (Ccw)
bSpeedActCompare BOOL	Result of the speed comparison (detection of speed=0)	
	TRUE	During open-loop operation: Speed setpoint < Comparison value (C00024)
		During closed-loop operation: Actual speed value < Comparison value (C00024)
bOverLoadActive BOOL	In preparation (output is not interconnected on the application level)	
bUnderLoadActive BOOL	In preparation (output is not interconnected on the application level)	
bImaxActive BOOL	"Current setpoint inside the limitation" status signal	
	TRUE	The current setpoint is internally limited (the inverter operates at the maximum current limit).
bSpeedSetReached BOOL	Status signal "setpoint = 0"	
	TRUE	Speed setpoint from the ramp function generator = 0
bSpeedActEqSet BOOL	TRUE	Actual speed value = speed setpoint
nMotorCurrent_a INT	Current stator current/effective motor current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	

Designator Data type	Value/meaning
nMotorSpeedSet_a INT	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)
nMotorSpeedAct_a INT	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011) • The absolute value is output (the sign has no meaning).
nMotorTorqueAct_a INT	Actual torque • In the "VFC (+encoder)" operating mode of the motor control, this value is determined from the current motor current and corresponds to the actual torque only by approximation. • Scaling: 16384 ≡ 100 % M_{max} (C00057) • The absolute value is output (the sign has no meaning).
nDCVoltage_a INT	Current DC-bus voltage • Scaling: 16384 ≡ 1000 V
nMotorVoltage_a INT	Current motor voltage/inverter output voltage • Scaling: 16384 ≡ 1000 V
MCK basic functions	
bMBrakeReleaseOut BOOL	Holding brake control: Trigger signal for the holding brake control switching element via a digital output • Use bit 0 in C02582 to activate inverted switching element triggering. FALSE: Apply brake. TRUE: Release brake.
bMBrakeReleased BOOL	Holding brake control: "Brake released" considering the brake release time • When the holding brake is triggered to close, *bMBrakeReleased* is immediately set to FALSE even if the brake closing time has not yet elapsed! TRUE: Brake released (after the brake release time has expired).
GP: GeneralPurpose The following outputs are interconnected with logic/arithmetic functions on application level for free usage. ▸ "GeneralPurpose" functions	
nGPAnalogSwitchInOut_a INT	Analog switch: Output signal
nGPArithmetikOut_a INT	Arithmetic: Output signal
nGPMulDivOut_a INT	Multiplication/Division: Output signal
bGPDigitalDelayOut BOOL	Binary delay element: Output signal
bGPLogicOut BOOL	Binary logic: Output signal
bGPCompareOut BOOL	Analog comparison: Output signal
bGPSignalOut1 ... bGPSignalOut4 BOOL	Binary signal monitor: Output signals • The signal sources to be output are selected in C00411/1...4. • A bit coded inversion of the output signals can be parameterised in C00412.
nGPSignalOut1_a ... nGPSignalOut4_a BOOL	Analog signal monitor: Output signals • The signal sources to be output are selected in C00410/1...4. • Gain and offset for each output signal can be parameterised in C00413/1...8.
bGPDFlipFlop_Out BOOL	D-FlipFlop: Output signal
bGPDFlipFlop_NegOut BOOL	D-FlipFlop: Negated output signal

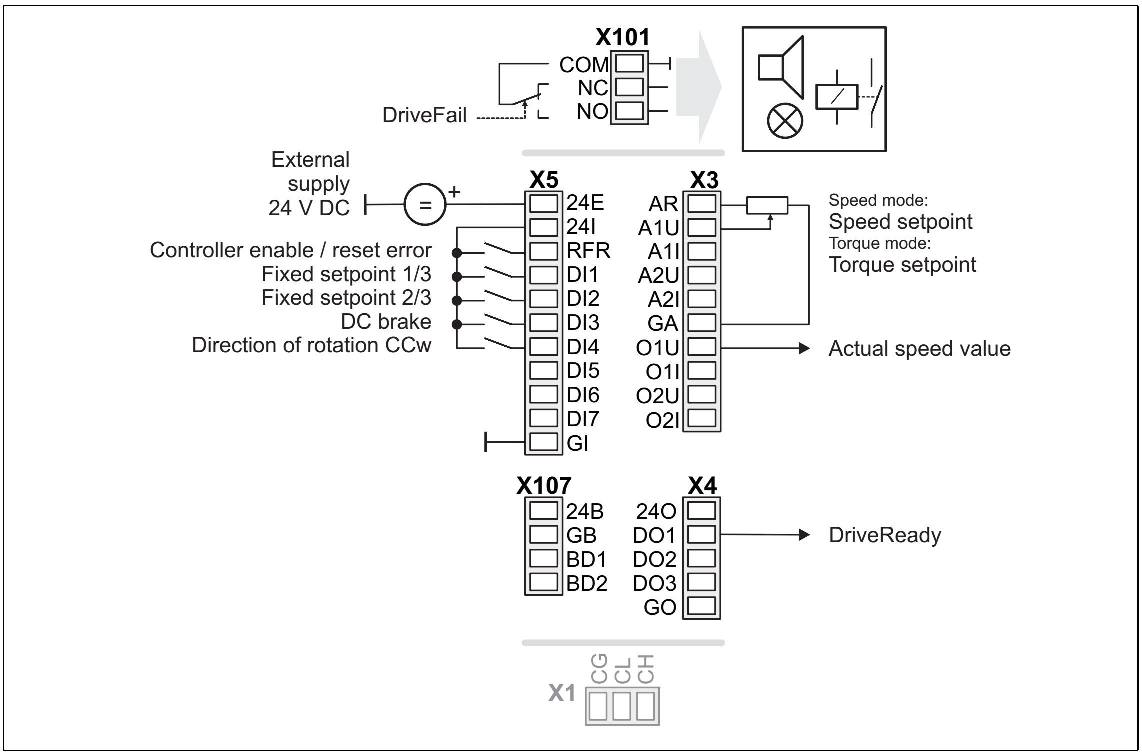
Designator Data type	Value/meaning
Free outputs The following outputs can freely be interconnected on the application level. The signals from the application level can be transferred to the I/O level via these outputs.	
bFreeOut1 ... bFreeOut8 BOOL	Free outputs for digital signals
wFreeOut1 ... wFreeOut4 WORD	Free outputs for 16-bit signals

8.3.4 Terminal assignment of the control modes

The following comparison provides information about which inputs/outputs of the application block **LA_NCtrl** are interconnected to the digital and analog input/output terminals of the inverter in the different control modes.

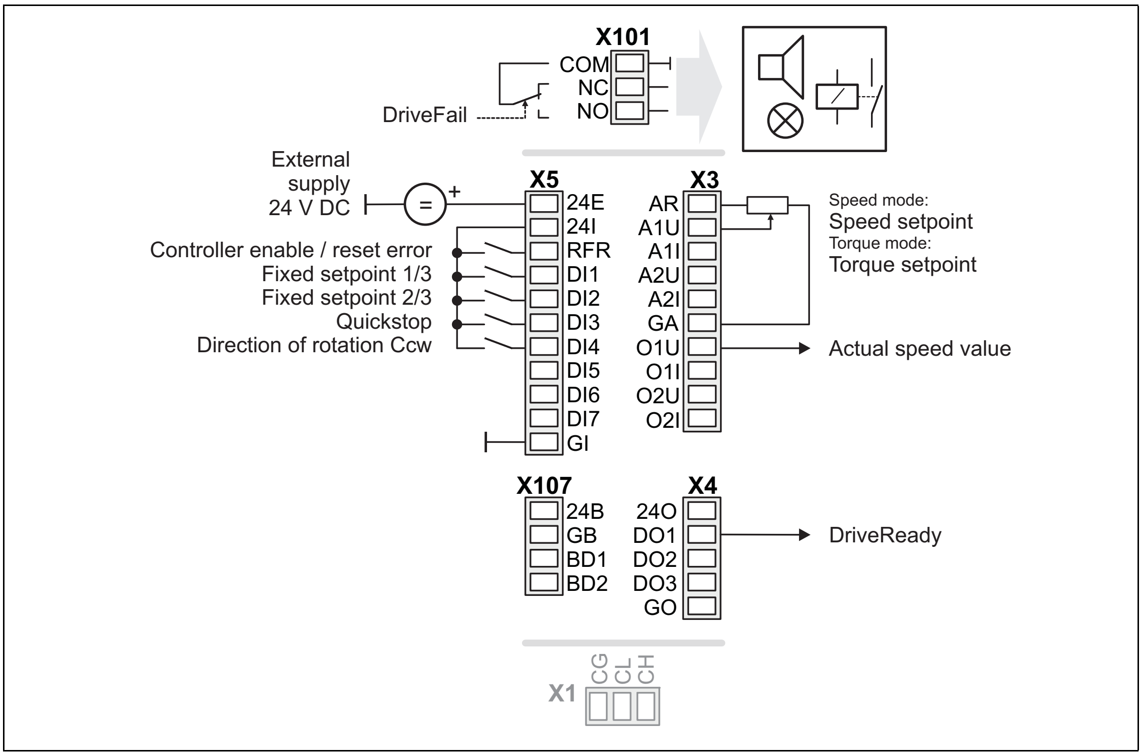
	Control mode (C00007)							
	10: Terminals 0	12: Terminals 2	14: Terminals 11	16: Terminal 16	20: Keypad	21: PC	30: CAN	40: MCI
Digital input terminals								
X5/RFR	**Controller enable / Reset of error message** bFailReset							
X5/DI1	**Fixed setpoint 1/3** bJogSpeed1		**Change of direction of rotation** bSetSpeedCcw	**Fixed setpoint 1/3** bJogSpeed1	-	-	Only if NetCtrl=0: **fixed setpoint 1/3** bJogSpeed1	
X5/DI2	**Fixed setpoint 2/3** bJogSpeed2		**Activate manual DC-injection braking (DCB)** bSetDCBrake	**Fixed setpoint 2/3** bJogSpeed2	-	-	Only if NetCtrl=0: **fixed setpoint 2/3** bJogSpeed2	
X5/DI3	**Activate manual DC-injection braking (DCB)** bSetDCBrake	**Quick stop** bSetQuickstop	**Motor potentiometer: Increase speed** bMPotUp	**CW rotation quick stop** bRLQCw	-	-	Only if NetCtrl=0: **Activate manual DC-injection braking (DCB)** bSetDCBrake	
X5/DI4	**Change of direction of rotation** bSetSpeedCcw		**Motor potentiometer: Decrease speed** bMPotDown	**CCW rotation quick stop** bRLQCcw	-	-	Only if NetCtrl=0: **change of direction of rotation** bSetSpeedCcw	
X5/DI5 ... DI7	-	-	-	-	-	-	-	-
Analog input terminals								
X3/A1U, A1I	**Local setpoint** nAuxSetValue_a Speed mode: 10 V ≡ 100 % reference speed (C00011) Torque mode: 10 V ≡ 100 % M_{max} (C00057); speed setpoint = 100 % (fixed)				-	-	Only if NetRef=0: **Local setpoint** nAuxSetValue_a Speed mode: 10 V ≡ 100 % reference speed (C00011) Torque mode: 10 V ≡ 100 % M_{max} (C00057)	
X3/A2U, A2I	-	-	-	-	-	-	-	-
Digital output terminals								
X4/DO1	**Status "Drive is ready"** bDriveReady							
X4/DO2 ... DO3	-	-	-	-	-	-	-	-
X107/BD1, BD2	-	-	-	-	-	-	-	-
X101/COM, NO	**Status "Error is pending"** bDriveFail							
Analog output terminals								
X3/O1U	**Actual speed value** nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)							
X3/O1I **X3/O2U** **X3/O2I**	-	-	-	-	-	-	-	-

8.3.4.1 Terminals 0



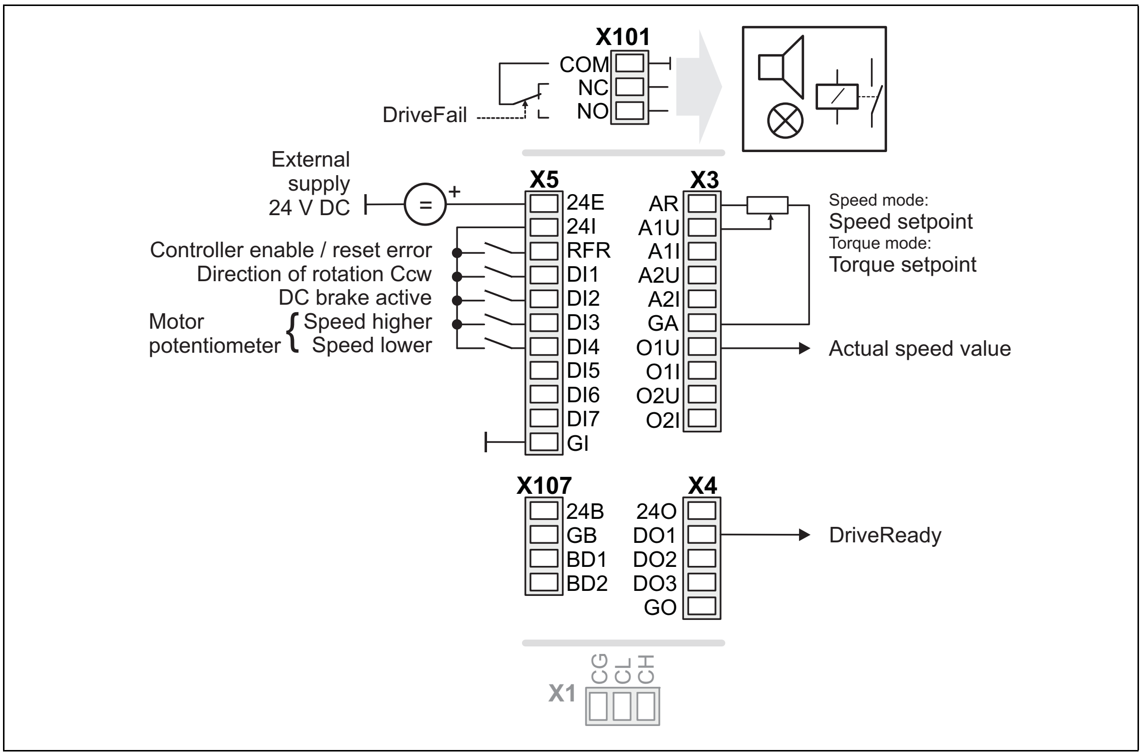
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl.bSetDCBrake		**X3/A2I**	-
X5/DI4	LA_NCtrl.bSetSpeedCcw		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.3.4.2 Terminals 2



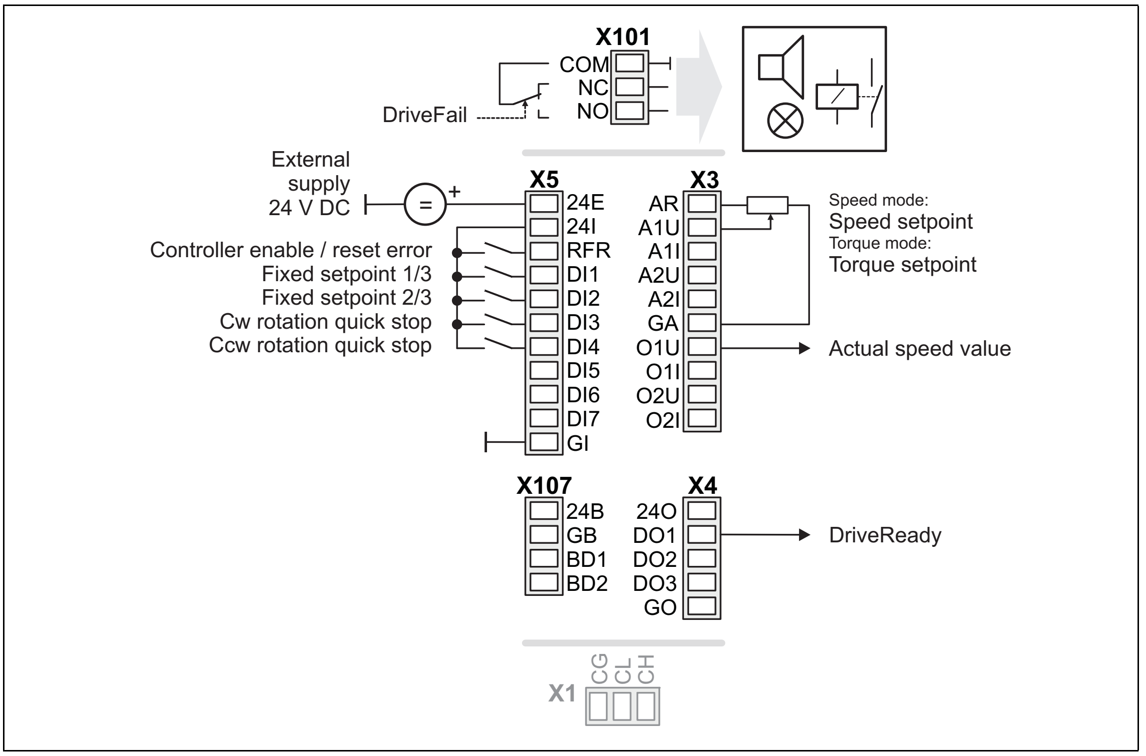
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl.bSetQuickstop		**X3/A2I**	-
X5/DI4	LA_NCtrl.bSetSpeedCcw		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.3.4.3 Terminals 11



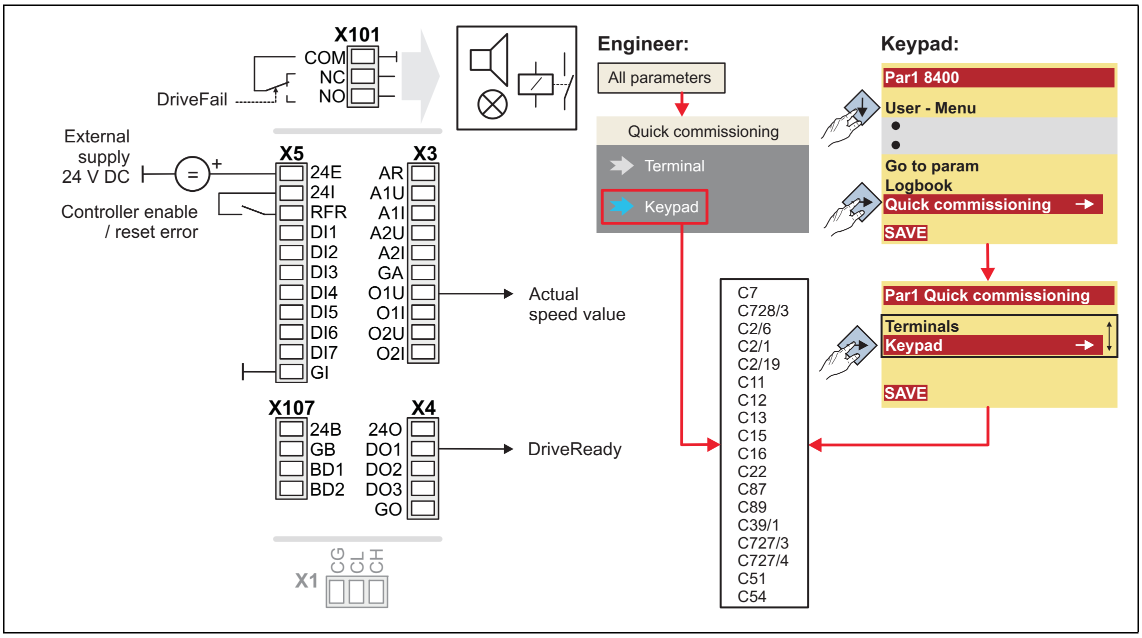
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bSetSpeedCcw		**X3/A1I**	-
X5/DI2	LA_NCtrl.bSetDCBrake		**X3/A2U**	-
X5/DI3	LA_NCtrl.bMPotUp		**X3/A2I**	-
X5/DI4	LA_NCtrl.bMPotDown		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.3.4.4 Terminal 16



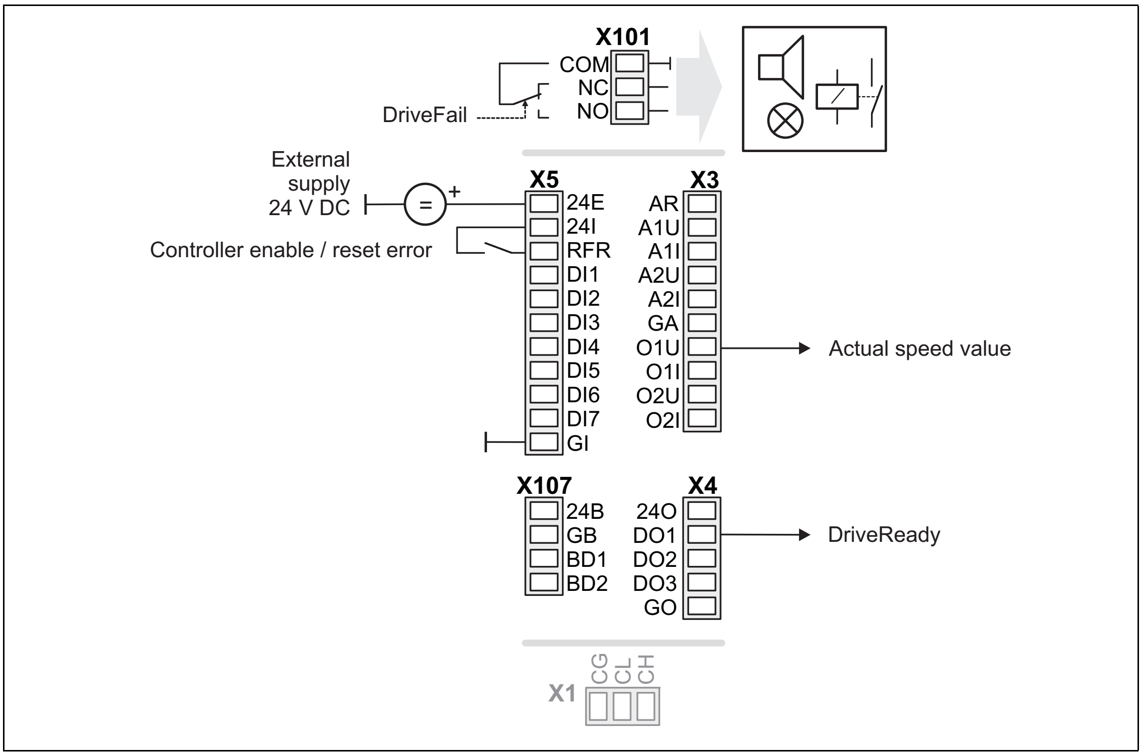
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl. bRLQCw		**X3/A2I**	-
X5/DI4	LA_NCtrl. bRLQCcw		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.3.4.5 Keypad



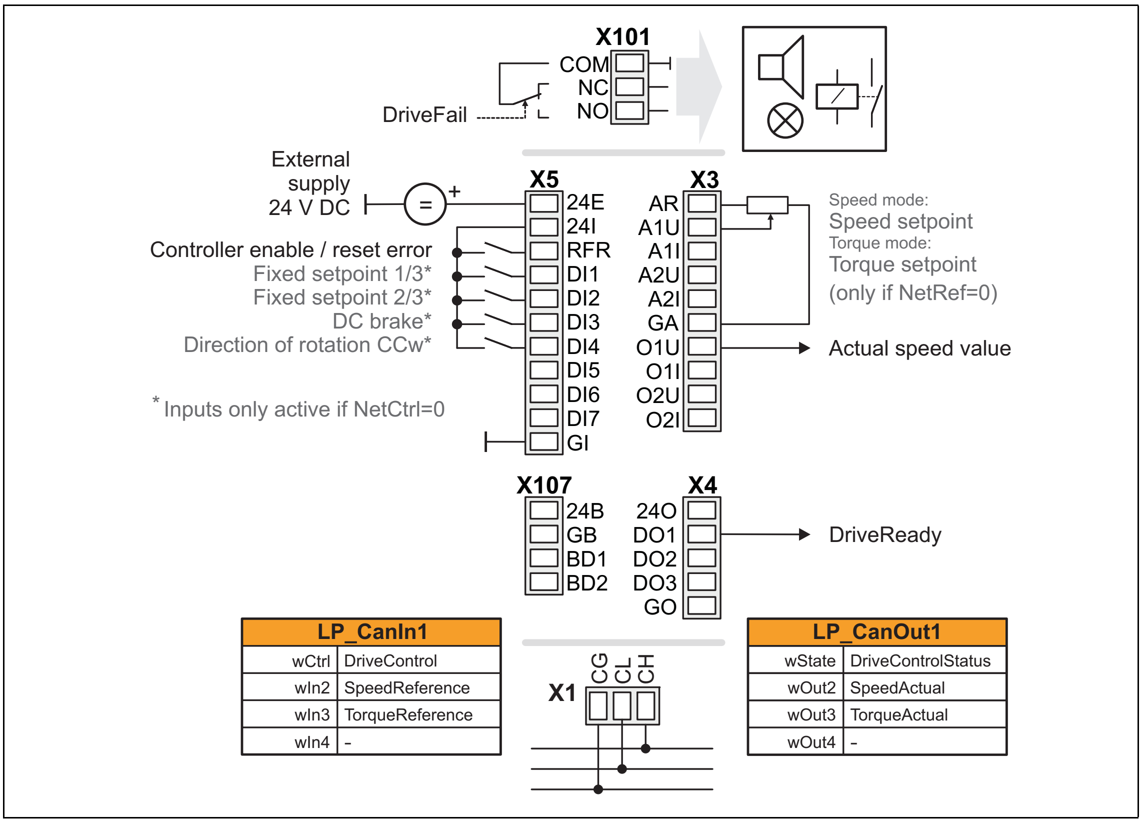
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		X3/A1U	-
X5/DI1	-		X3/A1I	-
X5/DI2	-		X3/A2U	-
X5/DI3	-		X3/A2I	-
X5/DI4	-		X3/O1U	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		X3/O1I	-
X5/DI6	-		X3/O2U	-
X5/DI7	-		X3/O2I	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		X4/DO1	LA_NCtrl.bDriveReady
X107/BD2	-		X4/DO2	-
			X4/DO3	-

8.3.4.6 PC



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	-
X5/DI1	-		**X3/A1I**	-
X5/DI2	-		**X3/A2U**	-
X5/DI3	-		**X3/A2I**	-
X5/DI4	-		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

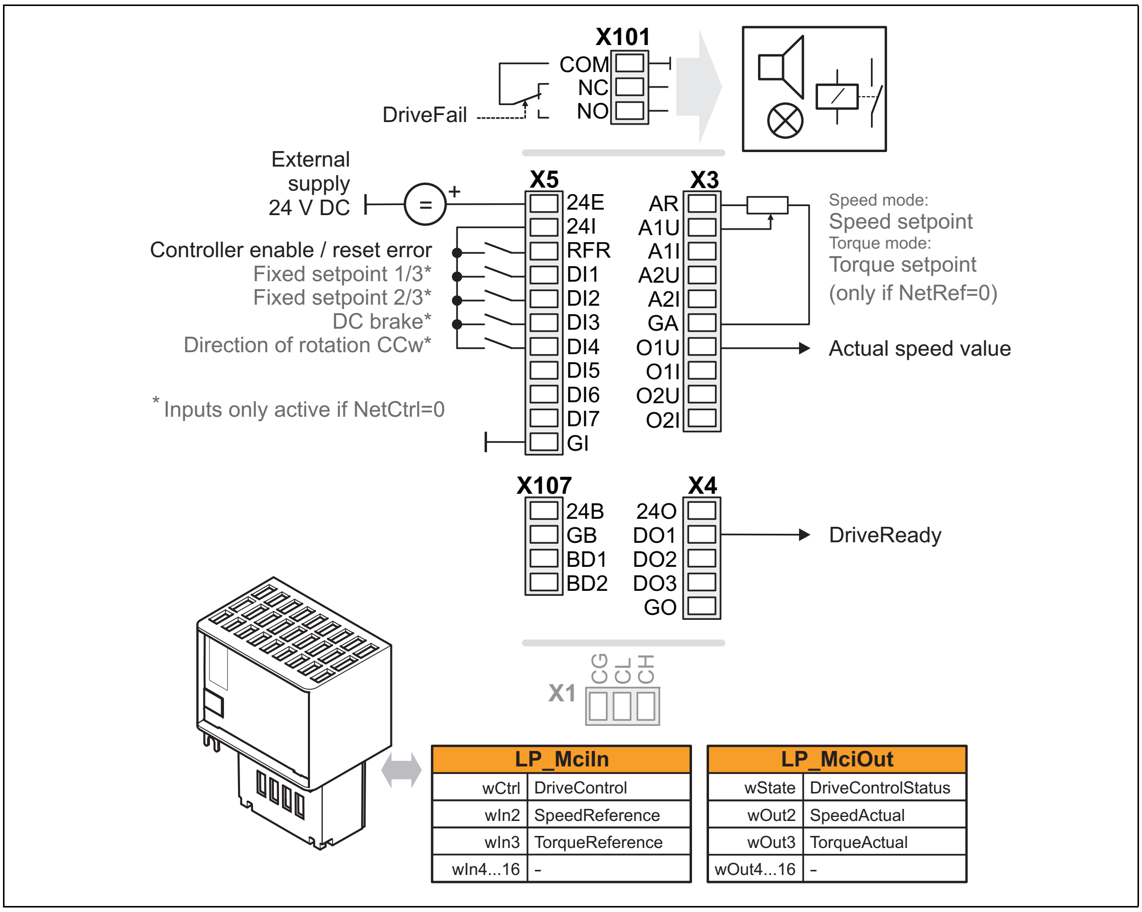
8.3.4.7 CAN



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl.bSetDCBrake		**X3/A2I**	-
X5/DI4	LA_NCtrl.bSetSpeedCcw		**X3/O1U**	LA_NCtrl.NMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

▶ Process data assignment for fieldbus communication (📖 474)

8.3.4.8 MCI



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_NCtrl.bDriveFail			
X5/RFR	LA_NCtrl.bFailReset		**X3/A1U**	LA_NCtrl.nAuxSetValue_a *
X5/DI1	LA_NCtrl.bJogSpeed1		**X3/A1I**	-
X5/DI2	LA_NCtrl.bJogSpeed2		**X3/A2U**	-
X5/DI3	LA_NCtrl.bSetDCBrake		**X3/A2I**	-
X5/DI4	LA_NCtrl.bSetSpeedCcw		**X3/O1U**	LA_NCtrl.nMotorSpeedAct_a *
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
				* 10 V ≡ 100 % reference speed (C00011)
X107/BD1	-		**X4/DO1**	LA_NCtrl.bDriveReady
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

▶ Process data assignment for fieldbus communication (📖 504)

8.3.5 Process data assignment for fieldbus communication

The fieldbus communication is connected (preconfigured) to the previously selected technology application by selecting the corresponding control mode in C00007:

- "30: CAN" for the connection to the system bus (CAN)
- "40: MCI" for the connection to a plugged-in communication module (e.g. EtherNet/IP™)

The assignment of the process data words depends only on the application, not on the bus system used:

Input words	Name	Assignment
Word 1	DriveControl	Control word • For bit assignment see the table below. • Display parameter: C01351/1
Word 2	SpeedReference	Speed setpoint in [rpm] ▸ Scaling of speed and torque values
Word 3	TorqueReference	Torque setpoint in [Nm] ▸ Scaling of speed and torque values
Word 4	-	Not preconfigured
Words 5 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Control word	Name	Function
Bit 0	Run Forward	Relationships between Run1 and Run2 and trigger events can be found in the chapter "Run/Stop event".
Bit 1	Run Backward	
Bit 2	Fault Reset	0↗1 ≡ Reset error 0 ≡ No response
Bit 3	Reserved	-
Bit 4	Reserved	-
Bit 5	NetCtrl	Run/Stop control: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 6	NetRef	Reference speed/reference torque: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 7 ... 15	Reserved	-

Output words	Name	Assignment
Word 1	DriveControlStatus	Status word • For bit assignment see the table below. • Display parameter: C01352/1
Word 2	SpeedActual	Actual speed value in [rpm] ▸ Scaling of speed and torque values
Word 3	TorqueActual	Actual torque in [Nm] ▸ Scaling of speed and torque values
Word 4	-	Not preconfigured
Words 5 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Status word	Name	Status
Bit 0	Faulted	0 ≡ No errors 1 ≡ Errors have occurred
Bit 1	Warning	0 ≡ No warnings 1 ≡ Warnings have occurred
Bit 2	Running1 (Fwd)	Relationships between Run1 and Run2 and trigger events can be found in the chapter "Run/Stop event".
Bit 3	Running2 (Rev)	
Bit 4	Ready	0 ≡ Different status than in case of "1" 1 ≡ Ready or Enabled or Stopping
Bit 5	Ctrl from Net	Run/Stop control: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 6	Ref from Net	Reference speed/reference torque: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 7	At Reference	1 ≡ The drive currently runs with reference speed (speed mode) or reference torque (Torque mode)
Bit 8	Drive State	The "Drive State" is coded as follows: 0: Manufacturer-specific (not used with 8400) 1: Start-up (drive initialisation) 2: Not_Ready (mains voltage switched off) 3: Ready (mains voltage switched-on) 4: Enabled (drive has received "Run" command) 5: Stopping (drive has received "Stop" command and is stopped) 6: Fault_Stop (drive is stopped due to an error) 7: Faulted (errors have occurred)
Bit 9	Drive State	
Bit 10	Drive State	
Bit 11	Drive State	
Bit 12	Drive State	
Bit 13	Drive State	
Bit 14	Drive State	
Bit 15	Drive State	

8.3.5.1 Run/Stop event

Relationships between Run1 and Run2:

	Starter					Drive
	Contactor	Starter	Reverser	Speed	Soft start	
Run1	Close	Run	RunFwd	RunLow	RunRamp1	RunFwd
Run2	No Action	No Action	RunRev	RunHigh	RunRamp2	RunRev

Run1 and Run2 trigger:

Run1	Run2	Trigger event	Run type
0	0	Stop,	No Action
0 → 1	0	Run	Run1
0	0 → 1	Run	Run2
0 → 1	0 → 1	No Action	No Action
1	1	No Action	No Action
1 → 0	1	Run	Run2
1	1 → 0	Run	Run1

8.3.5.2 Scaling of speed and torque values

Scaling of the speed values

The speed setpoint is defined by the bus in [rpm]. The conversion then takes place in the inverter as all speed-related signals are processed to a reference variable in percent. An adjustable scaling factor serves to make an additional scaling.

Equation for scaling the speed setpoint

$$\text{Speed setpoint}_{\text{Application}} = \text{Speed setpoint}_{\text{Bus}}[\text{rpm}] \cdot \frac{16384}{\text{Reference speed [rpm]}} \cdot \frac{1}{2^{\text{Scaling factor}}}$$

Parameters	Name	Description
C00011	Appl.: Reference speed	Reference variable for speed-related signals
C01353/1	ACDrive: Speed scaling	In the Lenze setting "0", no scaling takes place ($2^0 = 1$)

For the output of the actual speed value to the bus, the following conversion is made:

Equation for scaling the actual speed value

$$\text{Actual speed value}_{\text{Bus}}[\text{rpm}] = \text{Actual speed value}_{\text{Application}} \cdot \frac{\text{Reference speed [rpm]}}{16384} \cdot 2^{\text{Scaling factor}}$$

Parameters	Name	Description
C00011	Appl.: Reference speed	Reference variable for speed-related signals
C01353/1	ACDrive: Speed scaling	In the Lenze setting "0", no scaling takes place ($2^0 = 1$)

Scaling of the torque values

The torque setpoint is defined by the bus in [Nm]. The conversion then takes place in the inverter as all torque-related signals are processed to a reference variable in percent. An adjustable scaling factor serves to make an additional scaling.

Equation for scaling the torque setpoint

$$\text{Torque setpoint}_{\text{Application}} = \text{Torque setpoint}_{\text{Bus}}[\text{Nm}] \cdot \frac{16384 \cdot 100}{\text{Maximum torque [0.01 Nm]}} \cdot \frac{1}{2^{\text{Scaling factor}}}$$

Parameters	Name	Description
C00057	Maximum torque	Reference variable for torque-related signals
C01353/2	ACDrive: Torque scaling	In the Lenze setting "0", no scaling takes place ($2^0 = 1$)

For the output of the actual torque value to the bus, the following conversion is made:

Equation for scaling the actual torque

$$\text{Actual torque}_{\text{Bus}}[\text{Nm}] = \text{Actual torque}_{\text{Application}} \cdot \frac{\text{Maximum torque [0.01 Nm]}}{16384 \cdot 100} \cdot 2^{\text{Scaling factor}}$$

Parameters	Name	Description
C00057	Maximum torque	Reference variable for torque-related signals
C01353/2	ACDrive: Torque scaling	In the Lenze setting "0", no scaling takes place ($2^0 = 1$)

8.3.6 AC Drive Profile diagnostic parameters

In the **All parameters** tab, the parameters for diagnostic purposes listed in the following table are displayed in the **AC Drive Profile** category.

Note!

These parameter are set by the EtherNet/IP™ communication module and should not be written by the user.

Parameters	Info	Lenze setting
C01350/1	ACDrive: DriveMode	1: Speed mode
C01351/1	ACDrive: Control word	-
C01352/1	ACDrive: Status word	-
C01353/1	ACDrive: Speed scaling	0
C01353/2	ACDrive: Torque scaling	0
Greyed out = display parameter		

8.3.7 Setting parameters (short overview)

Parameters	Info	Lenze setting	
		Value	Unit
C00012	Accel. time - main setpoint	2.000	s
C00013	Decel. time - main setpoint	2.000	s
C00019	Auto-DCB: Threshold	3	rpm
C00024	LS_DriveInterface: bNActCompare	0.00	%
C00036	DCB braking: Current	50.00	%
C00039/1	Preset setpoint 1	40.00	%
C00039/2	Preset setpoint 2	60.00	%
C00039/3	Preset setpoint 3	80.00	%
C00039/4...15	Fixed setpoint 4 ... 15	0.00	%
C00101/1...15	Add. accel. time 1 ... 15	0.000	s
C00103/1...15	Add. decel. time 1 ... 15	0.000	s
C00105	Decel. time - quick stop	2.000	s
C00106	Auto-DCB: Hold time	0.500	s
C00107	DCB braking: Hold time	999.000	s
C00134	L_NSet_1: Ramp smoothing	0: Off	
C00182	L_NSet_1: S-ramp time PT1	20.00	s
C00190	L_NSet_1: Setpoint arithmetic	0: Out = Set	
C00220	L_NSet_1: Acceleration time - add. setpoint	0.000	s
C00221	L_NSet_1: Deceleration time - add. setpoint	0.000	s
C00222	L_PCTRL_1: Vp	1.0	
C00223	L_PCTRL_1: Tn	400	ms
C00224	L_PCTRL_1: Kd	0.0	

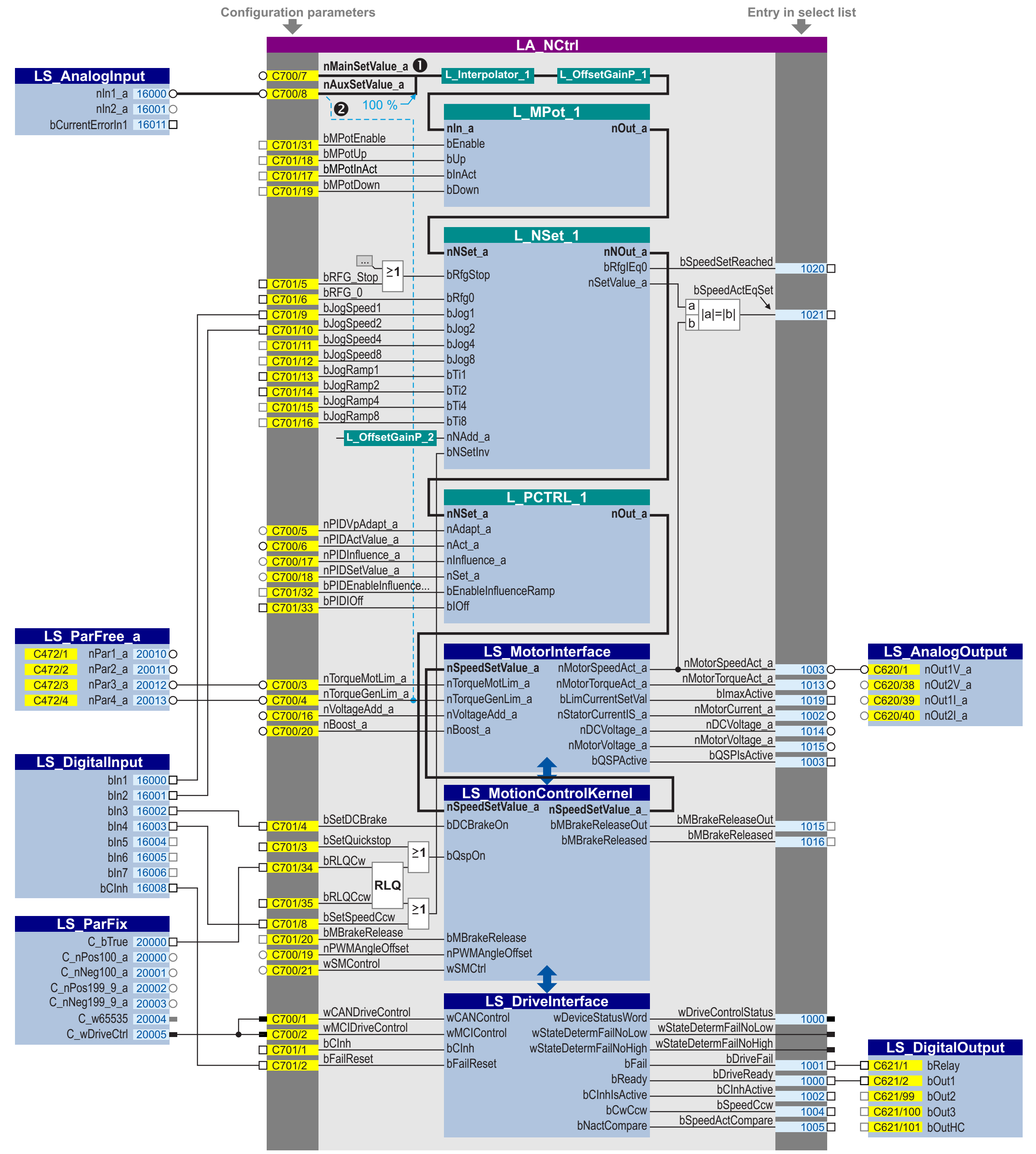
Parameters	Info	Lenze setting	
		Value	Unit
C00225	L_PCTRL_1: MaxLimit	199.99	%
C00226	L_PCTRL_1: MinLimit	-199.99	%
C00227	L_PCTRL_1: Acceleration time	0.010	s
C00228	L_PCTRL_1: Deceleration time	0.010	s
C00233	L_PCTRL_1: Root function	0: Off	
C00241	L_NSet_1: Hyst. NSet reached	0.50	%
C00242	Operating mode of process controller	0: Off	
C00243	L_PCTRL_1: Accel. time influence	5.000	s
C00244	L_PCTRL_1: Deceleration time influence	5.000	s
C00632/1	L_NSet_1: Blocking speed 1 max	0.00	%
C00632/2	L_NSet_1: Blocking speed 2 max	0.00	%
C00632/3	L_NSet_1: Blocking speed 3 max	0.00	%
C00633/1	L_NSet_1: Blocking speed 1 min	0.00	%
C00633/2	L_NSet_1: Blocking speed 2 min	0.00	%
C00633/3	L_NSet_1: Blocking speed 3 min	0.00	%
C00635	L_NSet_1: nMaxLimit	199.99	%
C00636	L_NSet_1: nMinLimit	-199.99	%
C00670	L_OffsetGainP_1: Gain	1.0000	
C00671	L_OffsetGainP_2: Gain	1.0000	
C00672	L_OffsetGainP_3: Gain	1.0000	
C00696	L_OffsetGainP_1: Offset	0.00	%
C00697	L_OffsetGainP_2: Offset	0.00	%
C00698	L_OffsetGainP_3: Offset	0.00	%
C00800	L_MPot_1: Upper limit	100.00	%
C00801	L_MPot_1: Lower limit	-100.00	%
C00802	L_MPot_1: Acceleration time	10.0	s
C00803	L_MPot_1: Deceleration time	10.0	s
C00804	L_MPot_1: Inactive fct.	0: Retain value	
C00805	L_MPot_1: Init fct.	0: Load last value	
C00806	Use of motor potentiometer	0: No	

Related topics:

▶ "GeneralPurpose" functions (📖 571)

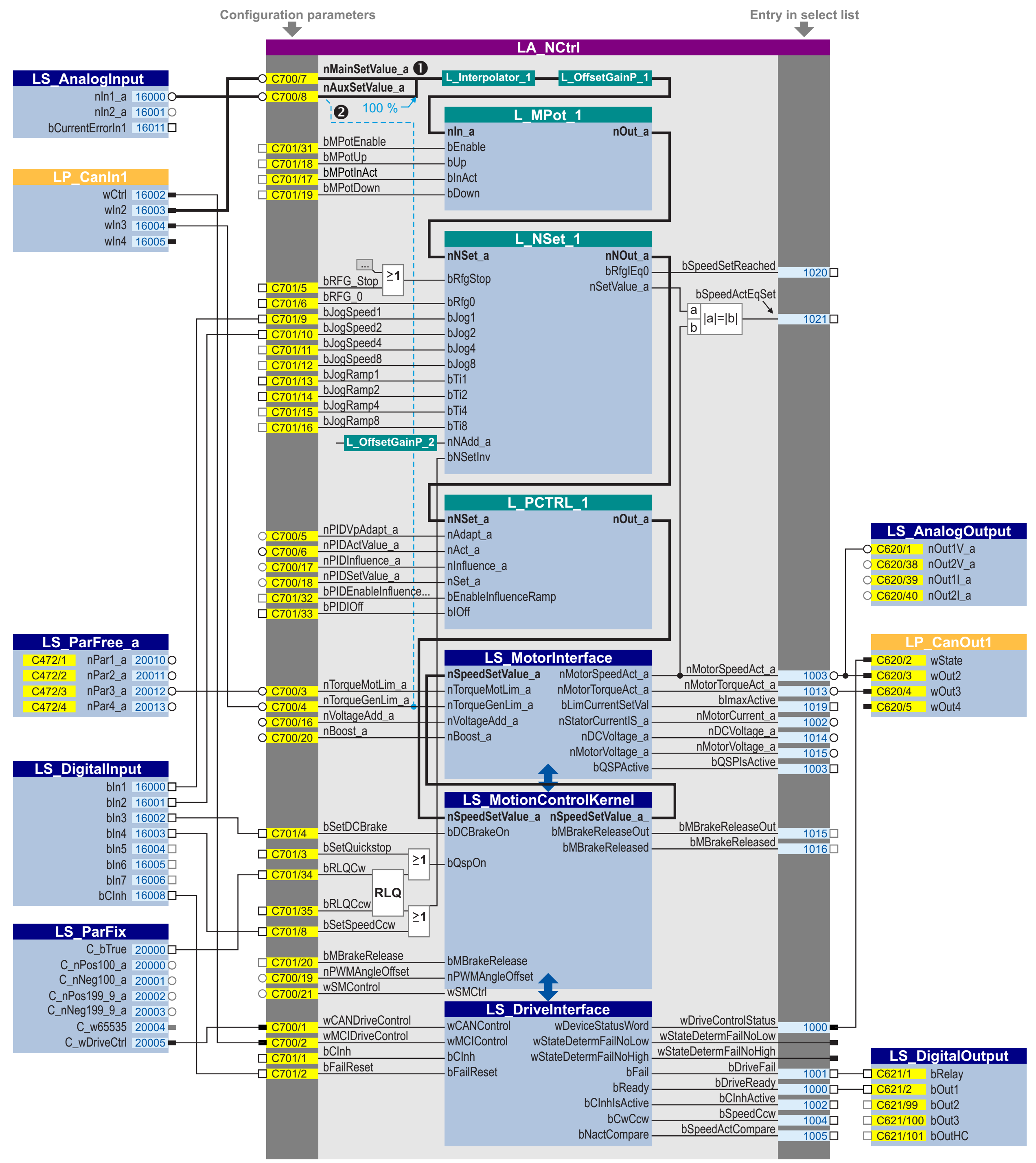
8.3.8 Configuration parameters

If required, the subcodes of C00700 and C00701 serve to change the pre-configured assignment of the application inputs:



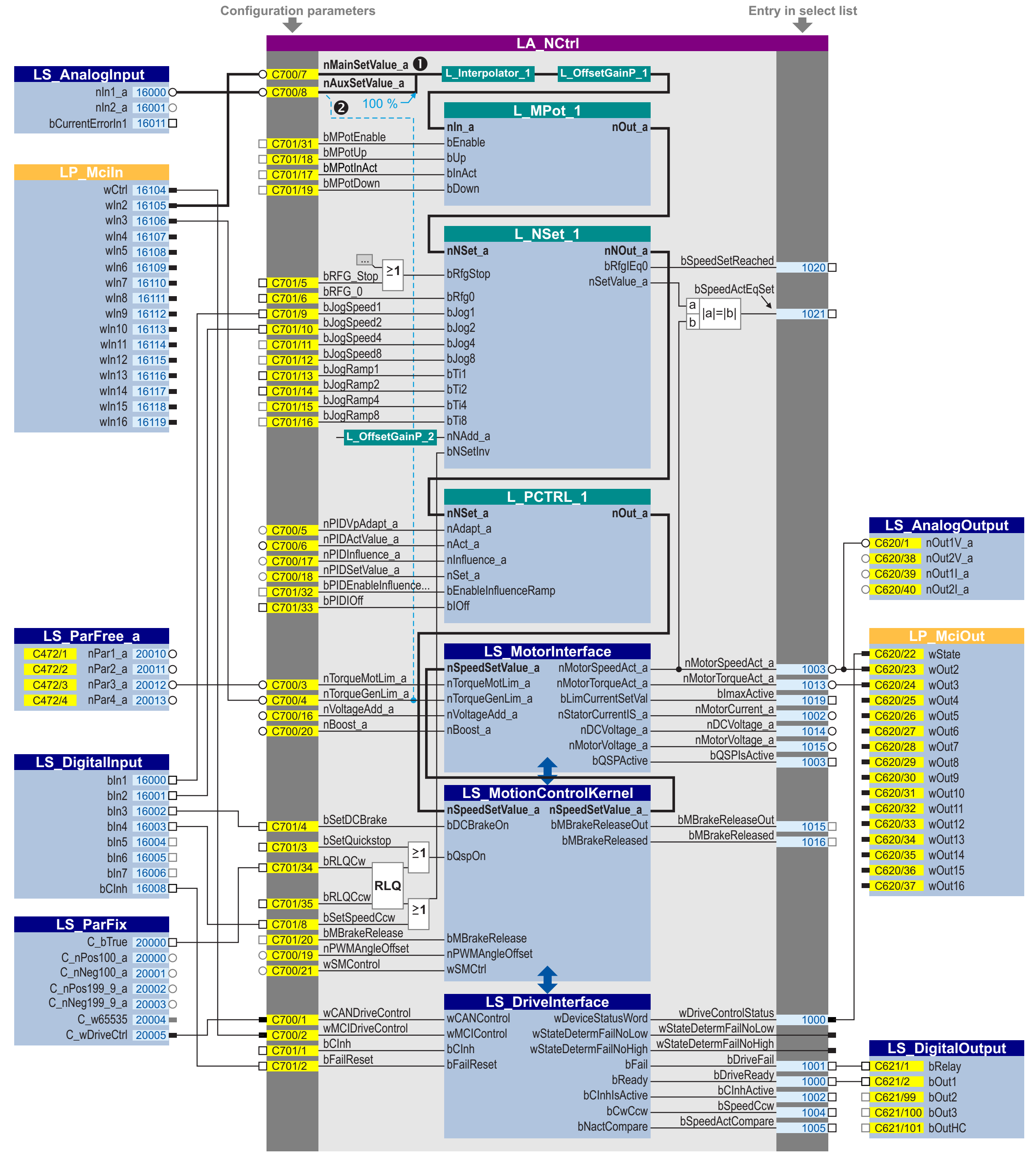
❶ *nMainSetValue_a* input is only effective when NetRef=1; *nAuxSetValue_a* input is only effective when NetRef=0
❷ Torque mode: *nAuxSetValue_a* input is internally connected to *nTorqueGenLim_a*; speed setpoint = 100 % (fixed)

[8-7] Pre-assignment of the "Actuating drive speed" (AC Drive Profile) application in the "Terminals 0" control mode



❶ *nMainSetValue_a* input is only effective when NetRef=1; *nAuxSetValue_a* input is only effective when NetRef=0
❷ Torque mode: *nAuxSetValue_a* input is internally connected to *nTorqueGenLim_a*; speed setpoint = 100 % (fixed)

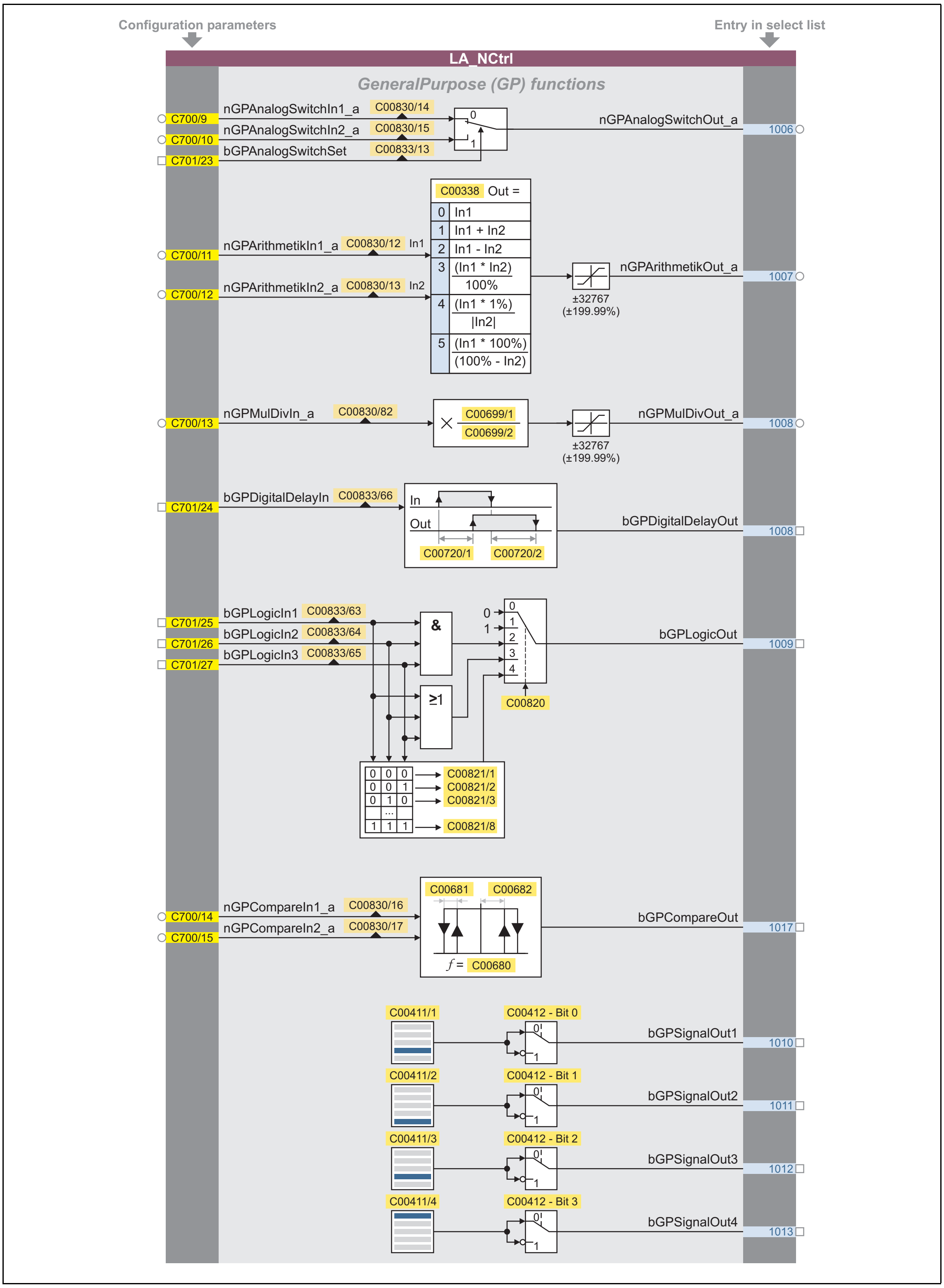
[8-8] Pre-assignment of the "Actuating drive speed" (AC Drive Profile) application in the "CAN " control mode



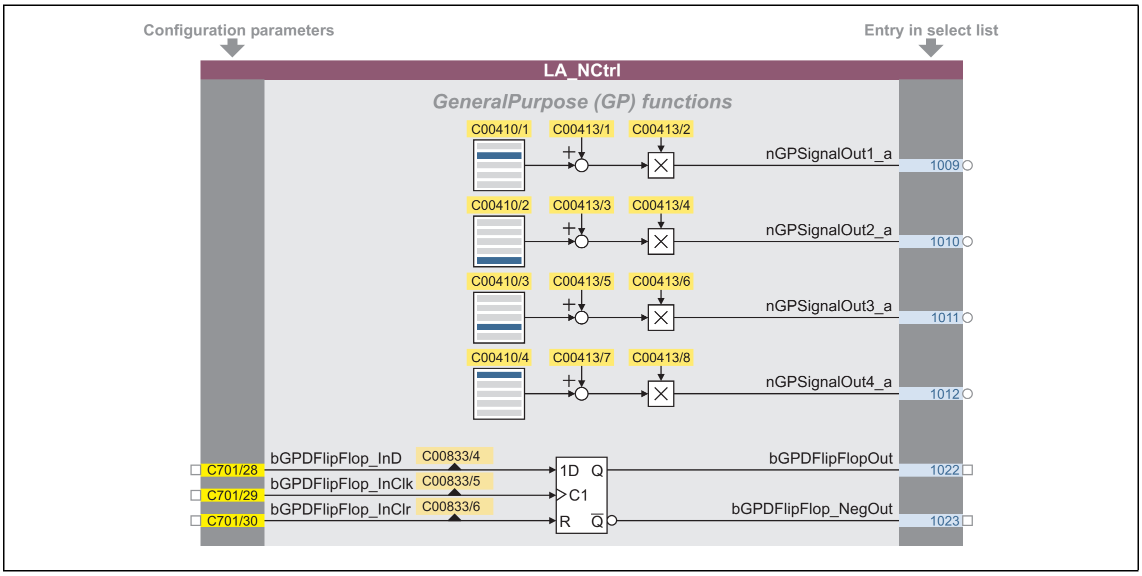
❶ *nMainSetValue_a* input is only effective when NetRef=1; *nAuxSetValue_a* input is only effective when NetRef=0
❷ Torque mode: *nAuxSetValue_a* input is internally connected to *nTorqueGenLim_a*; speed setpoint = 100 % (fixed)

[8-9] Pre-assignment of the "Actuating drive speed" (AC Drive Profile) application in the "MCI" control mode

Configuration parameters for "GeneralPurpose" functions



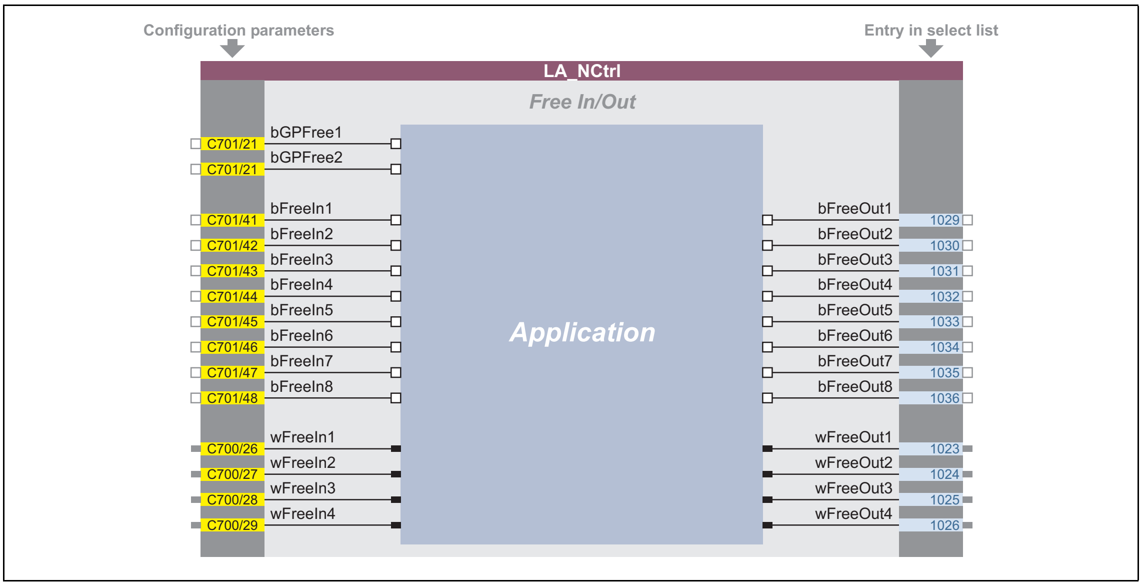
[8-10] "GeneralPurpose" functions



[8-11] "GeneralPurpose" functions (continuation)

Free inputs and outputs

These inputs can be freely interconnected in the application level. They can be used to transfer signals from the I/O level to the application level and vice versa.



[8-12] Free inputs/outputs

Related topics:

- User-defined terminal assignment (📖 445)
- "GeneralPurpose" functions (📖 571)

8.4 TA "Table positioning"

Numerous functions for the motion control of a single axis can be executed with the "Table positioning" technology application. For this, the technology application accesses the basic functions implemented in the **Motion Control Kernel** which are described in detail in the main chapter "Basic drive functions (MCK)". (📖 577)

- Manual jog
- Retracting from limit switches
- Reference setting/homing in 18 different modes (*in preparation*)
- Positioning (absolute, relative)
- Continuous travel (speed mode)
- Speed override
- Acceleration override
- Profile sequence block control
- Travel block restart within the profile (online profile start)
- Specification of target position via process signal interface (e.g. via fieldbus)

Further functions

- Up to 15 travel sets can be set via parameters
- Parameterisation of the profile data in physical units or relative values
- Adjustable torque limitation
- Freely selectable, variable ramp shape
- Motor potentiometer function
- Software limit position monitoring
- Following error monitoring (with static limits)
- Automatic holding brake control
- Quick stop (QSP) with adjustable ramp time
- Enable of individual functions via control word
- Status and diagnostic displays
- Operating mode changeover (manual jog, homing, speed follower, positioning)
- Integrated, freely available "GeneralPurpose" functions:
  Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-FlipFlop, counter
- From version 12.00.00, the application interconnection can be extended by an internal sequence control using the FB L_Sequencer_1 if required.

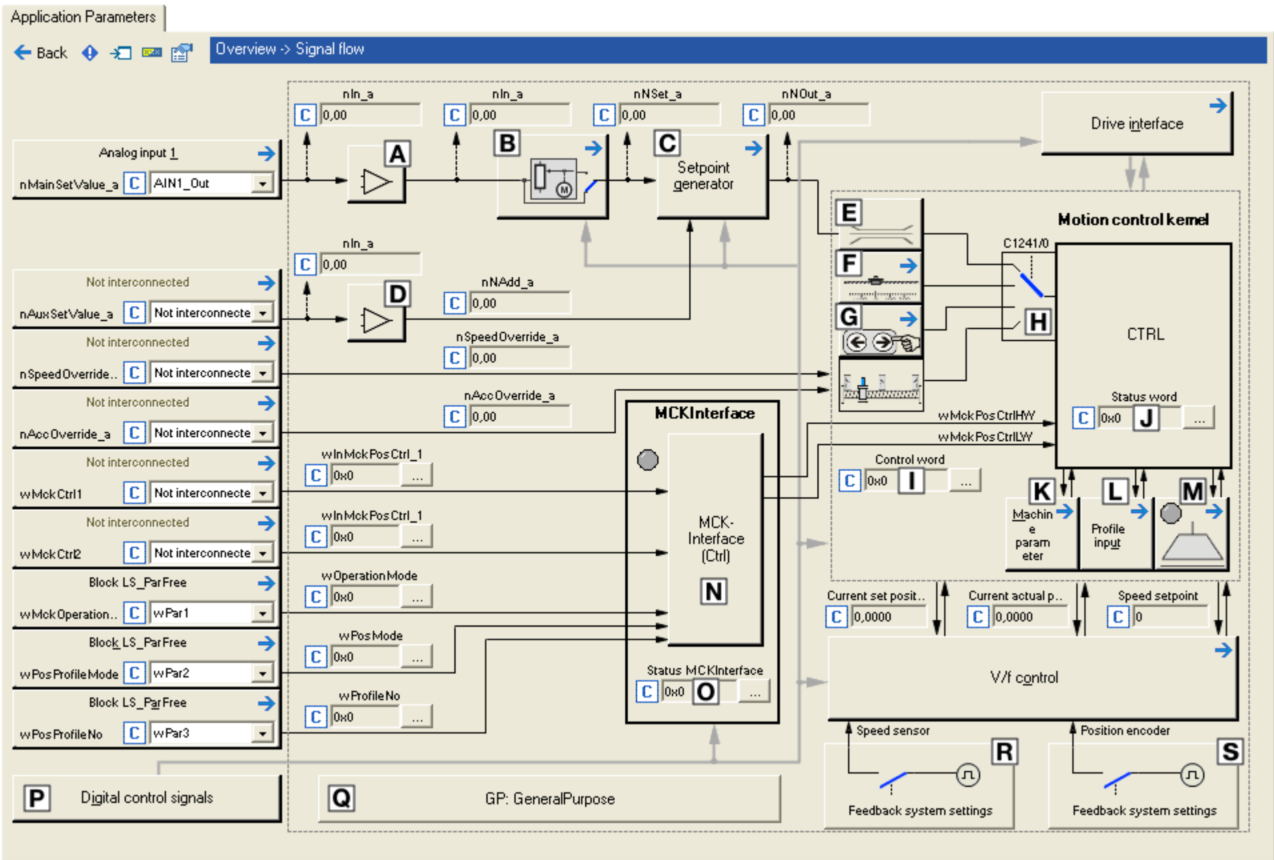
Related topics:

▸ Commissioning of the "Table positioning" technology application (📖 73)

8.4.1 Basic signal flow

The main element of the technology application is the **Motion Control Kernel**.

- The **Motion Control Kernel** is controlled via a 32-bit double control word. All motion profiles in the different operating modes can be controlled with this interface.
- Upstream to **the Motion Control Kernel** there is a so-called MCKInterface called which provides additional process inputs for the application in order to control the **Motion Control Kernel**.



[8-13] Signal flow of the table positioning

- A Offset and gain (L_OffsetGainP_1)
- B Motor potentiometer function (L_MPot_1)
- C Setpoint generator (L_NSet_1)
- D Offset and gain (L_OffsetGainP_2)
- E Speed setpoint input limitation
- F Basic function "Homing"
- I MCK control word
- J MCK status word
- K Selection of the machine parameters
- L Profile entry for the "Positioning" basic function.
- STOP Holding brake control
- Warn
- Warn
- Code
- Warn
- Warn
- Bus
- IMP
- Warn
- N MCKInterface

- G Basic function "Manual jog"
- H Operating mode changeover (is executed by the MCKInterface)
- O Status word of the MCKInterface
- P Terminal assignment & display of digital control signals
- Q Integrated disposable "GeneralPurpose" functions: Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-flipflop

8.4.1.1 Possibilities for the position selection

The following options are available for the selection of different positions:

1. Changeover of profiles
2. Changing the profile position via parameter data (SDO)
3. Changing the profile position via process data (PDO)
   - The profile position is defined in [increments] or in the application unit [units].

	Changeover of profiles	Changing the profile position via SDO	Changing the profile position via PDO	
Number of positions	max. 15	n > 15	n > 15	
Position selection	Parameterisation in the profile	Selection via SDO	Selection via PDO	
Unit	units	units	Increments	units
Remainder allowance	in the drive	in the drive	in the PLC	in the drive

Note!

Setting the machine parameters is a basic prerequisite for the operating modes "Homing", "Manual jog" and "Positioning".

The more precisely the machine parameters are set, the better the results of positioning!

▸ Machine parameters (📖 611)

Possibility 1: Changeover of profiles

A total of 15 profiles is available in the 8400 TopLine.

- A profile describes a motion task that can be converted into a rotary motion by the **Motion Control Kernel** in the "Positioning" operating mode.
- A detailed explanation of all profile parameters can be found in the subchapter "Profile entry". (📖 671) of the description of the "Positioning" basic drive function.
- The selection of the profile to be executed can either be executed as a data word via the input *wPosProfileNo* or binary coded via the inputs *bPosProfileNo_1* ... *bPosProfileNo_4*.
- The chosen profile is started via the process input *bPosExecute* of the FB L_MckCtrlInterface or the control bit "PosExecute" in the MCK control word.

Possibility 2: Changing the profile position via parameter data (SDO)

The parameter data channel of a fieldbus can also be used to change the position in a profile.

- The position of the profiles 1 ... 15 are stored in the subcodes 1 ... 15 of the code C01301.
- More than 15 positions are available via the profile changeover (as described under possibility 1).

Possibility 3: Changing the profile position via process data (PDO)

For selecting the position in [increments], the *dnPosProfilePosition_p* input at the application block is available.

Note!

For selecting the position in [units], the two inputs *wPosProfileUnitsLW* and *wPosProfileUnitsHW* at the application block are available.

- In the application level, the position in the FB L_MckCtrlInterface is converted from [units] to [increments] again and output to *dnPosSetOut_p*. For this purpose, the respective mode for position calculation must be selected in C01296/1.
- Since the conversion is not executed within one controller cycle, the data at the *dnPosSetOut_p* output can be out-of-date. The *bPosSetDataValid* output is set to TRUE when the conversion is completed and the position in increments is valid. Then, the position can be transmitted to the profile (see the following section).

The acceptance of the position into the currently selected profile is executed with a FALSE-TRUE edge at the *bPosSetProfilePosition* input.

- If bit 2 is set in C01297, the applied setpoint position is accepted into the currently selected profile and the profile is started directly when setting the process input *bPosExecute* to TRUE or setting the control bit "PosExecute" in the MCK control word.
- If bit 3 is set in C01297, the setpoint positions at the MCKInterface are automatically accepted into the profile with the applied profile number if a change of data is detected at the corresponding input for the setpoint position.
- The accepted position is stored in the code C01301/x in the application unit [units]. By reading out the code, you can check if the position was correctly accepted into the profile.

Note!

How to store changed profile parameters safely against mains failure in the memory module: Set C00002/11 = "1: On / Start".

8.4.2 Internal interfaces | application block "LA_TabPos"

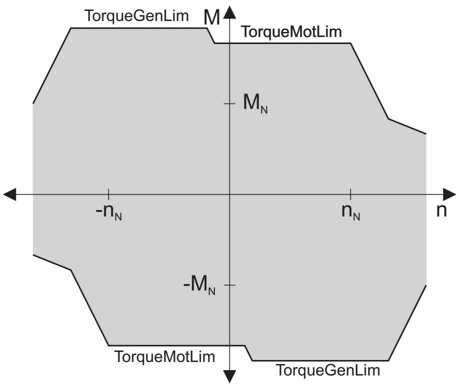
Note!

The connectors greyed out in the following table are hidden in the function block editor in the Lenze setting.

- These connections can be shown via the **Connector visibilities** command in the *Context menu* of the application block.

inputs

Designator Data type	Information/possible settings	
wCANDriveControl WORD	Control word via system bus (CAN) for device control • See the "wCANControl/wMCIControl control words" subchapter of the chapter on device control for a detailed description of the individual control bits.	
wMCIDriveControl WORD	Control word via communication module (e.g. PROFIBUS) for device control • See the "wCANControl/wMCIControl control words" subchapter of the chapter on device control for a detailed description of the individual control bits.	
wSMControl WORD	Interface to the optional safety system. • Setting control bit 0 ("SafeStop1") in this control word causes e.g. the automatic deceleration of the drive to standstill within this application (in the **Motion Control Kernel**). • See the subchapter "Interface to safety system" of the chapter on basic drive functions for a detailed description of the individual control bits.	
bCInh BOOL	Enable/inhibit inverter	
	FALSE	Enable inverter: The inverter switches to the "OperationEnabled" device status if no other source for controller inhibit is active. • C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.
	TRUE	Inhibit inverter (controller inhibit): The inverter switches to the "SwitchedOn" device status.
bFailReset BOOL	Reset error message In the Lenze setting this input is connected to the digital input controller enable so that a possibly existing error message is reset together with the controller enable (if the cause for the fault is eliminated).	
	TRUE	The current fault is reset, if the cause for the fault is eliminated. • If the fault still exists, the error status remains unchanged.
bSetQuickstop BOOL	Activate quick stop (QSP) • Also see device command "Activate/deactivate quick stop".	
	TRUE	Activate quick stop • Motor control is decoupled from the setpoint selection and, within the deceleration time parameterised in C00105, the motor is brought to a standstill (n_{act} = 0). • The motor is kept at a standstill during closed-loop operation. • A pulse inhibit is set if the auto-DCB function has been activated via C00019.
	FALSE	Deactivate quick stop • The quick stop is deactivated if no other source for the quick stop is active. • C00159 displays a bit code of active sources/causes for the quick stop.

Designator Data type	Information/possible settings	
nTorqueMotLim_a nTorqueGenLim_a INT	Torque limitation in motor mode and in generator mode • These input signals are directly transferred to the motor control to limit the inverter's maximum torque in motor and generator mode. • The drive cannot output a higher torque in motor/generator mode than set here. • The applied values (any polarity) are internally interpreted as absolute values. • If V/f characteristic control (VFCplus) is selected, limitation is indirectly performed via a so-called I_{max} controller. • If sensorless vector control (SLVC) or servo control (SC) is selected, limitation has a direct effect on the torque-producing current component. • Scaling: 16384 ≡ 100 % M_{max} (C00057) **Note:** Setting this input is ineffective in the reference modes 14 and 15 ("Homing to positive stop").	
	Torque limits in motor and generator mode:  	
bSetSpeedCcw BOOL	Change of direction of rotation • For instance if a motor or gearbox is fixed laterally reversed to a machine part, but the setpoint selection should still be executed for the positive direction of rotation.	
	FALSE	Clockwise rotation (Cw)
	TRUE	Direction of rotation to the left (Ccw)
nMainSetValue_a INT	Main speed setpoint • Offset and gain of this input signal can be set in C00696 and C00670 for a simple signal adjustment of a setpoint encoder. • Scaling: 16384 ≡ 100 % reference speed (C00011) • The main setpoint is transformed to a speed setpoint in the setpoint encoder via a ramp function generator with linear or S-shaped ramps. • Upstream to the ramp function generator, a blocking speed masking function and a setpoint MinMax limitation are effective. • For a detailed functional description see the L_NSet FB.	
nAuxSetValue_a INT	Additional speed setpoint • Offset and gain of this input signal can be set in C00697 and C00671 for a simple signal adjustment of a setpoint encoder. • Scaling: 16384 ≡ 100 % reference speed (C00011) • The additional speed setpoint can be linked arithmetically with the main speed setpoint behind the ramp function generator. • The additional speed setpoint can be shown via ramp times of a second ramp function generator. • For a detailed functional description see the L_NSet FB.	
bJogSpeed1 bJogSpeed2 BOOL	Inputs for overriding fixed setpoints (JOG setpoints) for the main setpoint • Selection inputs are binary coded.	

Designator Data type	Information/possible settings	
bJogSpeed1 bJogSpeed2 BOOL	Inputs for overriding fixed setpoints (JOG setpoints) for the main setpoint • A fixed setpoint for the setpoint generator can be activated instead of the main setpoint via these selection inputs. • The two selection inputs are binary coded, therefore you can select 3 fixed setpoints. • In the case of binary coded selection "0" (all inputs = FALSE or not assigned), main setpoint *nMainSetValue_a* is active. • The selection of the fixed setpoints is carried out in C00039/1...3 in [%] based on the reference speed (C00011). • For a detailed functional description see the L_NSet FB.	
Motor potentiometer Alternatively to the input signal *nMainSetValue_a*, the main setpoint can also be generated by a motor potentiometer function. • In the Lenze setting, the motor potentiometer function is deactivated. • Activation is possible via C00806 or via the *bMPotEnable* input. • The behaviour of the motor potentiometer during switch-on of the drive system can be selected in C00805. • For a detailed functional description see the L_MPot FB.		
bMPotEnable BOOL	Activating the motor potentiometer function • This input and C00806 are OR'd.	
	TRUE	The motor potentiometer function is active; the speed setpoint can be changed via the *bMPotUp* and *bMPotDown* control inputs.
bMPotUp BOOL	Increasing the speed setpoint	
	TRUE	Approach the upper speed limit value set in C00800 with the acceleration time set in C00802.
bMPotDown BOOL	Decreasing the speed setpoint	
	TRUE	Approach the lower speed limit value set In C00801 with the deceleration time set in C00803.
MCK basic functions		
bMBrakeRelease BOOL	Holding brake control: Release/apply brake • In conjunction with the operating mode selected in C02580 (Lenze setting: "Brake control off").	
	FALSE	Apply brake. • During automatic operation, the internal brake logic controls the brake.
	TRUE	Release brake manually (forced release). • **Note!** The brake can also be released when the controller is inhibited! • During automatic operation, the internal brake logic is deactivated and the brake is released (supervisor operation). If a controller inhibit has been set by the brake control, it will be deactivated. • In semi-automatic operation, the brake is released including feedforward control.
wMckCtrl1 wMckCtrl2 WORD	Direct selection of MCK control words • E.g. via a master control which has been connected to the fieldbus, too. For this purpose, the control word inputs can directly be connected to the **LP_MciIn** or **LP_CanIn** fieldbus interface. • See the "MCK control word" chapter for a detailed description of the individual control bits.	

Designator Data type	Information/possible settings	
wMckOperationMode WORD	Selection of the operating mode of the **Motion Control Kernel** • Only bit 0 ... bit 3 of *wMckOperationMode* is evaluated. • If an invalid operating mode is selected, the response set in C00595/11 is activated (Lenze setting: "Warning"). • The current operating mode is displayed in C01243.	
	0	Speed follower
	1	Homing
	2	Manual jog
	3	Positioning
	4	Stop
	5	Position follower
	6 ... 15	Reserved for future extensions
bMckOperationMode_1 ... bMckOperationMode_8 BOOLBOOL	Binary-coded selection of the operating mode of the **Motion Control Kernel** • See the "MCK control word" chapter for a detailed description of the individual control bits. • If an invalid operating mode is selected, the response set in C00595/11 is activated (Lenze setting: "Warning"). • The current operating mode is displayed in C01243.	
bPosCtrlOn BOOL	Position control/Angle control • Pre-configured assignment: TRUE (position control active)	
	FALSE	Deactivate position control/angle control
	TRUE	Activate position control/angle control
nPosCtrlOutLimit_a INT	Limitation of the position controller output • Scaling: 16384 ≡ 100 % reference speed (C00011) • Pre-configured assignment: 100 %	
nPosCtrlPAdapt_a INT	Adaptation of the position controller gain • Scaling: 16384 ≡ 100 % Vp (C00254) • Pre-configured assignment: 100 %	
bLimitSwitchPos bLimitSwitchNeg BOOL	Limit position monitoring: Inputs for positive/negative limit switch	
bReleaseLimitSwitch BOOL	Manual jog: Retract operated limit switch	
	TRUE	Retract operated limit switch (in opposite direction)
bManJogPos bManJogNeg BOOL	Manual jog: *bManJogPos* = TRUE: Manual jog right *bManJogNeg* = TRUE: Manual jog left Both inputs = TRUE: No change compared to previous state Both inputs = FALSE: Stop manual jog	
bManJogExecute2ndVel BOOL	Manual jog: Changeover to speed 2	
	FALSE	Speed 1 (C01231/1) active
	TRUE	Speed 2 (C01231/2) active
bEnableSpeedOverride BOOL	Speed override	
	TRUE	Activate speed override
nSpeedOverride_a INT	Value for Speed override • Percentage multiplier for the currently active speed. • 16384 ≡ 100 % of the maximum traversing speed (display in C01211/1). • If the override value is 0 %, the drive is brought to a standstill.	
bEnableAccOverride BOOL	Acceleration override	
	TRUE	Activate acceleration override

Designator Data type	Information/possible settings	
nAccOverride_a INT	Value for Acceleration override • Percentage multiplier for the currently active acceleration. • 16384 ≡ 100 % of the parameterised acceleration of the corresponding operating mode. • If the override value is 0 %, acceleration ceases.	
bHomeStartStop BOOL	Homing: Start/stop homing	
	TRUE	Start homing
	TRUE↘FALSE	Stop homing
bHomeSetPosition BOOL	Homing: Set home position	
	TRUE	Set home position
bHomeResetPosition BOOL	Homing: Reset home position	
	TRUE	Reset the "Reference known" status
bHomeMark BOOL	Homing: Input for reference switch (pre-switch off mark) • This input responds to the FALSE status (fail-safe) and is to be connected to the corresponding digital input to which the reference switch is connected.	
bPosSetProfilePosition BOOL	Position teaching: MCK setpoint position	
	TRUE	Teach MCK setpoint position into the selected profile.
bPosSetActualPosition BOOL	Position teaching: Current position	
	TRUE	Teach current position into the selected profile.
bPosExecute BOOL	Positioning: Start travelling	
	FALSE↗TRUE	Execute selected profile
bPosFinishTarget BOOL	Positioning: Complete cancelled profile	
	FALSE↗TRUE	A positioning process previously cancelled, e.g. by *bPosStop* or due to a device error, is resumed by travelling to the original target.
bPosDisableFollowProfile BOOL	Positioning: Do not execute sequence profile (switch-off profile linkage)	
	TRUE	Evaluation of the sequence profile number parameterised in C01307/1...15 for the selected profile is suppressed.
bPosStop BOOL	Positioning: Cancel travelling	
	TRUE	Stop positioning From version 02.00.00, more travel requests will be inhibited ("PosExecute" will be blocked).

Designator	Data type		Information/possible settings
wPosProfileMode	WORD		Override of the positioning mode set in the profile data • Via this input, an override of the positioning mode parameterised in C01300/1...15 for the selected profile is possible. • The value set in C01300/1...15 is not overwritten in this case. • Only bit 0 ... bit 3 of *wPosMode* are evaluated. • If *wPosMode* = 0 is selected, the positioning mode set in C01300/1...15 is used.
		0	Positioning mode = setting in C01300/1...15
		1	Absolute (shortest path)
		2	Continuous
		3	Relative
		4	absolute (Cw)
		5	absolute (Ccw)
		8	Absolute (shortest path) to TP
		9	Continuous to TP
		10	Relative to TP
		11	Absolute (Cw) on TP
		12	Absolute (Ccw) on TP
			All other possible settings are reserved for future extensions!
wPosProfileNo	WORD		Stipulation of the profile to be executed • Optionally as a data word or binary coded.
bPosProfileNo_1 ... bPosProfileNo_8	BOOL		
dnPosProfilePosition_p	DINT		Selection of the target position in [increments] • The mode for calculating the position is selected in C01296/1.
wPosProfileUnitsLW wPosProfileUnitsHW	WORD		Selection of the target position in [units] • *wPosProfileUnitsLW* = Low word, *wPosProfileUnitsHW* = High word • The mode for calculating the position is selected in C01296/1.
GP: GeneralPurpose The following inputs are interconnected with logic/arithmetic functions on application level for free usage. ▸ "GeneralPurpose" functions			
nGPAnalogSwitchIn1_a nGPAnalogSwitchIn2_a	INT		Analog switch: Input signals • The input signal selected via the selection input *bGPAnalogSwitchSet* is output at output *nGPAnalogSwitchOut_a*.
bGPAnalogSwitchSet	BOOL		Analog switch: Selection input
		FALSE	*nGPAnalogSwitchOut_a* = *nGPAnalogSwitchIn1_a*
		TRUE	*nGPAnalogSwitchOut_a* = *nGPAnalogSwitchIn2_a*
nGPArithmetikIn1_a nGPArithmetikIn2_a	INT		Arithmetic: Input signals • The arithmetic function is selected in C00338. • The result is output at output *nGPArithmetikOut_a*.
nGPMulDivIn_a	INT		Multiplication/Division: Input signal • The factor for the multiplication can be set in C00699/1 (numerator) and C00699/2 (denominator). • The result is output at output *nGPMulDivOut_a*.
bGPDigitalDelayIn	BOOL		Binary delay element: Input signal • The on-delay can be set in C00720/1. • The off-delay can be set in C00720/2. • The time-delayed input signal is output at output *bGPDigitalDelayOut*.

Designator Data type	Information/possible settings
bGPLogicIn1 bGPLogicIn2 bGPLogicIn3 BOOL	Binary logic: Input signals • The logic operation is selected in C00820. • The result is output at output *bGPLogicOut*.
nGPCompareIn1_a nGPCompareIn2_a INT	Analog comparison: Input signals • The comparison operation is selected in C00680. • Hysteresis and window size can be set in C00680 and C00682. • If the comparison statement is true, the output *bGPCompareOut* will be set to TRUE.
bGPDFlipFlop_InD bGPDFlipFlop_InClk bGPDFlipFlop_InClr BOOL	D-FlipFlop: Input signals • Data, clock and reset input
bGPCounter1ClkUp BOOL	Counter: Clock input • With each edge, the module counts up by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.
bGPCounter1ClkDown BOOL	Counter: Clock input • With each edge, the module counts down by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.
bGPCounter1Load BOOL	Counter: Loading input • The input has the highest priority. TRUE: Accept the starting value *wGPCounter1LdVal*.
wGPCounter1LdVal WORD	Counter: Starting value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.
wGPCounter1CmpVal WORD	Counter: Comparison value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.
Free inputs The following inputs can freely be interconnected on the application level. The signals can be transferred from the I/O level to the application level via these inputs.	
bFreeIn1 ... bFreeIn8 BOOL	Free inputs for digital signals
wFreeIn1 ... wFreeIn4 WORD	Free inputs for 16-bit signals
dnFreeIn1_p ... dnFreeIn2_p DINT	Free inputs for 32-bit signals

outputs

Designator Data type	Value/meaning
wDriveControlStatus WORD	Status word of the inverter • The status word contains information on the currents status of the inverter. • See the "wDeviceStatusWord status word" subchapter of the chapter on device control for a detailed description of the bit assignment.
wStateDetermFailNoLow WORD	Display of the status determining error (LOW word)
wStateDetermFailNoHigh WORD	Display of the status determining error (HIGH word)
bDriveFail BOOL	TRUE: Inverter in error status. • "Fault" device status is active.

Designator / Data type		Value/meaning
bDriveReady (BOOL)	TRUE	Inverter is ready for operation. • "SwitchedOn" device status is active. • The drive is in this device status if the DC bus voltage is applied and the inverter is still inhibited by the user (controller inhibit).
bCInhActive (BOOL)	TRUE	Controller inhibit is active.
bQSPIsActive (BOOL)	TRUE	Quick stop is active.
bSpeedCcw (BOOL)	Current direction of rotation	
	FALSE	Clockwise rotation (Cw)
	TRUE	Direction of rotation to the left (Ccw)
bSpeedActCompare (BOOL)	Result of the speed comparison (detection of speed=0)	
	TRUE	During open-loop operation: Speed setpoint < Comparison value (C00024)
		During closed-loop operation: Actual speed value < Comparison value (C00024)
bImaxActive (BOOL)	"Current setpoint inside the limitation" status signal	
	TRUE	The current setpoint is internally limited (the inverter operates at the maximum current limit).
bSpeedSetReached (BOOL)	Status signal "setpoint = 0"	
	TRUE	Speed setpoint from the ramp function generator = 0
nMotorCurrent_a (INT)	Current stator current/effective motor current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	
nMotorSpeedSet_a (INT)	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)	
nMotorSpeedAct_a (INT)	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)	
nMotorTorqueAct_a (INT)	Actual torque • In the "VFC (+encoder)" operating mode of the motor control, this value is determined from the current motor current and corresponds to the actual torque only by approximation. • Scaling: 16384 ≡ 100 % M_{max} (C00057)	
nDCVoltage_a (INT)	Current DC-bus voltage • Scaling: 16384 ≡ 1000 V	
nMotorVoltage_a (INT)	Current motor voltage/inverter output voltage • Scaling: 16384 ≡ 1000 V	
MCK basic functions		
bMBrakeReleaseOut (BOOL)	Holding brake control: Trigger signal for the holding brake control switching element via a digital output • Use bit 0 in C02582 to activate inverted switching element triggering.	
	FALSE	Apply brake.
	TRUE	Release brake.
bMBrakeReleased (BOOL)	Holding brake control: "Brake released" considering the brake release time • When the holding brake is triggered to close, *bMBrakeReleased* is immediately set to FALSE even if the brake closing time has not yet elapsed!	
	TRUE	Brake released (after the brake release time has expired).
wMckState1 wMckState2 (WORD)	Output of the MCK status words • For a detailed description of each status bit see chapter "MCK status word".	
wMckActOperationMode (WORD)	Active setpoint-generating state of the MCK. • Bit B0 ... B3 contain the information of the MCK status word. • Bits B4 ... B15 are fixed at "0".	

Designator Data type	Value/meaning	
bHomeDone BOOL	TRUE	Homing has been executed.
bHomePosAvailable BOOL	TRUE	Home position is known.
bProfileDone BOOL	TRUE	Target position from the profile has been approached.
bProfileBusy BOOL	TRUE	Profile positioning is active.
bAccelerating BOOL	TRUE	Acceleration phase active.
bConstantDuty BOOL	TRUE	Constant phase active.
bDecelerating BOOL	TRUE	Braking phase active.
bDwellTime BOOL	TRUE	Settling in target position is active
bInTarget BOOL	TRUE	Target position (actual value) is in the target window.
wActProfileNo WORD	Current profile number • Bit B0 ... B7 contain the information of the MCK status word. • Bits B8 ... B15 are fixed at "0".	
wActPosMode WORD	Current positioning mode • Bit B0 ... B3 contain the information of the MCK status word. • Bits B4 ... B15 are fixed at "0".	
dnTargetPos_p DINT	Target position in [increments] • 65535 ≡ 1 revolution of the motor shaft	
dnSetPos_p DINT	Absolute position setpoint	
dnPosAct_p DINT	Current position of the motor shaft in [increments]	
dnDeltaPosAct_p DINT	Current following error in [increments] • Following error = Difference between set position and actual position	
wPosOutUnitsLW wPosOutUnitsHW WORD	Output of the target position in [units] • *wPosOutUnitsLW* = Low-Word, *wPosOutUnitsHW* = High-Word • The mode for calculating the position is selected in C01296/2.	

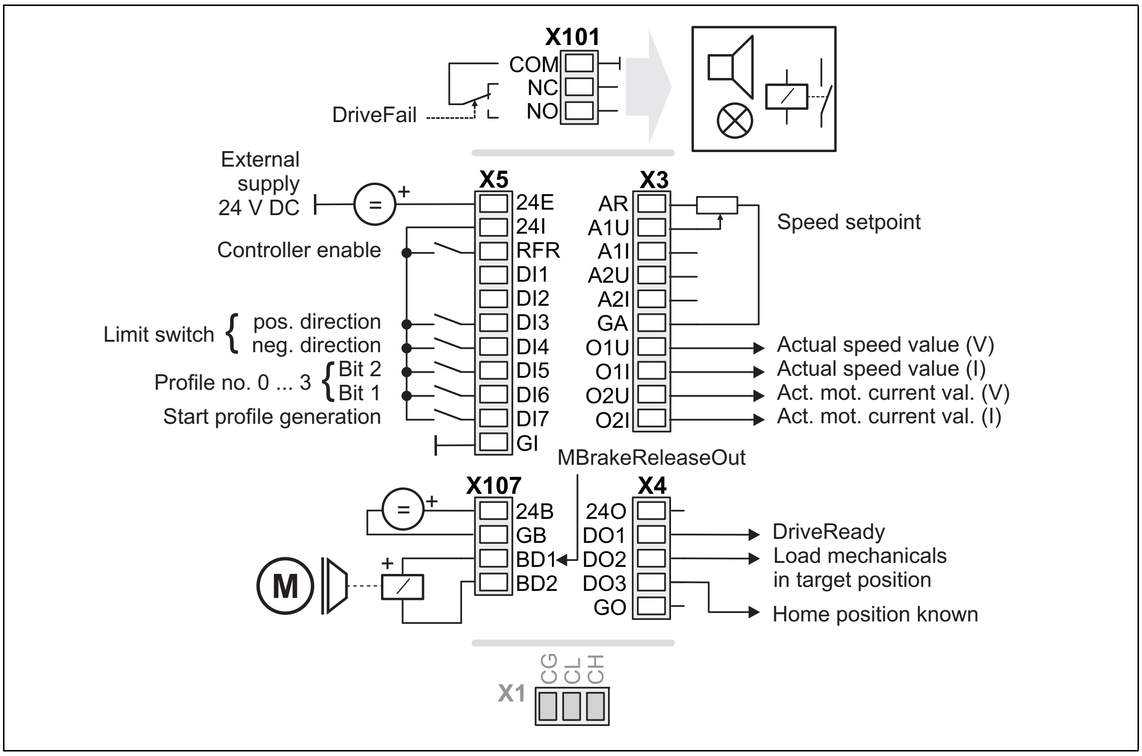
Designator Data type	Value/meaning
GP: GeneralPurpose The following outputs are interconnected with logic/arithmetic functions on application level for free usage. "GeneralPurpose" functions	
nGPAnalogSwitchInOut_a INT	Analog switch: Output signal
nGPArithmetikOut_a INT	Arithmetic: Output signal
nGPMulDivOut_a INT	Multiplication/Division: Output signal
bGPDigitalDelayOut BOOL	Binary delay element: Output signal
bGPLogicOut BOOL	Binary logic: Output signal
bGPCompareOut BOOL	Analog comparison: Output signal
bGPSignalOut1 ... bGPSignalOut4 BOOL	Binary signal monitor: Output signals • The signal sources to be output are selected in C00411/1...4. • A bit coded inversion of the output signals can be parameterised in C00412.
nGPSignalOut1_a ... nGPSignalOut4_a BOOL	Analog signal monitor: Output signals • The signal sources to be output are selected in C00410/1...4. • Gain and offset for each output signal can be parameterised in C00413/1...8.
bGPDFlipFlop_Out BOOL	D-FlipFlop: Output signal
bGPDFlipFlop_NegOut BOOL	D-FlipFlop: Negated output signal
bGPCounter1Equal BOOL	Counter: Status signal "Comparison value reached" TRUE: Comparison value *wGPCounter1CmpVal* reached.
wGPCounter1Out WORD	Counter: Counter content • Internal limitation to ± 32767 • The most significant bit determines the sign!
Free outputs The following outputs can freely be interconnected on the application level. The signals from the application level can be transferred to the I/O level via these outputs.	
bFreeOut1 ... bFreeOut8 BOOL	Free outputs for digital signals
wFreeOut1 ... wFreeOut4 WORD	Free outputs for 16-bit signals
dnFreeOut1_p dnFreeOut2_p WORD	Free outputs for 32-bit signals

8.4.3 Terminal assignment of the control modes

The following comparison provides information about which inputs/outputs of the application block **LA_TabPos** are interconnected to the digital and analog input/output terminals of the inverter in the different control modes.

<table>
<tr><th rowspan="2"></th><th colspan="8">Control mode (C00007)</th></tr>
<tr><th>10: Terminals 0</th><th>12: Terminals 2</th><th>14: Terminals 11</th><th>16: Terminal 16</th><th>20: Keypad</th><th>21: PC</th><th>30: CAN</th><th>40: MCI</th></tr>
<tr><td colspan="9">Digital input terminals</td></tr>
<tr><td>X5/RFR</td><td colspan="8">Controller enable / Reset of error message
bFailReset</td></tr>
<tr><td>X5/DI1</td><td>-</td><td>-</td><td>Positive limit switch
bLimitSwitchPos</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>X5/DI2</td><td>-</td><td>-</td><td>Negative limit switch
bLimitSwitchNeg</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>X5/DI3</td><td>Positive limit switch
bLimitSwitchPos</td><td>-</td><td>-</td><td>Manual jog
in positive direction
bManJogPos</td><td>-</td><td>-</td><td colspan="2">Positive limit switch
bLimitSwitchPos</td></tr>
<tr><td>X5/DI4</td><td>Negative limit switch
bLimitSwitchNeg</td><td colspan="2">Positioning profile, array bit (valency 4)
bPosProfileNo_4</td><td>Manual jog
in negative direction
bManJogNeg</td><td>-</td><td>-</td><td colspan="2">Negative limit switch
bLimitSwitchNeg</td></tr>
<tr><td>X5/DI5</td><td colspan="4">Positioning profile, array bit (valency 2)
bPosProfileNo_2</td><td>-</td><td>-</td><td colspan="2">Home position
bHomeMark</td></tr>
<tr><td>X5/DI6</td><td colspan="4">Positioning profile, array bit (valency 1)
bPosProfileNo_1</td><td>-</td><td>-</td><td colspan="2">Quick stop
bSetQuickstop</td></tr>
<tr><td>X5/DI7</td><td colspan="4">Start of profile generation
bPosExecute</td><td>-</td><td>-</td><td>Status word - bit 7
LP_CanOut1: bState_B7</td><td>Status word - bit 7
LP_MciOut: bState_B7</td></tr>
<tr><td colspan="9">Analog input terminals</td></tr>
<tr><td>X3/A1U</td><td colspan="4" rowspan="2">Main speed setpoint
nMainSetValue_a
10 V ≡ 100 % reference speed (C00011)</td><td rowspan="2">-</td><td rowspan="2">-</td><td colspan="2" rowspan="2">Main speed setpoint
nMainSetValue_a
10 V ≡ 100 % reference speed (C00011)</td></tr>
<tr><td>X3/A1I</td></tr>
<tr><td>X3/A2U</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>X3/A2I</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td colspan="9">Digital output terminals</td></tr>
<tr><td>X4/DO1</td><td colspan="8">Status "Drive is ready" (bDriveReady)</td></tr>
<tr><td>X4/DO2</td><td colspan="8">Status "Target position (actual value) is in the target window" (bInTarget)</td></tr>
<tr><td>X4/DO3</td><td colspan="6">Status "Home position is known"(bHomePosAvailable)</td><td colspan="2">Status "Maximum current limit" (bImaxActive)</td></tr>
<tr><td>X107/BD1, BD2</td><td colspan="8">Control of the holding brake (bMBrakeReleaseOut)</td></tr>
<tr><td>X101/COM, NO</td><td colspan="8">Status "Error is pending" (bDriveFail)</td></tr>
<tr><td colspan="9">Analog output terminals</td></tr>
<tr><td>X3/O1U</td><td colspan="8" rowspan="2">Actual speed value
nMotorSpeedAct_a
10 V ≡ 100 % reference speed (C00011)</td></tr>
<tr><td>X3/O1I</td></tr>
<tr><td>X3/O2U</td><td colspan="8" rowspan="2">Actual motor current
nMotorCurrent_a
10V ≡ 100% of I_{max_mot} (C00022)</td></tr>
<tr><td>X3/O2I</td></tr>
</table>

8.4.3.1 Terminals 0

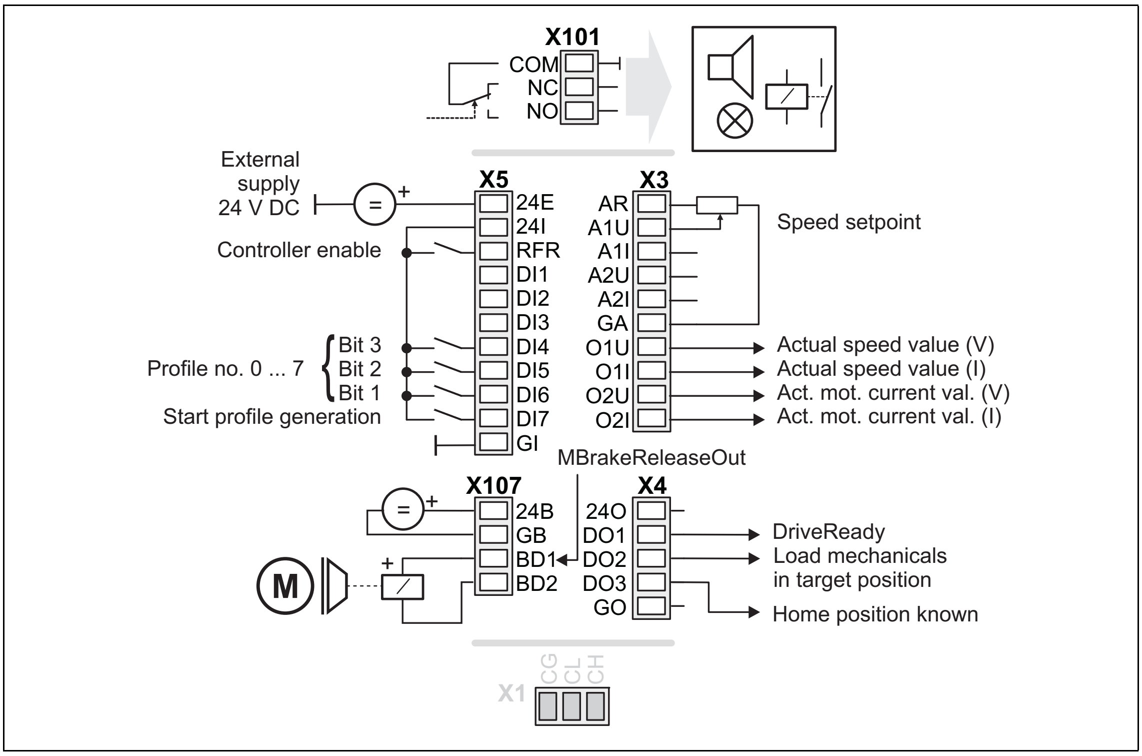


Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		**X3/A1U**	LA_TabPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	-		**X3/A1I**	
X5/DI2	-		**X3/A2U**	-
X5/DI3	LA_TabPos.bLimitSwitchPos		**X3/A2I**	-
X5/DI4	LA_TabPos.bLimitSwitchNeg		**X3/O1U**	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	LA_TabPos.bPosProfileNo_2		**X3/O1I**	
X5/DI6	LA_TabPos.bPosProfileNo_1		**X3/O2U**	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	LA_TabPos.bPosExecute		**X3/O2I**	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		**X4/DO1**	LA_TabPos.bDriveReady
X107/BD2	-		**X4/DO2**	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bHomePosAvailable

When the profile is defined, the operating mode in the Lenze setting is changed simultaneously:

bPosProfileNo_2 (DI5)	bPosProfileNo_1 (DI6)	Selected profile	Activation of operating mode
FALSE	FALSE	0	Speed follower
FALSE	TRUE	1	Homing
TRUE	FALSE	2	Manual jog
TRUE	TRUE	3	Positioning

8.4.3.2 Terminals 2

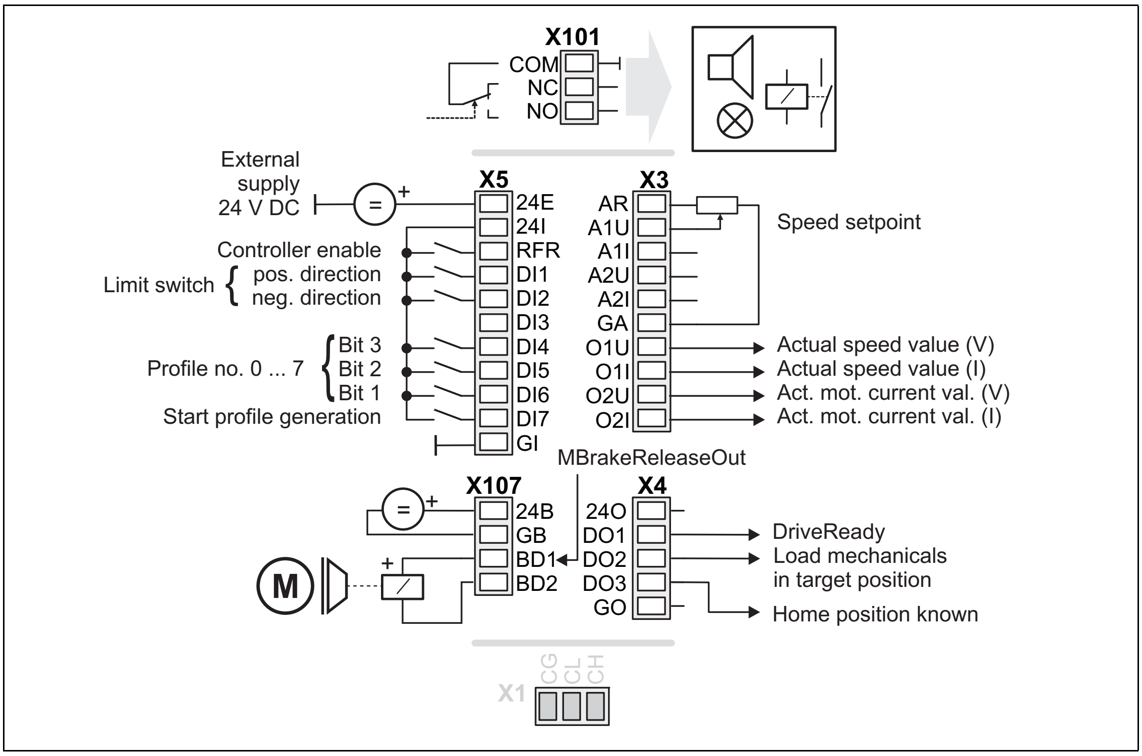


Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		**X3/A1U**	LA_TabPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	-		**X3/A1I**	
X5/DI2	-		**X3/A2U**	-
X5/DI3	-		**X3/A2I**	-
X5/DI4	LA_TabPos.bPosProfileNo_4		**X3/O1U**	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	LA_TabPos.bPosProfileNo_2		**X3/O1I**	
X5/DI6	LA_TabPos.bPosProfileNo_1		**X3/O2U**	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	LA_TabPos.bPosExecute		**X3/O2I**	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		**X4/DO1**	LA_TabPos.bDriveReady
X107/BD2	-		**X4/DO2**	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bHomePosAvailable

When the profile is defined, the operating mode in the Lenze setting is changed simultaneously:

bPosProfileNo_4 **(DI4)**	bPosProfileNo_2 **(DI5)**	bPosProfileNo_1 **(DI6)**	**Selected profile**	**Activation of operating mode**
FALSE	FALSE	FALSE	0	Speed follower
FALSE	FALSE	TRUE	1	Homing
FALSE	TRUE	FALSE	2	Manual jog
FALSE	TRUE	TRUE	3	Positioning
	...		...	
TRUE	TRUE	TRUE	7	

8.4.3.3 Terminals 11

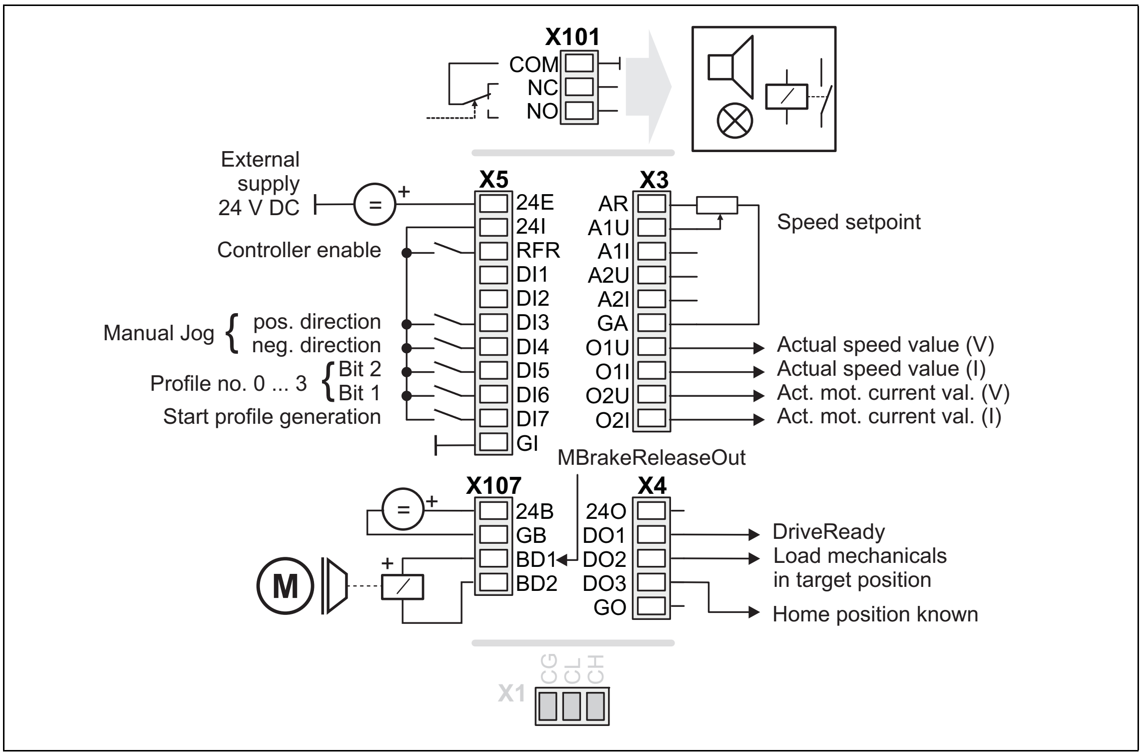


Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		**X3/A1U**	LA_TabPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	LA_TabPos.bLimitSwitchPos		**X3/A1I**	
X5/DI2	LA_TabPos.bLimitSwitchNeg		**X3/A2U**	-
X5/DI3	-		**X3/A2I**	-
X5/DI4	LA_TabPos.bPosProfileNo_4		**X3/O1U**	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	LA_TabPos.bPosProfileNo_2		**X3/O1I**	
X5/DI6	LA_TabPos.bPosProfileNo_1		**X3/O2U**	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	LA_TabPos.bPosExecute		**X3/O2I**	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		**X4/DO1**	LA_TabPos.bDriveReady
X107/BD2	-		**X4/DO2**	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bHomePosAvailable

When the profile is defined, the operating mode in the Lenze setting is changed simultaneously:

bPosProfileNo_4 **(DI4)**	bPosProfileNo_2 **(DI5)**	bPosProfileNo_1 **(DI6)**	**Selected profile**	**Activation of operating mode**
FALSE	FALSE	FALSE	0	Speed follower
FALSE	FALSE	TRUE	1	Homing
FALSE	TRUE	FALSE	2	Manual jog
FALSE	TRUE	TRUE	3	Positioning
	...		...	
TRUE	TRUE	TRUE	7	

8.4.3.4 Terminal 16

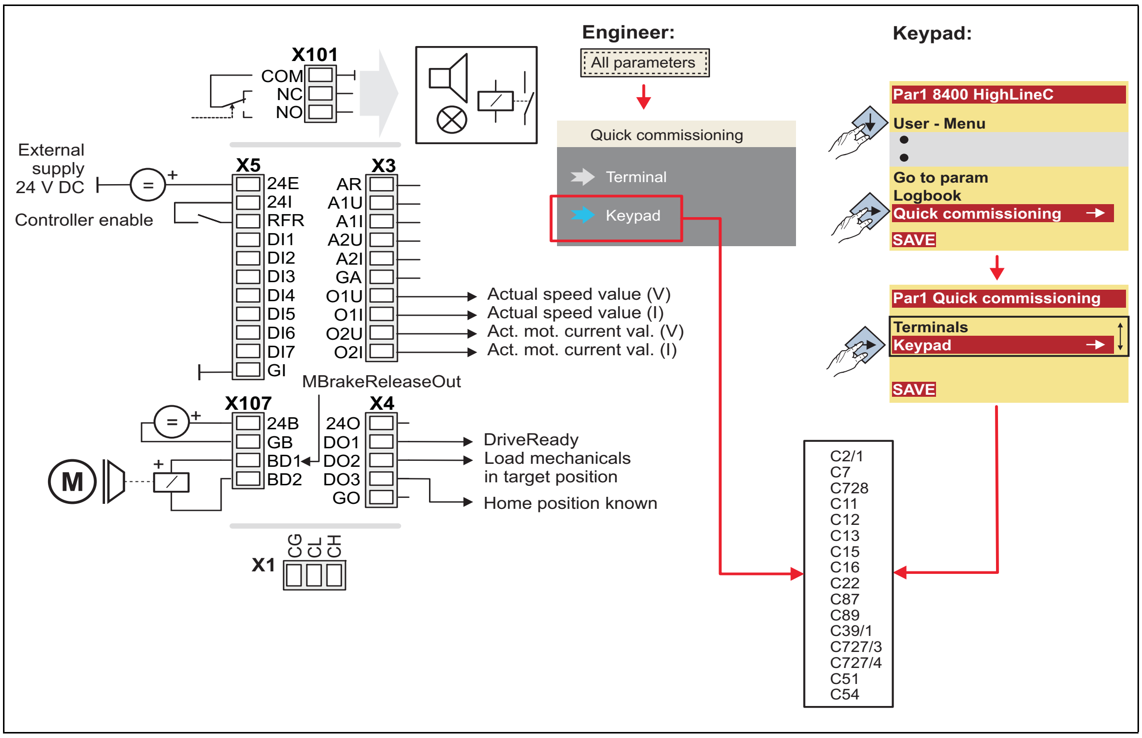


Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		**X3/A1U**	LA_TabPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	-		**X3/A1I**	
X5/DI2	-		**X3/A2U**	-
X5/DI3	LA_TabPos.bManJogPos		**X3/A2I**	-
X5/DI4	LA_TabPos.bManJogNeg		**X3/O1U**	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	LA_TabPos.bPosProfileNo_2		**X3/O1I**	
X5/DI6	LA_TabPos.bPosProfileNo_1		**X3/O2U**	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	LA_TabPos.bPosExecute		**X3/O2I**	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		**X4/DO1**	LA_TabPos.bDriveReady
X107/BD2	-		**X4/DO2**	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bHomePosAvailable

When the profile is defined, the operating mode in the Lenze setting is changed simultaneously:

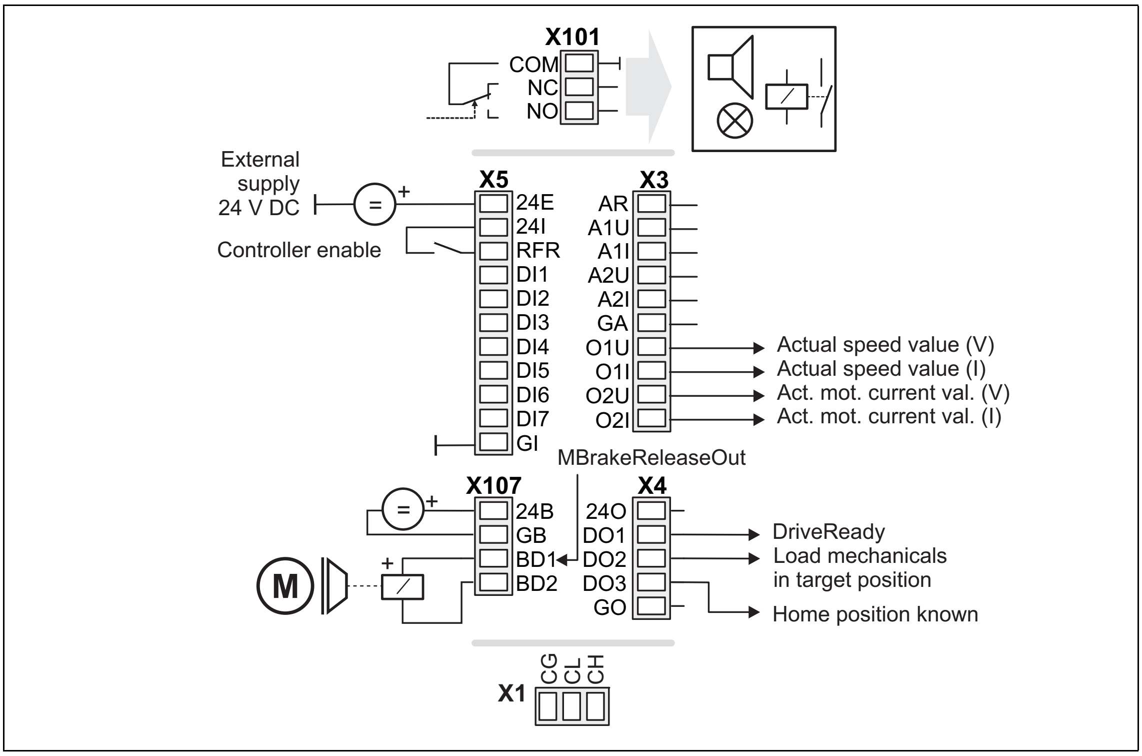
bPosProfileNo_2 (DI5)	bPosProfileNo_1 (DI6)	Selected profile	Activation of operating mode
FALSE	FALSE	0	Speed follower
FALSE	TRUE	1	Homing
TRUE	FALSE	2	Manual jog
TRUE	TRUE	3	Positioning

8.4.3.5 Keypad



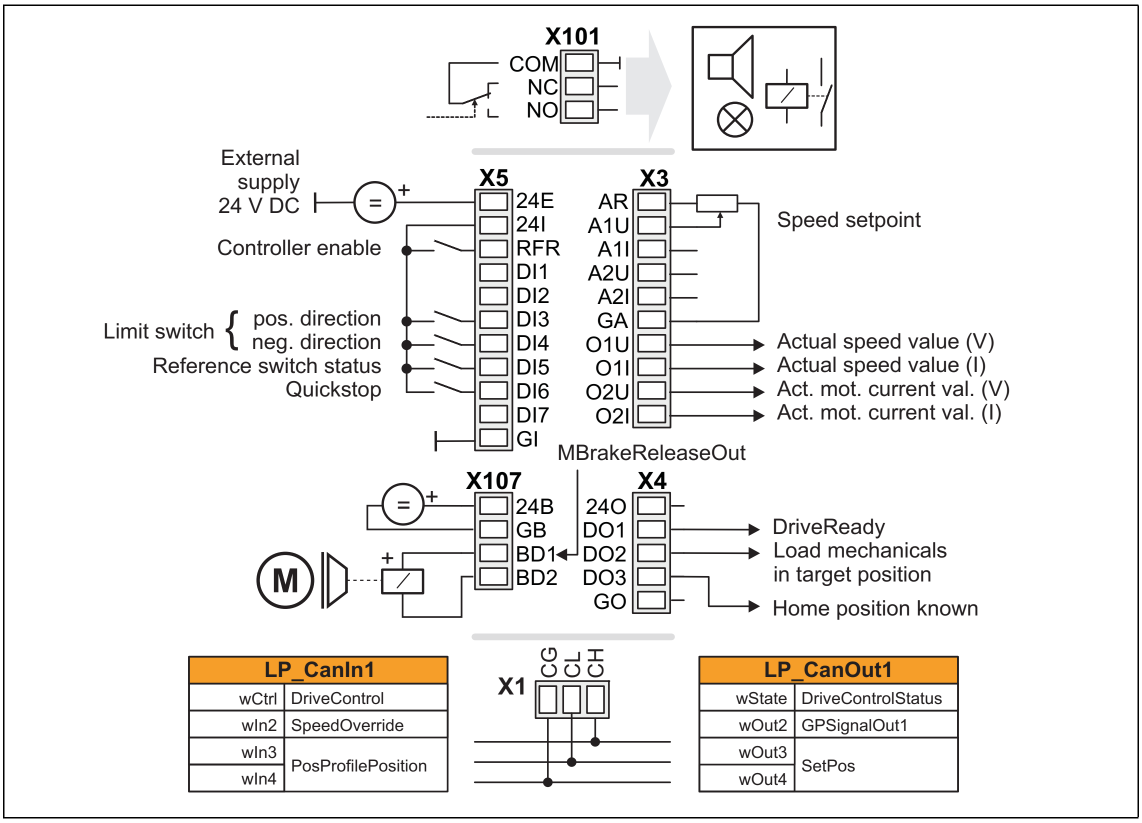
Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		X3/A1U	LA_TabPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	-		X3/A1I	
X5/DI2	-		X3/A2U	-
X5/DI3	-		X3/A2I	-
X5/DI4	-		X3/O1U	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	-		X3/O1I	
X5/DI6	-		X3/O2U	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	-		X3/O2I	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		X4/DO1	LA_TabPos.bDriveReady
X107/BD2	-		X4/DO2	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bHomePosAvailable

8.4.3.6 PC



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		**X3/A1U**	-
X5/DI1	-		**X3/A1I**	-
X5/DI2	-		**X3/A2U**	-
X5/DI3	-		**X3/A2I**	-
X5/DI4	-		**X3/O1U**	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	-		**X3/O1I**	
X5/DI6	-		**X3/O2U**	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	-		**X3/O2I**	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		**X4/DO1**	LA_TabPos.bDriveReady
X107/BD2	-		**X4/DO2**	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bHomePosAvailable

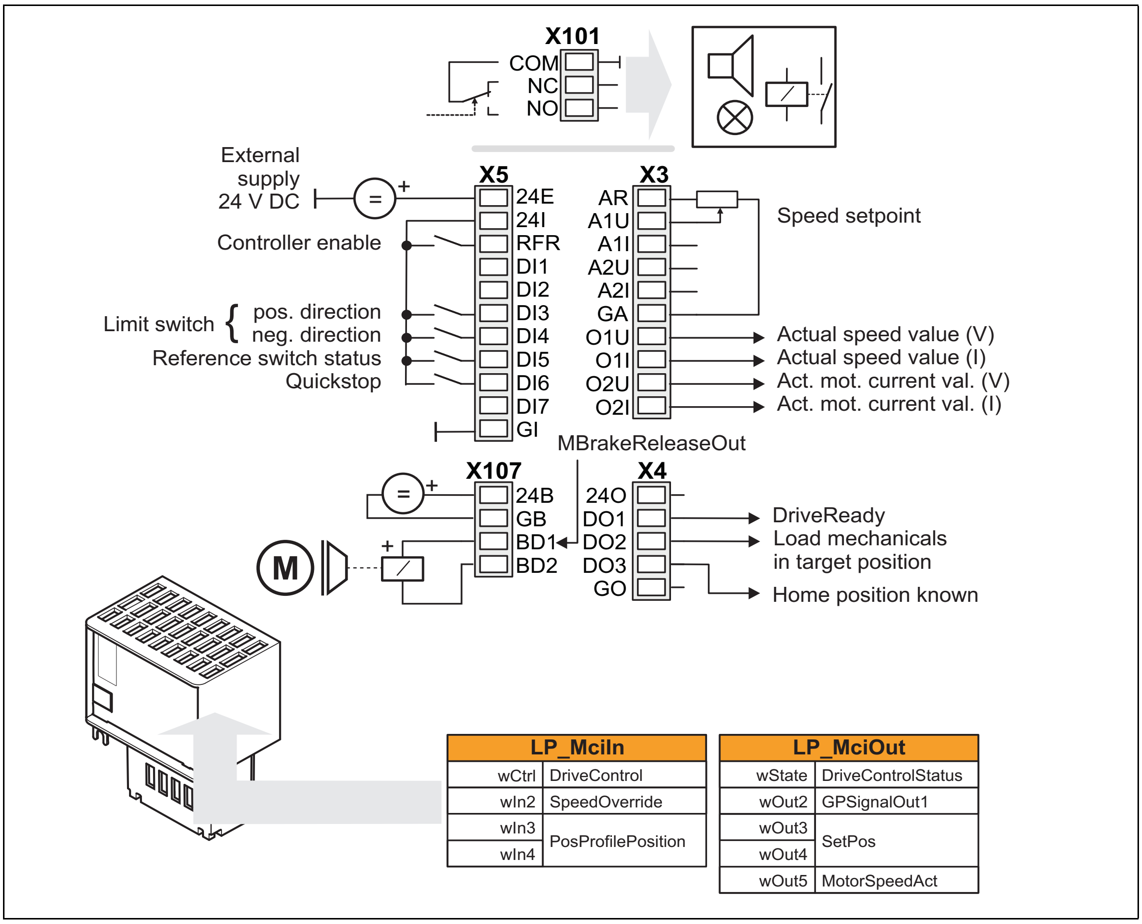
8.4.3.7 CAN



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		**X3/A1U**	LA_TabPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	-		**X3/A1I**	
X5/DI2	-		**X3/A2U**	-
X5/DI3	LA_TabPos.bLimitSwitchPos		**X3/A2I**	-
X5/DI4	LA_TabPos.bLimitSwitchNeg		**X3/O1U**	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	LA_TabPos.bHomeMark		**X3/O1I**	
X5/DI6	LA_TabPos.bSetQuickstop		**X3/O2U**	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	LP_CanOut1: bState_B7		**X3/O2I**	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		**X4/DO1**	LA_TabPos.bDriveReady
X107/BD2	-		**X4/DO2**	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bImaxActive

▸ Process data assignment for fieldbus communication (📖 538)

8.4.3.8 MCI



Connection	Assignment		Connection	Assignment
X101/NC-NO	LA_TabPos.bDriveFail			
X5/RFR	LA_TabPos.bFailReset		**X3/A1U**	LA_TabPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	-		**X3/A1I**	
X5/DI2	-		**X3/A2U**	-
X5/DI3	LA_TabPos.bLimitSwitchPos		**X3/A2I**	-
X5/DI4	LA_TabPos.bLimitSwitchNeg		**X3/O1U**	LA_TabPos.nMotorSpeedAct_a 10 V ≡ 100 % reference speed (C00011)
X5/DI5	LA_TabPos.bHomeMark		**X3/O1I**	
X5/DI6	LA_TabPos.bSetQuickstop		**X3/O2U**	LA_TabPos.nMotorCurrent_a 10V ≡ 100% of I_{max_mot} (C00022)
X5/DI7	LP_MciOut: bState_B7		**X3/O2I**	
X107/BD1	LA_TabPos.bMBrakeReleaseOut		**X4/DO1**	LA_TabPos.bDriveReady
X107/BD2	-		**X4/DO2**	LA_TabPos.bInTarget
			X4/DO3	LA_TabPos.bImaxActive

▸ Process data assignment for fieldbus communication (📖 538)

8.4.4 Process data assignment for fieldbus communication

The fieldbus communication is connected (preconfigured) to the previously selected technology application by selecting the corresponding control mode in C00007:

- "30: CAN" for the connection to the system bus (CAN)
- "40: MCI" for the connection to a plugged-on communication module (e.g. PROFIBUS)

The assignment of the process data words depends only on the application, not on the bus system used:

Input words	Name	Assignment
Word 1	DriveControl	Control word • For bit assignment see the table below.
Word 2	SpeedOverride	Value for speed override • Percentage multiplier for the currently active speed. • 16384 ≡ 100 % of the maximum traversing speed (display in C01211/1). • Values > 16384 are ignored. • If the override value is 0 %, the drive is brought to a standstill. ▸ Speed override (📖 698)
Word 3 and 4	PosProfilePosition	Selection of the target position in [increments] • The mode for calculating the position is selected in C01296/1. • 65535 [Increments] ≡ 1 motor revolution • The position resolution, i.e. how many increments form one unit, is displayed in C01205.
Words 5 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Control word	Name	Function
Bit 0	MckOperationMode_1	Binary-coded selection of the operating mode of the **Motion Control Kernel** • For a detailed description of the individual control bits, see chapter "MCK control word". (📖 586)
Bit 1	MckOperationMode_2	
Bit 2	MckOperationMode_4	
Bit 3	CINH	1 ≡ Inhibit inverter (controller inhibit): The inverter switches to the "SwitchedOn" device status. ▸ Enable/inhibit inverter (📖 113)
Bit 4	PosProfileNo_1	Selection of the profile number ▸ Stipulation of the profile to be executed (📖 681)
Bit 5	PosProfileNo_2	
Bit 6	PosProfileNo_4	
Bit 7	PosProfileNo_8	
Bit 8	PosExecute	1 ≡ Start travel job ▸ Positioning (📖 666)
Bit 9	EnableSpeedOverride	1 ≡ activate speed override ▸ Speed override (📖 698)
Bit 10	HomeSetPosition	1 ≡ Set home position ▸ Homing (📖 637)
Bit 11	FailReset	1 ≡ Reset error message ▸ Reset error message (📖 747)
Bit 12	ManJogNeg	Manual jog in positive/negative direction ▸ Manual jog (📖 658)
Bit 13	ManJogPos	
Bit 14	-	Free control bit 14 (not assigned, freely assignable)
Bit 15	-	Free control bit 15 (not assigned, freely assignable)

Output words	Name	Assignment
Word 1	DriveControlStatus	Status word • For bit assignment see the table below.
Word 2	GPSignalOut1	Analog signal monitor: Output signal 1 • The selection of the signal source to output is executed in C00410/1. • Gain and offset for the output signal can be parameterised in C00413/1 and C00413/2. • For a detailed functional description see the L_SignalMonitor_a FB.
Word 3 and 4	SetPos	Absolute position setpoint in [increments] • 65535 [Increments] ≡ 1 motor revolution • The position resolution, i.e. how many increments form one unit, is displayed in C01205.
Word 5	MotorSpeedAct	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011) • Only available in control mode "40: MCI".
Word 6 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Status word	Name	Status
Bit 0	DriveFail	1 ≡ Inverter in the error status • "Fault" device status is active.
Bit 1	GPSignalOut1	Binary signal monitor: Output signals 1 & 2 • The signal sources to be output are selected in C00411/1...2. • A bit coded inversion of the output signals can be parameterised in C00412. • For a detailed functional description see FB L_SignalMonitor_b.
Bit 2	GPSignalOut2	
Bit 3	CInhActive	1 ≡ Controller inhibit is active
Bit 4	DriveReady	1 ≡ Inverter is ready for operation • "SwitchedOn" device status is active. • The drive is in this device status if the DC bus voltage is applied and the inverter is still inhibited by the user (controller inhibit).
Bit 5	DigitalInput5	Signal from the digital input DI5
Bit 6	DigitalInput6	Signal from the digital input DI6
Bit 7	DigitalInput7	Signal from the digital input DI7
Bit 8	InTarget	1 ≡ Target position (actual value) is in the target window
Bit 9	ProfileBusy	1 ≡ Profile positioning is active
Bit 10	HomePosAvailable	1 ≡ Home position is known
Bit 11	SpeedActCompare	Result of the speed comparison (detection of speed=0) • In case of the "Open loop" operation: 1 ≡ Speed setpoint < comparison value (C00024) • For "Closed loop" operation: 1 ≡ actual speed value < comparison value (C00024)
Bit 12	DigitalInput4	Signal from the digital input DI4
Bit 13	DigitalInput3	Signal from the digital input DI3
Bit 14	QSPIsActive	1 ≡ Quick stop is active
Bit 15	-	Free status bit 15 (not assigned, freely assignable)

8.4.5 Setting parameters (short overview)

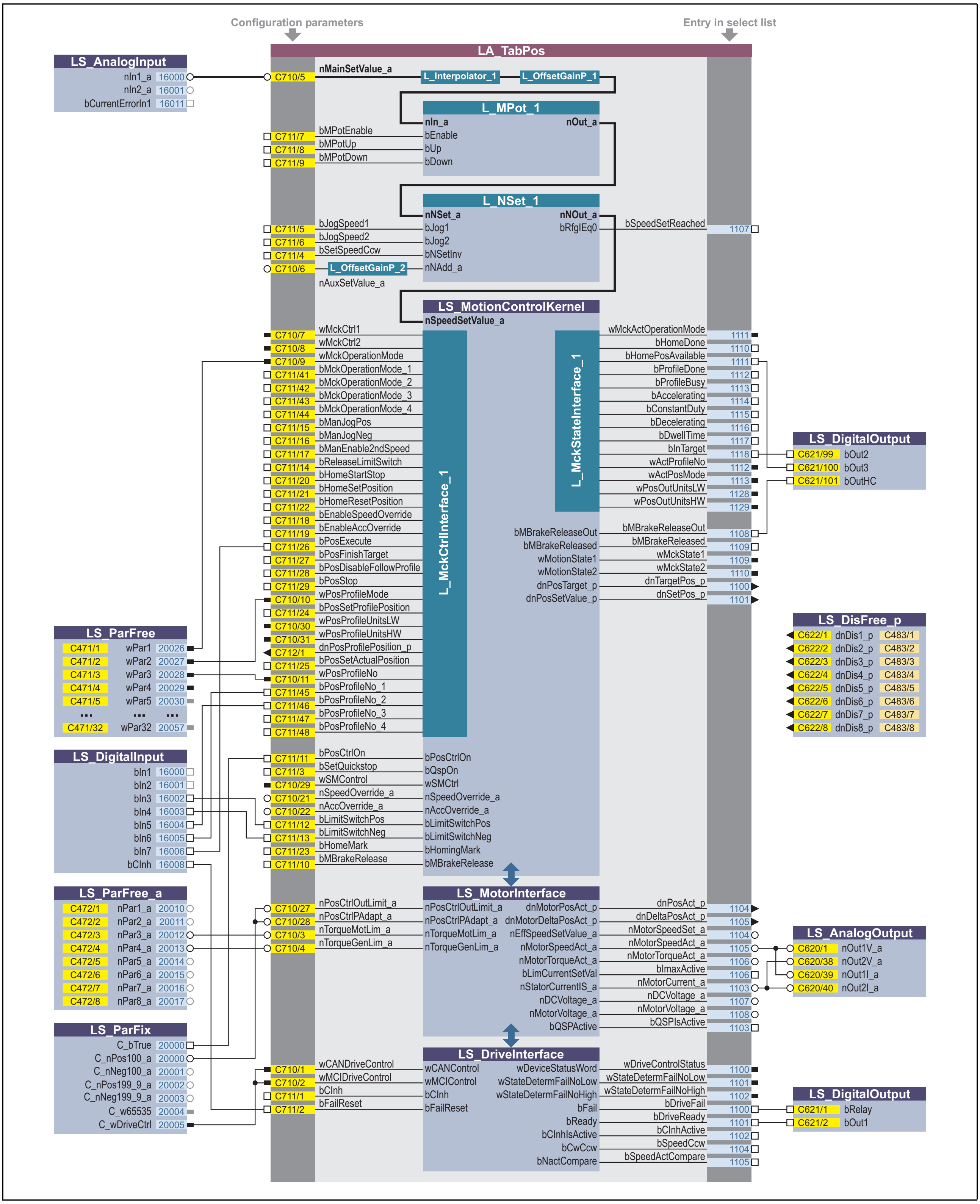
Parameters	Info	Lenze setting	
		Value	Unit
C00012	Accel. time - main setpoint	2.000	s
C00013	Decel. time - main setpoint	2.000	s
C00024	Comparison value N_Act	0.00	%
C00039/1	Preset setpoint 1	40.00	%
C00039/2	Preset setpoint 2	60.00	%
C00182	S-ramp time PT1	20.00	s
C00190	Setpoint arithmetic	0: NOut = NSet	
C00220	Accel. time - add. setpoint	0.000	s
C00221	Decel. time - add. setpoint	0.000	s
C00632/1	L_NSet_1: Blocking speed 1 max	0.00	%
C00632/2	L_NSet_1: Blocking speed 2 max	0.00	%
C00632/3	L_NSet_1: Blocking speed 3 max	0.00	%
C00633/1	L_NSet_1: Blocking speed 1 min	0.00	%
C00633/2	L_NSet_1: Blocking speed 2 min	0.00	%
C00633/3	L_NSet_1: Blocking speed 3 min	0.00	%
C00635	L_NSet_1: nMaxLimit	199.99	%
C00636	L_NSet_1: nMinLimit	-199.99	%
C00670	L_OffsetGainP_1: Gain	1.0000	
C00671	L_OffsetGainP_2: Gain	1.0000	
C00696	L_OffsetGainP_1: Offset	0.00	%
C00697	L_OffsetGainP_2: Offset	0.00	%
C00800	L_MPot_1: Upper limit	100.00	%
C00801	L_MPot_1: Lower limit	-100.00	%
C00802	L_MPot_1: Acceleration time	10.0	s
C00803	L_MPot_1: Deceleration time	10.0	s
C00804	L_MPot_1: Inactive fct.	0: Retain value	
C00805	L_MPot_1: Init fct.	0: Load last value	
C00806	L_MPot_1: Use	0: No	
C01297	Alternative function	Bit coded	
C01298/1	MCK operating mode at profile no. 0	1: Follower	
C01298/2	MCK operating mode at profile no. 1	2: Homing	
C01298/3	MCK operating mode at profile no. 2	3: ManualJog	
C01298/4	MCK operating mode at profile no. 3...15	4: Positioning	
C01299	MCKI: Status MCKInterface	-	

Related topics:

▶ "GeneralPurpose" functions (📖 571)

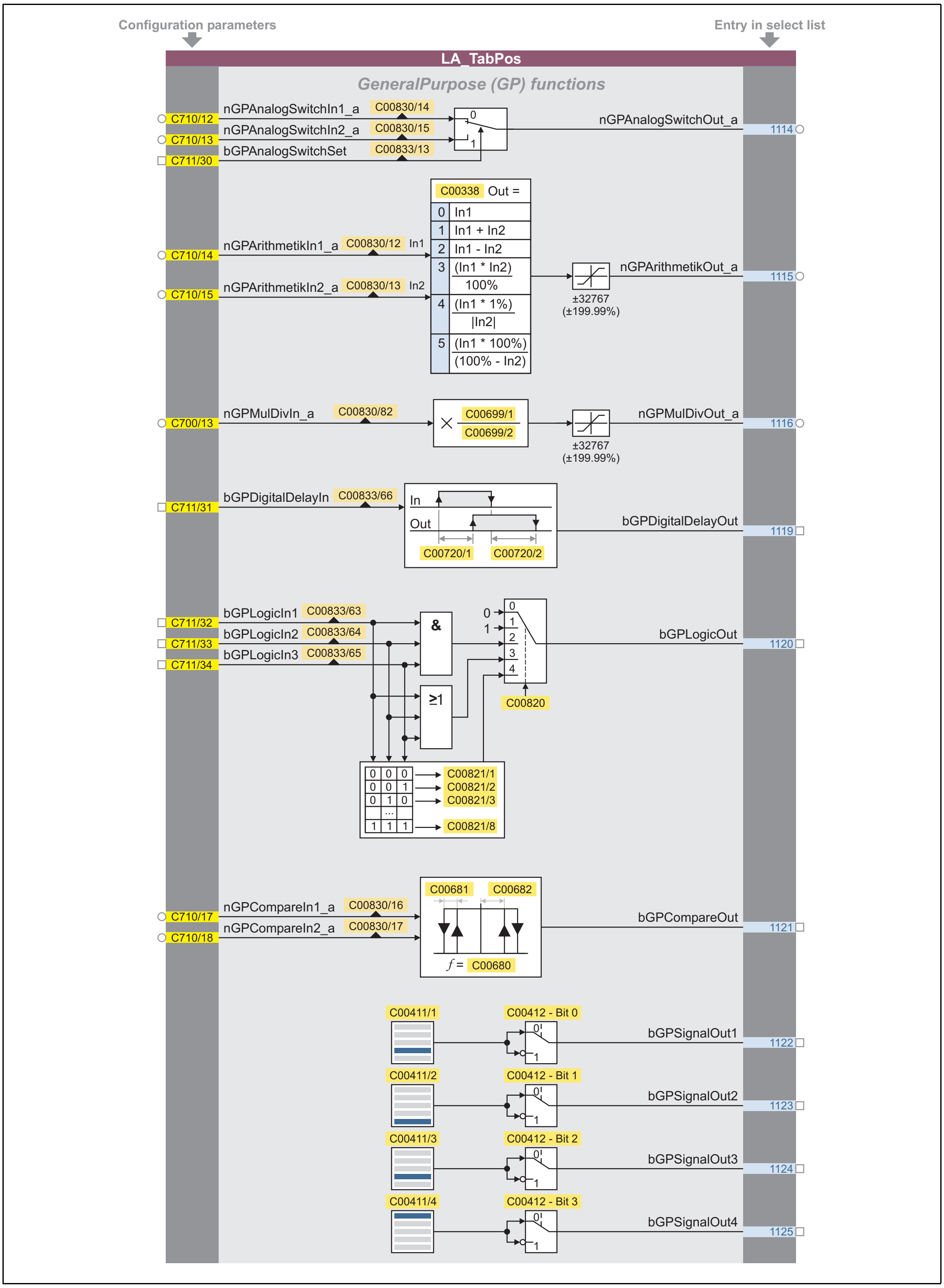
8.4.6 Configuration parameters

If required, the subcodes of C00710, C00711 and C00712 serve to change the pre-configured assignment of the application inputs:

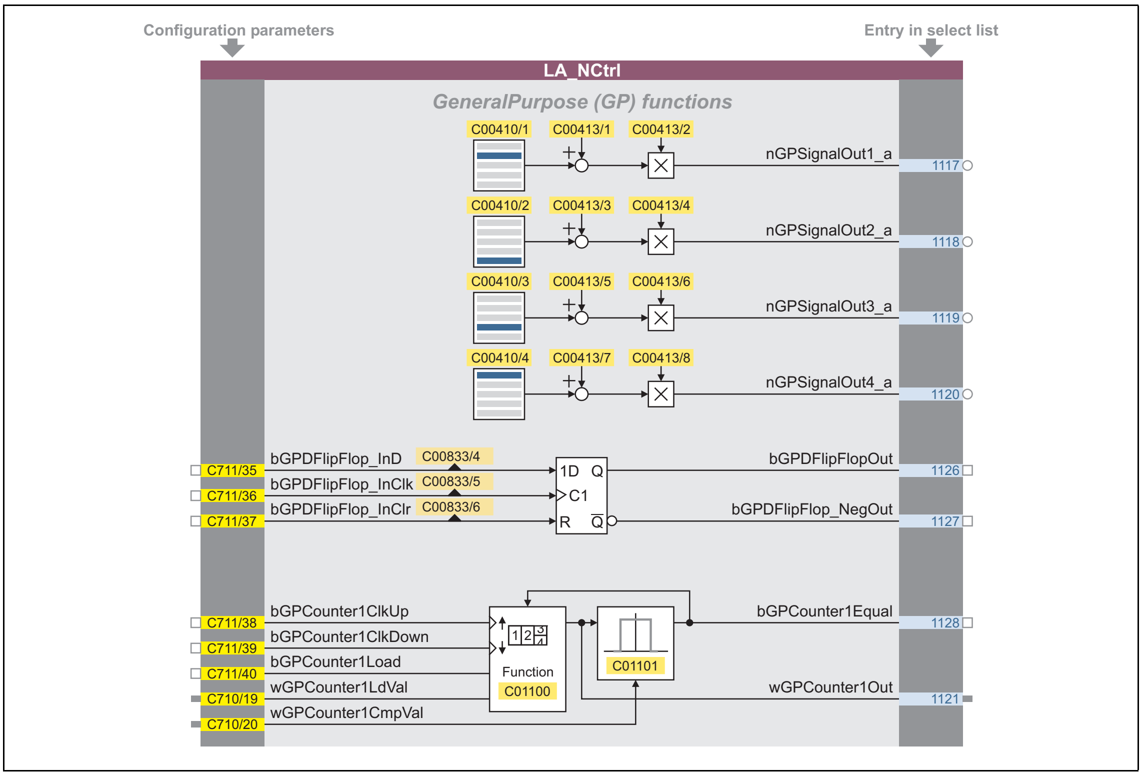


[8-14] Pre-assignment of the "Table positioning" application in the "Terminals 0" control mode

Configuration parameters for "GeneralPurpose" functions



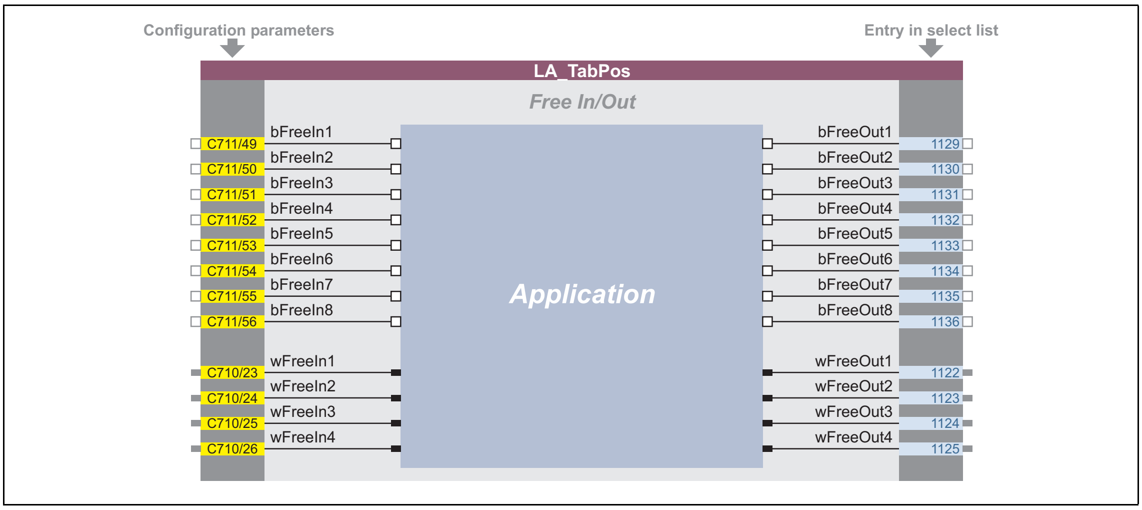
[8-15] "GeneralPurpose" functions



[8-16] "GeneralPurpose" functions (continuation)

Free inputs and outputs

These inputs can be freely interconnected in the application level. They can be used to transfer signals from the I/O level to the application level and vice versa.



[8-17] Free inputs/outputs

Related topics:

- User-defined terminal assignment (📖 445)
- "GeneralPurpose" functions (📖 571)

8.5 TA "Switch-off positioning"

The basic principle of this technology application is to travel to a switch-off sensor (e.g. a limit switch) in a speed-controlled manner and to stop as close as possible at this position. Unlike other positioning controls, the switch-off positioning neither has a position feedback nor calculates the path in advance. Thus, the accuracy that can be achieved depends on various factors such as the speed at which the switch-off sensor is advanced.

In addition, a pre-switch off can be implemented which requires a sufficient number of unassigned digital inputs on the inverter which can be used to connect other sensors for the additional stop positions. These sensors effect a reduction in speed before the last switch-off sensor is reached.

Product features

- Pre-configured control modes for terminals and bus control (with predefined process data connection to the fieldbus)
- Free configuration of input and output signals
- Offset, gain, and negation of main setpoint & additional setpoint
- Up to 15 fixed setpoints for speed and ramp time
- Adjustable setpoint ramp times
- Freely selectable, variable ramp shape
- Automatic holding brake control
- Quick stop (QSP) with adjustable ramp time
- Integrated, freely available "GeneralPurpose" functions:
  Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-flipflop
- Interface to the safety module (optional)
- Integration of encoder feedback
- Switch-off sensor management for the implementation of a pre-switch off

Decision criteria

Criteria	Switch-off positioning with constant load	Switch-off positioning with variable load
Operating mode	V/f characteristic without speed sensor. Alternatively for large breakaway torques: Use of a sensorless vector control (only applicable for horizontal movements).	
Limit switch evaluation	One limit switch is required per direction of movement. When the limit switch is reached, the drive is brought to a standstill led by the deceleration ramp or the QSP ramp.	One limit switch and an initiator are required for fast/slow changeover per direction of movement. When the initiator has been reached, the speed of the drive is reduced to a creeping speed (fixed setpoint 2) in a controlled way. When the limit switch is reached, the drive is brought to a standstill led by the deceleration ramp or the QSP ramp.

Criteria	Switch-off positioning with constant load	Switch-off positioning with variable load
Positioning accuracy at the motor shaft The positioning accuracy of the load depends, among other things, on the clearance and friction of the selected mechanics and has to be determined individually.	The ideal case is 5-10° at the motor shaft. Consider the influence of the motor temperature. In the case of a constant load, you can assume a good repeat accuracy during positioning. In the case of variable loads, you must take significant deviations into account.	5-10° at the motor shaft. As the positioning is executed in a creeping speed, a good repeat accuracy is reached even for variable loads.
Speed setting range	1 : 50, based on 50Hz and M_n	1 : 50, based on 50Hz and M_n
Typical applications	Switch-off positioning with constant load, e.g. travelling drive, roll-up door.	Switch-off positioning with variable load, e.g. travelling drive, conveying belt, hoists approaching a stop position.

System limits and exclusion criteria

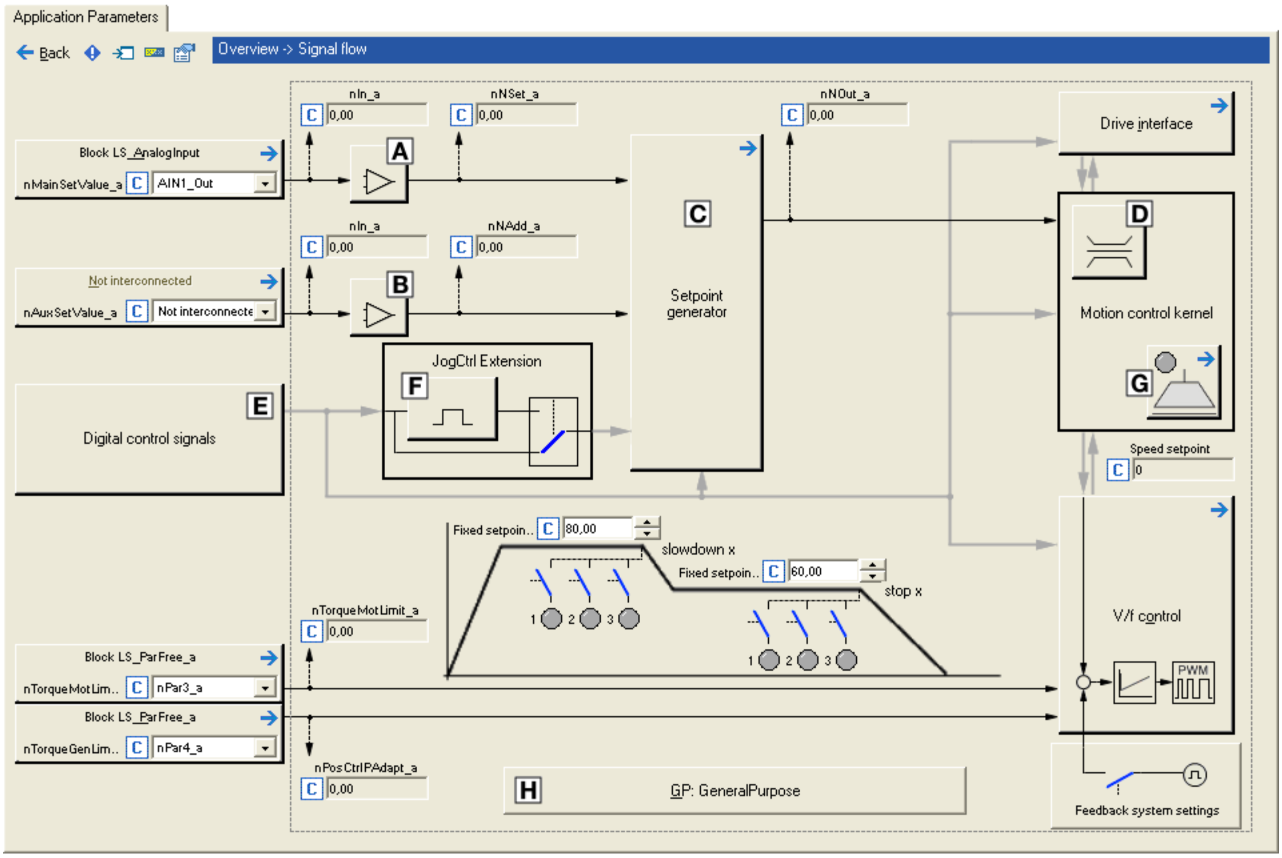
They result from the non-compliance with the decision criteria.

- Compared to systems with speed feedback, the positioning and repeat accuracy is reduced.
- Due to the mechanical hardware limit switches, this concept is only applicable for systems with only a few fixed positions. Changing the target position during the operation or the teaching is not possible.
- If necessary, additional functions like manual jog or homing must be realised externally, e.g. via a control.
- As the 8400 TopLine inverter does not meet safety-related functions except STO (Safe Torque Off), you must observe that all safety-related aspects are realised by the plant instructor.
- Especially in the case of an outdoor use or in wet areas, you must consider the corresponding discharge currents when operated with a fault current circuit breaker.
- A table positioning or sequential positioning control is required for highly dynamic applications and jerk-free traversing profiles which is available with the "HighLine" device version.

Related topics:

▸ Commissioning of the "Switch-off positioning" technology application (📖 88)

8.5.1 Basic signal flow



[8-18] Signal flow of the switch-off positioning

- A Main speed setpoint offset and gain (L_OffsetGainP_1)
- B Additional speed setpoint offset and gain (L_OffsetGainP_2)
- C Setpoint generator (L_NSet_1)
- D Speed setpoint input limitation
- E Terminal assignment & display of digital control signals
- F Selection of edge/level for tripping the ramp down and stop functions (L_JogCtrlExtension_1)
- G Holding brake control
- H Integrated disposable "GeneralPurpose" functions: Analog switch, arithmetic, multiplication/division, binary delay element, binary logic, analog comparison, D-flipflop

8.5.2 Internal interfaces | application block "LA_SwitchPos"

Note!

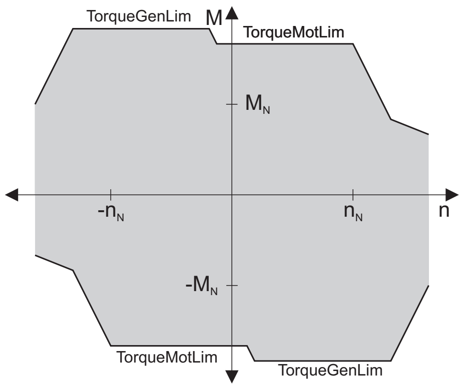
The connectors greyed out in the following table are hidden in the function block editor in the Lenze setting.

- These connections can be shown via the **Connector visibilities** command in the *Context menu* of the application block.

inputs

Designator Data type	Information/possible settings	
wCANDriveControl WORD	Control word via system bus (CAN) for device control • See the "wCANControl/wMCIControl control words" subchapter of the chapter on device control for a detailed description of the individual control bits.	
wMCIDriveControl WORD	Control word via communication module (e.g. PROFIBUS) for device control • See the "wCANControl/wMCIControl control words" subchapter of the chapter on device control for a detailed description of the individual control bits.	
wSMControl WORD	Interface to the optional safety system. • Setting control bit 0 ("SafeStop1") in this control word causes e.g. the automatic deceleration of the drive to standstill within this application (in the **Motion Control Kernel**). • See the subchapter "Interface to safety system" of the chapter on basic drive functions for a detailed description of the individual control bits.	
bCInh BOOL	Enable/inhibit inverter	
	FALSE	Enable inverter: The inverter switches to the "OperationEnabled" device status if no other source for controller inhibit is active. • C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.
	TRUE	Inhibit inverter (controller inhibit): The inverter switches to the "SwitchedOn" device status.
bFailReset BOOL	Reset error message In the Lenze setting this input is connected to the digital input controller enable so that a possibly existing error message is reset together with the controller enable (if the cause for the fault is eliminated).	
	TRUE	The current fault is reset, if the cause for the fault is eliminated. • If the fault still exists, the error status remains unchanged.
bSetQuickstop BOOL	Activate quick stop (QSP) • Also see device command "Activate/deactivate quick stop".	
	TRUE	Activate quick stop • Motor control is decoupled from the setpoint selection and, within the deceleration time parameterised in C00105, the motor is brought to a standstill (n_{act} = 0). • The motor is kept at a standstill during closed-loop operation. • A pulse inhibit is set if the auto-DCB function has been activated via C00019.
	FALSE	Deactivate quick stop • The quick stop is deactivated if no other source for the quick stop is active. • C00159 displays a bit code of active sources/causes for the quick stop.

<table>
<tr><th>Designator
Data type</th><th colspan="2">Information/possible settings</th></tr>
<tr><td rowspan="4">bSetDCBrake
BOOL</td><td colspan="2">Manual DC-injection braking (DCB)
• Detailed information on DC-injection braking is provided in the motor control chapter, subchapter "DC-injection braking".</td></tr>
<tr><td colspan="2">Note!
Holding braking is not possible when this braking mode is used!
Use the basic "Holding brake control" function for controlling the holding brake with a low rate of wear.</td></tr>
<tr><td>FALSE</td><td>Deactivate DC-injection braking.</td></tr>
<tr><td>TRUE</td><td>Activate DC-injection braking, i.e. the drive is brought to a standstill by means of DC-injection braking.
• The braking effect stops when the rotor is at standstill.
• After the hold time (C00107) has expired, the controller sets the pulse inhibit.</td></tr>
<tr><td rowspan="2">bRFG_Stop
BOOL</td><td colspan="2">Ramp function generator: Maintain the current value of the main setpoint integrator
• The speed, for instance, of a running ramp process is immediately kept constant when bRFG_Stop is activated. At the same time, the acceleration/deceleration jumps to the value "0".
• For a detailed functional description see the L_NSet FB.</td></tr>
<tr><td>TRUE</td><td>The current value of the main setpoint integrator is held.</td></tr>
<tr><td rowspan="2">nVoltageAdd_a
INT</td><td colspan="2">Additive voltage impression
• An additional setpoint for the motor voltage can be specified via this process input.
• If there are, for instance, different loads at the motor output end, it is possible to apply a voltage boost at the starting time.
• If the value is negative, the voltage is reduced.
• Scaling: 16384 ≡ 1000 V</td></tr>
<tr><td colspan="2">Stop!
Values selected too high may cause the motor to heat up due to the resulting current!</td></tr>
<tr><td rowspan="2">nBoost_a
INT</td><td colspan="2">Additional setpoint for the motor voltage at speed = 0
• The entire voltage-frequency characteristic is provided with an offset.
• Scaling: 16384 ≡ 1000 V</td></tr>
<tr><td colspan="2">Stop!
Values selected too high may cause the motor to heat up due to the resulting current!</td></tr>
<tr><td>nPWMAngleOffset
INT</td><td colspan="2">Additional offset for the electrical angle of rotation
• If a torque is connected, e.g. dynamic acceleration processes can be generated.
• Scaling: ±32767 ≡ ±180 ° angle of rotation</td></tr>
</table>

Designator / Data type	Information/possible settings
nTorqueMotLim_a nTorqueGenLim_a INT	Torque limitation in motor mode and in generator mode • These input signals are directly transferred to the motor control to limit the inverter's maximum torque in motor and generator mode. • The drive cannot output a higher torque in motor/generator mode than set here. • The applied values (any polarity) are internally interpreted as absolute values. • If V/f characteristic control (VFCplus) is selected, limitation is indirectly performed via a so-called I_{max} controller. • If sensorless vector control (SLVC) or servo control (SC) is selected, limitation has a direct effect on the torque-producing current component. • Scaling: 16384 ≡ 100 % M_{max} (C00057)
	Torque limits in motor and generator mode: 
bSetSpeedCcw BOOL	Change of direction of rotation • For instance if a motor or gearbox is fixed laterally reversed to a machine part, but the setpoint selection should still be executed for the positive direction of rotation. FALSE: Clockwise rotation (Cw) TRUE: Direction of rotation to the left (Ccw)
bRLQCw BOOL	Activate clockwise rotation (fail-safe) • For a detailed functional description see the L_RLQ FB. FALSE: Quick stop TRUE: CW rotation
bRLQCcw BOOL	Activate counter-clockwise rotation (fail-safe) • For a detailed functional description see the L_RLQ FB. FALSE: Quick stop TRUE: CCW rotation
nMainSetValue_a INT	Main speed setpoint • Offset and gain of this input signal can be set in C00696 and C00670 for a simple signal adjustment of a setpoint encoder. • Scaling: 16384 ≡ 100 % reference speed (C00011) • The main setpoint is transformed to a speed setpoint in the setpoint encoder via a ramp function generator with linear or S-shaped ramps. • Upstream to the ramp function generator, a blocking speed masking function and a setpoint MinMax limitation are effective. • For a detailed functional description see the L_NSet FB.
nAuxSetValue_a INT	Additional speed setpoint • Offset and gain of this input signal can be set in C00697 and C00671 for a simple signal adjustment of a setpoint encoder. • Scaling: 16384 ≡ 100 % reference speed (C00011) • The additional speed setpoint can be linked arithmetically with the main speed setpoint behind the ramp function generator. • The additional speed setpoint can be shown via ramp times of a second ramp function generator. • For a detailed functional description see the L_NSet FB.

Designator Data type	Information/possible settings
Switch-off positioning	
bJogCtrlInputSel1 bJogCtrlInputSel2 BOOL	Selection inputs for a binary coded selection of the switch-off position 1 ... 3 • Activation of the signal pairs *bJogCtrlSlowDown1/bJogCtrlStop1*, *bJogCtrlSlowDown2/bJogCtrlStop2* or *bJogCtrlSlowDown3/bJogCtrlStop3* according to the Truth table for activating the pre-switch off.
bJogCtrlRfgIn BOOL	Ramping down of the setpoint generator in the downstream L_NSet FB according to the Truth table for activating the pre-switch off
bJogCtrlJog1 bJogCtrlJog2 BOOL	Inputs for overriding fixed setpoints (JOG setpoints) for the main setpoint • If the pre-switch off is inactive (*bJogCtrlInputSel1* and *bJogCtrlInputSel2* are both set to FALSE), the two control signals are passed through 1:1 to the downstream FB L_NSet. • To achieve the desired behaviour (starting at high speed, pre-switch off at low speed), both inputs must be set to TRUE. • Fixed setpoint 2 must be less than fixed setpoint 3! Otherwise, the drive will start at a low speed and accelerate after the pre-switch off. • If in additional to the inputs *bJogCtrlJog1* and *bJogCtrlJog2* the selection inputs *bJogSpeed4* and *bJogSpeed8* are assigned, different fixed setpoints can result from this and the drive may travel with different speeds than selected via *bJogCtrlJog1* and *bJogCtrlJog2*.
bJogCtrlSlowDown1 bJogCtrlSlowDown2 bJogCtrlSlowDown3 BOOL	Activation of fixed setpoint 2 in the downstream L_NSet FB • These inputs only have a function if they have been previously activated via *bJogCtrlInputSel1* and *bJogCtrlInputSel2* (see Truth table for activating the pre-switch off).
bJogCtrlStop1 bJogCtrlStop2 bJogCtrlStop3 BOOL	Ramping down of the ramp function generator in the downstream L_NSet FB • These inputs only have a function if they have been previously activated via *bJogCtrlInputSel1* and *bJogCtrlInputSel2* (see Truth table for activating the pre-switch off).
bJogSpeed4 bJogSpeed8 BOOL	Inputs for overriding fixed setpoints (JOG setpoints) for the main setpoint • A fixed setpoint for the setpoint generator can be activated instead of the main setpoint via these selection inputs. • The selection inputs are binary coded. • For a detailed functional description see the L_NSet FB.
bJogRamp1 ... bJogRamp8 BOOL	Selection inputs for alternative acceleration/deceleration times for the main setpoint • The four selection inputs are binary coded, therefore 15 alternative acceleration/deceleration times can be selected. • For main setpoint *nMainSetValue_a*, the set acceleration time (C00012) and deceleration time (C00013) are active in the case of the binary coded selection "0" (all inputs = FALSE or not assigned). • Alternative acceleration times are selected in C00101/1...15. • The selection of the alternative deceleration times is carried out in C00103/1...15. • For a detailed functional description see the L_NSet FB.
MCK basic functions	
bMBrakeRelease BOOL	Holding brake control: Release/apply brake • In conjunction with the operating mode selected in C02580 (Lenze setting: "Brake control off"). FALSE: Apply brake. • During automatic operation, the internal brake logic controls the brake. TRUE: Release brake manually (forced release). • **Note!** The brake can also be released when the controller is inhibited! • During automatic operation, the internal brake logic is deactivated and the brake is released (supervisor operation). If a controller inhibit has been set by the brake control, it will be deactivated. • In semi-automatic operation, the brake is released including feedforward control.

Designator Data type	Information/possible settings
GP: GeneralPurpose The following inputs are interconnected with logic/arithmetic functions on application level for free usage. ▸ "GeneralPurpose" functions	
nGPAnalogSwitchIn1_a nGPAnalogSwitchIn2_a INT	Analog switch: Input signals • The input signal selected via the selection input *bGPAnalogSwitchSet* is output at output *nGPAnalogSwitchOut_a*.
bGPAnalogSwitchSet BOOL	Analog switch: Selection input FALSE: *nGPAnalogSwitchOut_a = nGPAnalogSwitchIn1_a* TRUE: *nGPAnalogSwitchOut_a = nGPAnalogSwitchIn2_a*
nGPArithmetikIn1_a nGPArithmetikIn2_a INT	Arithmetic: Input signals • The arithmetic function is selected in C00338. • The result is output at output *nGPArithmetikOut_a*.
nGPMulDivIn_a INT	Multiplication/Division: Input signal • The factor for the multiplication can be set in C00699/1 (numerator) and C00699/2 (denominator). • The result is output at output *nGPMulDivOut_a*.
bGPDigitalDelayIn BOOL	Binary delay element: Input signal • The on-delay can be set in C00720/1. • The off-delay can be set in C00720/2. • The time-delayed input signal is output at output *bGPDigitalDelayOut*.
bGPLogicIn1 bGPLogicIn2 bGPLogicIn3 BOOL	Binary logic: Input signals • The logic operation is selected in C00820. • The result is output at output *bGPLogicOut*.
nGPCompareIn1_a nGPCompareIn2_a INT	Analog comparison: Input signals • The comparison operation is selected in C00680. • Hysteresis and window size can be set in C00680 and C00682. • If the comparison statement is true, the output *bGPCompareOut* will be set to TRUE.
bGPDFlipFlop_InD bGPDFlipFlop_InClk bGPDFlipFlop_InClr BOOL	D-FlipFlop: Input signals • Data, clock and reset input
Free inputs The following inputs can freely be interconnected on the application level. The signals can be transferred from the I/O level to the application level via these inputs.	
bFreeIn1 ... bFreeIn8 BOOL	Free inputs for digital signals
wFreeIn1 ... wFreeIn4 WORD	Free inputs for 16-bit signals
dnFreeIn1_p ... dnFreeIn2_p DINT	Free inputs for 32-bit signals

outputs

Designator Data type	Value/meaning
wDriveControlStatus WORD	Status word of the inverter • The status word contains information on the currents status of the inverter. • See the "wDeviceStatusWord status word" subchapter of the chapter on device control for a detailed description of the bit assignment.
wStateDetermFailNoLow WORD	Display of the status determining error (LOW word)
wStateDetermFailNoHigh WORD	Display of the status determining error (HIGH word)

Designator / Data type	Value/meaning	
bDriveFail BOOL	TRUE	Inverter in error status. • "Fault" device status is active.
bWarningActive BOOL	TRUE	A monitoring in the inverter, for which the error response "Warning" or "Warning locked" has been parameterised, responded.
bSafeTorqueOff BOOL	TRUE	Safe torque off. • "SafeTorqueOff" device status is active.
bDriveReady BOOL	TRUE	Inverter is ready for operation. • "SwitchedOn" device status is active. • The drive is in this device status if the DC bus voltage is applied and the inverter is still inhibited by the user (controller inhibit).
bCInhActive BOOL	TRUE	Controller inhibit is active.
bImplsActive BOOL	TRUE	Pulse inhibit is active.
bQSPIsActive BOOL	TRUE	Quick stop is active.
bSpeedCcw BOOL	Current direction of rotation	
	FALSE	Clockwise rotation (Cw)
	TRUE	Direction of rotation to the left (Ccw)
bSpeedActCompare BOOL	Result of the speed comparison (detection of speed=0)	
	TRUE	During open-loop operation: Speed setpoint < Comparison value (C00024)
		During closed-loop operation: Actual speed value < Comparison value (C00024)
bImaxActive BOOL	"Current setpoint inside the limitation" status signal	
	TRUE	The current setpoint is internally limited (the inverter operates at the maximum current limit).
bSpeedSetReached BOOL	Status signal "setpoint = 0"	
	TRUE	Speed setpoint from the ramp function generator = 0
bSpeedActEqSet BOOL	TRUE	Actual speed value = speed setpoint
nMotorCurrent_a INT	Current stator current/effective motor current • Scaling: 16384 ≡ 100 % I_{max_mot} (C00022)	
nMotorSpeedSet_a INT	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)	
nMotorSpeedAct_a INT	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)	
nMotorTorqueAct_a INT	Actual torque • In the "VFC (+encoder)" operating mode of the motor control, this value is determined from the current motor current and corresponds to the actual torque only by approximation. • Scaling: 16384 ≡ 100 % M_{max} (C00057)	
nDCVoltage_a INT	Current DC-bus voltage • Scaling: 16384 ≡ 1000 V	
nMotorVoltage_a INT	Current motor voltage/inverter output voltage • Scaling: 16384 ≡ 1000 V	

Designator Data type	Value/meaning	
MCK basic functions		
bMBrakeReleaseOut BOOL	Holding brake control: Trigger signal for the holding brake control switching element via a digital output • Use bit 0 in C02582 to activate inverted switching element triggering.	
	FALSE	Apply brake.
	TRUE	Release brake.
bMBrakeReleased BOOL	Holding brake control: "Brake released" considering the brake release time • When the holding brake is triggered to close, *bMBrakeReleased* is immediately set to FALSE even if the brake closing time has not yet elapsed!	
	TRUE	Brake released (after the brake release time has expired).
GP: GeneralPurpose The following outputs are interconnected with logic/arithmetic functions on application level for free usage. ▸ "GeneralPurpose" functions		
nGPAnalogSwitchInOut_a INT	Analog switch: Output signal	
nGPArithmetikOut_a INT	Arithmetic: Output signal	
nGPMulDivOut_a INT	Multiplication/Division: Output signal	
bGPDigitalDelayOut BOOL	Binary delay element: Output signal	
bGPLogicOut BOOL	Binary logic: Output signal	
bGPCompareOut BOOL	Analog comparison: Output signal	
bGPSignalOut1 ... bGPSignalOut4 BOOL	Binary signal monitor: Output signals • The signal sources to be output are selected in C00411/1...4. • A bit coded inversion of the output signals can be parameterised in C00412.	
nGPSignalOut1_a ... nGPSignalOut4_a BOOL	Analog signal monitor: Output signals • The signal sources to be output are selected in C00410/1...4. • Gain and offset for each output signal can be parameterised in C00413/1...8.	
bGPDFlipFlop_Out BOOL	D-FlipFlop: Output signal	
bGPDFlipFlop_NegOut BOOL	D-FlipFlop: Negated output signal	
Free outputs The following outputs can freely be interconnected on the application level. The signals from the application level can be transferred to the I/O level via these outputs.		
bFreeOut1 ... bFreeOut8 BOOL	Free outputs for digital signals	
wFreeOut1 ... wFreeOut4 WORD	Free outputs for 16-bit signals	
dnFreeOut1_p dnFreeOut2_p WORD	Free outputs for 32-bit signals	

8.5.2.1 Truth table for activating the pre-switch off

Input		**Function**	**Response in the setpoint generator (FB L_NSet)**
bJogCtrl InputSel1	**bJogCtrl InputSel2**		
FALSE	FALSE	Pre-switch off inactive	No response • The input signal *bJogCtrlRfgIn* is output directly at output *bRfgOut*. • The input signals *bJogCtrlJog1* and *bJogCtrlJog2* are passed through 1:1 to the downstream FB L_NSet for the selection of fixed setpoints.
TRUE	FALSE	The *bJogCtrlSlowDown1* and *bJogCtrlStop1* inputs are evaluated.	**Pre-switch off can be activated** • If the slowdown function is activated via the selected *bJogCtrlSlowDown* input, fixed setpoint 2 is activated in the setpoint generator. • If the stop function is activated via the selected *bJogCtrlStop* input, the setpoint generator is deactivated.
FALSE	TRUE	The *bJogCtrlSlowDown2* and *bJogCtrlStop2* inputs are evaluated.	
TRUE	TRUE	The inputs *bSJogCtrllowDown3* and *bJogCtrlStop3* are evaluated.	

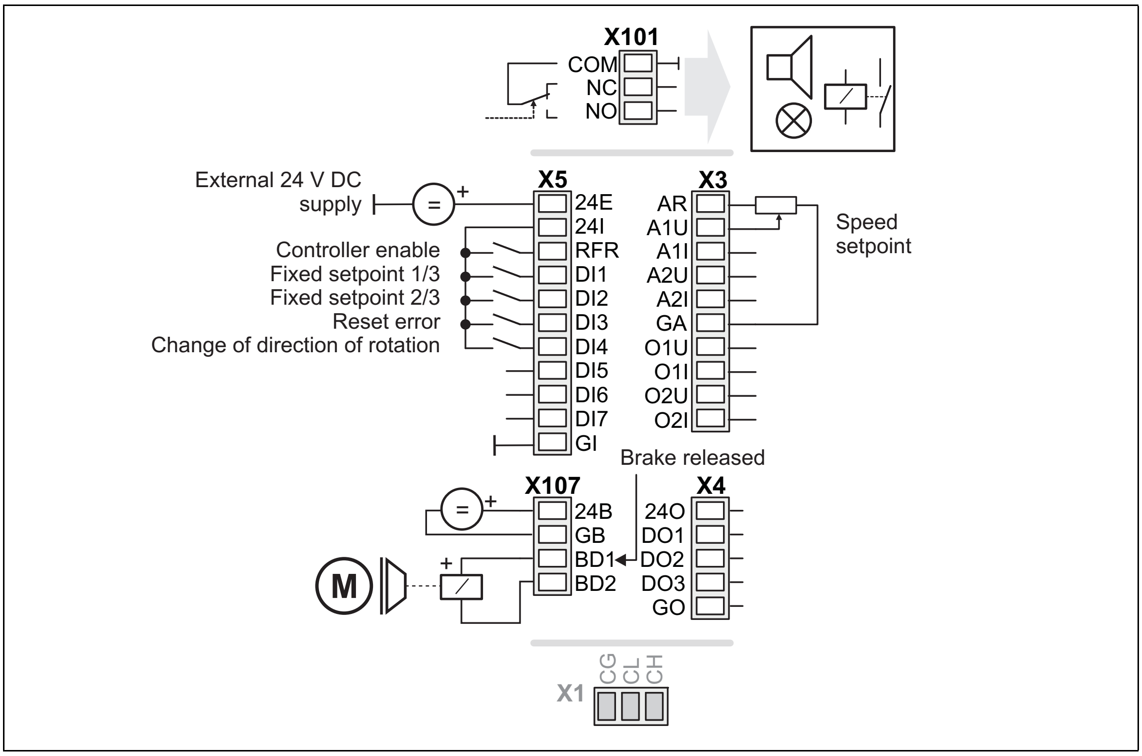
[8-1] Truth table for activating the pre-switch off

8.5.3 Terminal assignment of the control modes

The following comparison provides information about which inputs/outputs of the application block **LA_SwitchPos** are interconnected to the digital and analog input/output terminals of the inverter in the different control modes.

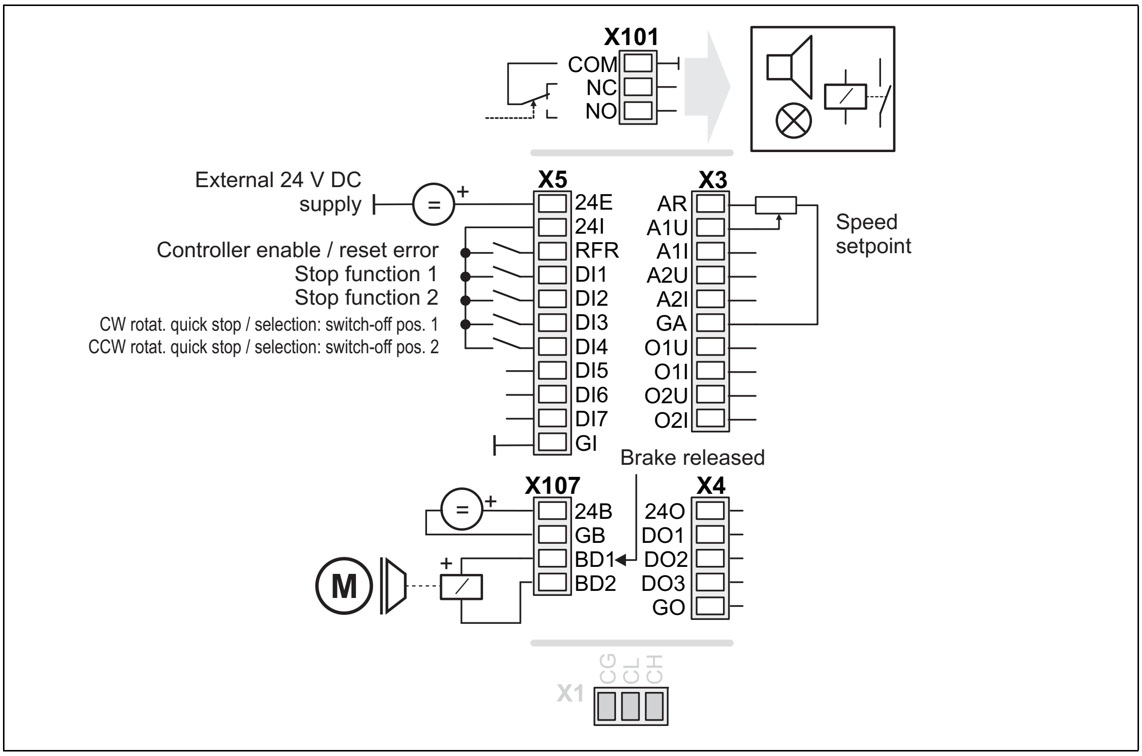
	Control mode (C00007)							
	10: Terminals 0	12: Terminals 2	14: Terminals 11	16: Terminal 16	20: Keypad	21: PC	30: CAN	40: MCI
Digital input terminals								
X5/RFR	**Controller enable**	**Controller enable / Reset of error message** bFailReset						
X5/DI1	**Fixed setpoint 1/3** bJogCtrlJog1	**Stop function 1** bJogCtrlStop1		**Fixed setpoint 1/3** bJogCtrlJog1	-	-	**Stop function 1** bJogCtrlStop1	
X5/DI2	**Fixed setpoint 2/3** bJogCtrlJog2	**Stop function 2** bJogCtrlStop2	**Selection: Pre-switch off 1** bJogCtrlSlowDown 1	**Fixed setpoint 2/3** bJogCtrlJog2	-	-	**Selection: Pre-switch off 1** bJogCtrlSlowDown 1	
X5/DI3	**Reset error message** bFailReset	**CW rotation quick stop** bRLQCw Selection: Switch-off position 1 bJogCtrlInputSel1		**CW rotation quick stop** bRLQCw	-	-	**Stop function 2** bJogCtrlStop2	
X5/DI4	**Change of direction of rotation** bSetSpeedCcw	**CCW rotation quick stop** bRLQCcw Selection: Switch-off position 2 bJogCtrlInputSel2		**CCW rotation quick stop** bRLQCcw	-	-	**Selection: Pre-switch off 2** bJogCtrlSlowDown 2	
X5/DI5	-	-	**Stop function 2** bJogCtrlStop2	-	-	-	**Stop function 3** bJogCtrlStop3	
X5/DI6	-	-	**Selection: Pre-switch off 2** bJogCtrlSlowDown 2	-	-	-	**Selection: Pre-switch off 3** bJogCtrlSlowDown 3	
X5/DI7	-	-	-	-	-	-	-	-
Analog input terminals								
X3/A1U, A1I	**Main speed setpoint** nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)				-	-	**Additional speed setpoint** nAuxSetValue_a 10 V ≡ 100 % reference speed (C00011)	
X3/A2U, A2I	-	-	-	-	-	-	-	-
Digital output terminals								
X4/DO1 ... DO3	-	-	-	-	-	-	-	-
X107/BD1, BD2	**Control of the holding brake** bMBrakeReleaseOut							
X101/COM, NO	-	-	-	-	-	-	-	-
Analog output terminals								
X3/O1U, O1I	-	-	-	-	-	-	-	-
X3/O2U, O2I								

8.5.3.1 Terminals 0



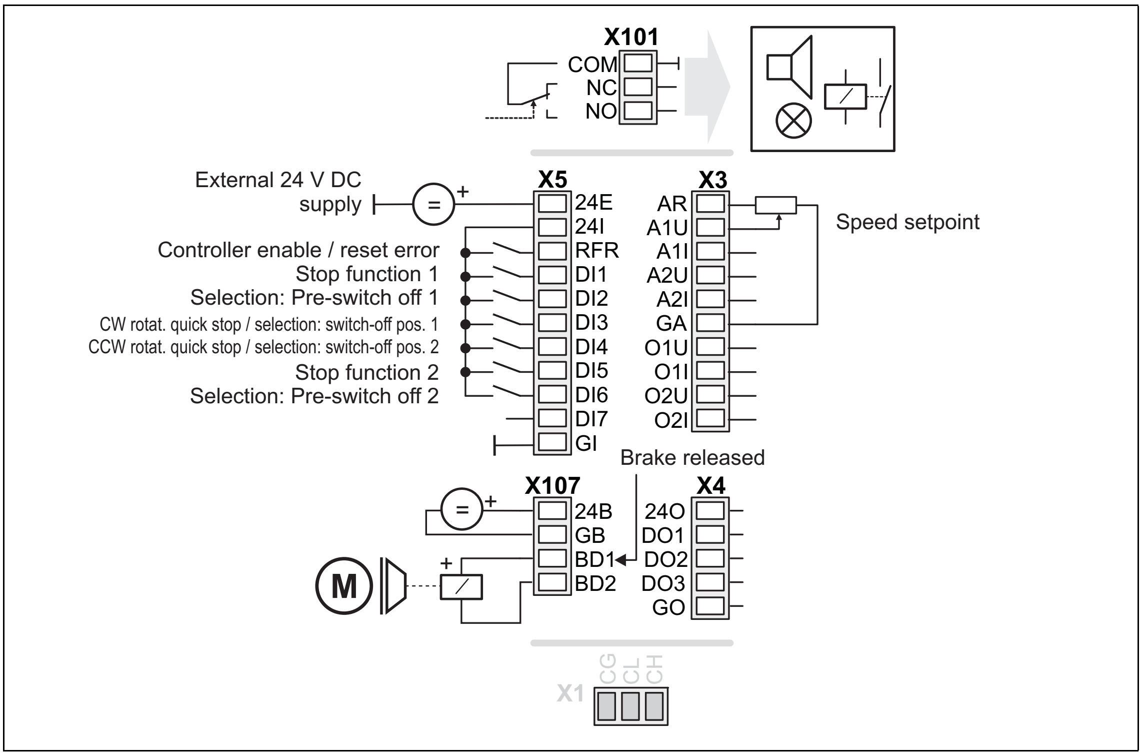
Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	-		X3/A1U	LA_SwitchPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	LA_SwitchPos.bJogCtrlJog1		X3/A1I	
X5/DI2	LA_SwitchPos.bJogCtrlJog2		X3/A2U	-
X5/DI3	LA_SwitchPos.bFailReset		X3/A2I	-
X5/DI4	LA_SwitchPos.bSetSpeedCcw		X3/O1U	-
X5/DI5	-		X3/O1I	-
X5/DI6	-		X3/O2U	-
X5/DI7	-		X3/O2I	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut		X4/DO1	-
X107/BD2	-		X4/DO2	-
			X4/DO3	-

8.5.3.2 Terminals 2



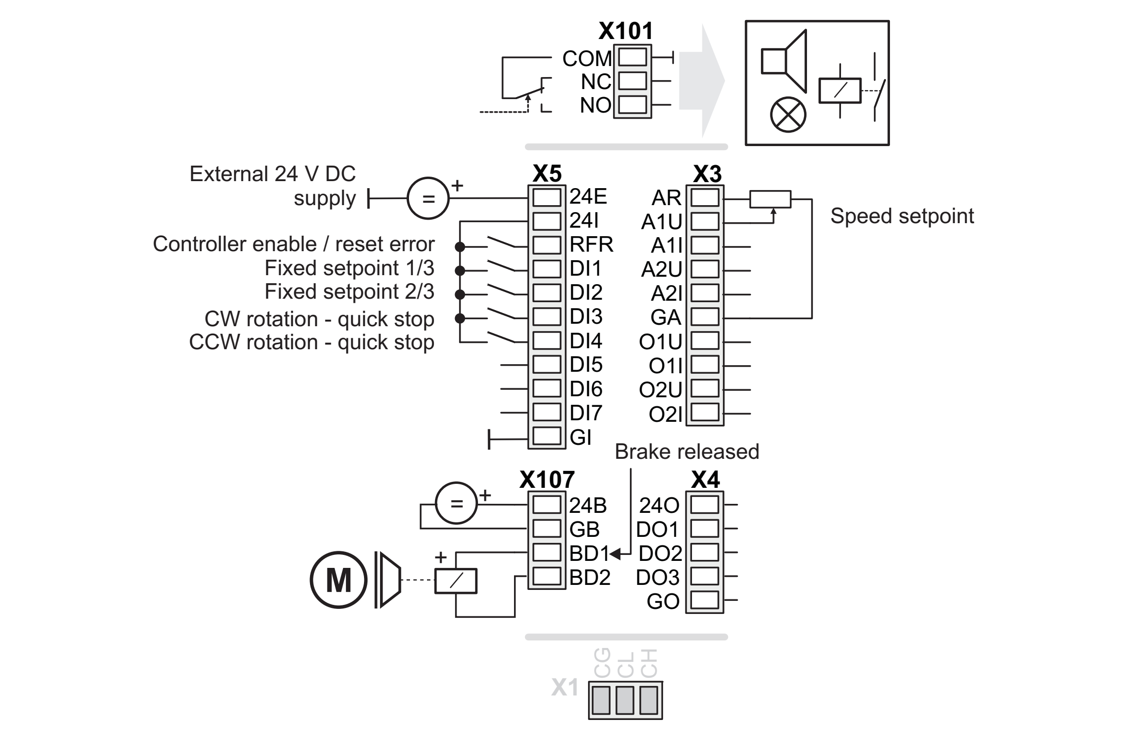
Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	LA_SwitchPos: bFailReset		**X3/A1U**	LA_SwitchPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	LA_SwitchPos: bJogCtrlStop1		**X3/A1I**	
X5/DI2	LA_SwitchPos: bJogCtrlStop2		**X3/A2U**	-
X5/DI3	LA_SwitchPos: bRLQCw		**X3/A2I**	-
	LA_SwitchPos: bJogCtrlInputSel1		**X3/O1U**	-
X5/DI4	LA_SwitchPos: bRLQCcw		**X3/O1I**	-
	LA_SwitchPos: bJogCtrlInputSel2		**X3/O2U**	-
X5/DI5	-			
X5/DI6	-		**X4/DO1**	-
X5/DI7	-		**X4/DO2**	-
			X4/DO3	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut			
X107/BD2	-			

8.5.3.3 Terminals 11



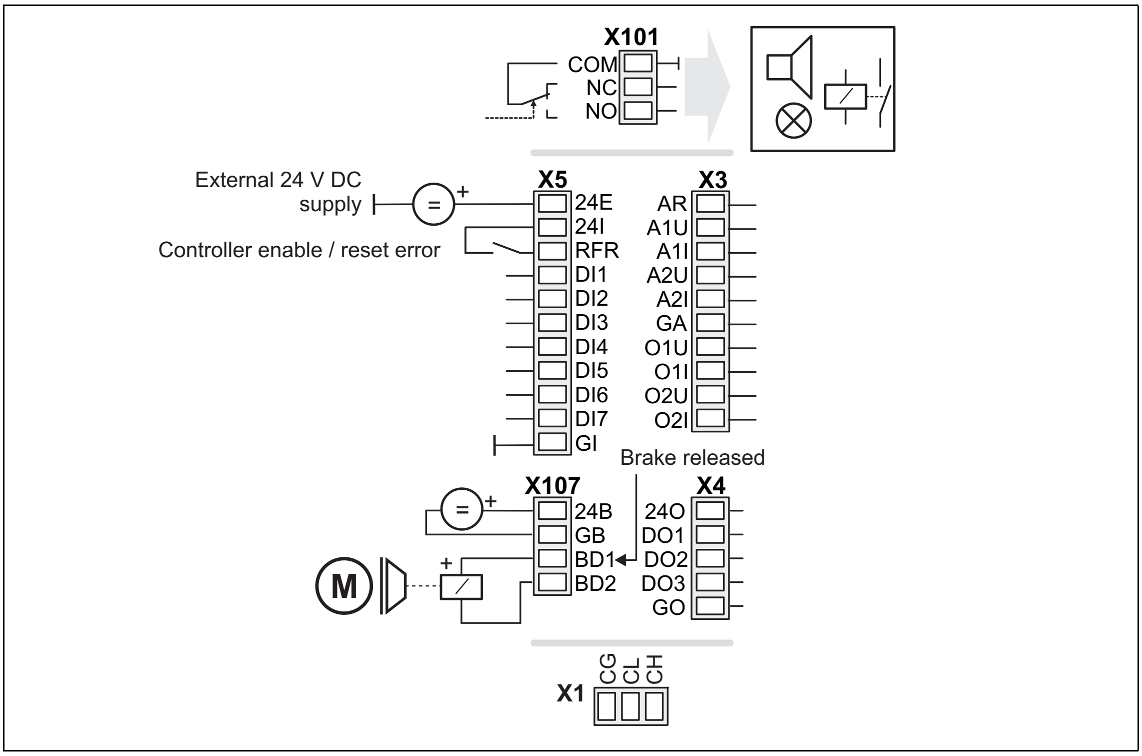
Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	LA_SwitchPos: bFailReset		X3/A1U	LA_SwitchPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	LA_SwitchPos: bJogCtrlStop1		X3/A1I	
X5/DI2	LA_SwitchPos: bJogCtrlSlowDown1		X3/A2U	-
X5/DI3	LA_SwitchPos: bRLQCw		X3/A2I	-
	LA_SwitchPos: bJogCtrlInputSel1		X3/O1U	-
X5/DI4	LA_SwitchPos: bRLQCcw		X3/O1I	-
	LA_SwitchPos: bJogCtrlInputSel2		X3/O2U	-
X5/DI5	LA_SwitchPos: bJogCtrlStop2			
X5/DI6	LA_SwitchPos: bJogCtrlSlowDown2		X4/DO1	-
X5/DI7	-		X4/DO2	-
			X4/DO3	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut			
X107/BD2	-			

8.5.3.4 Terminal 16



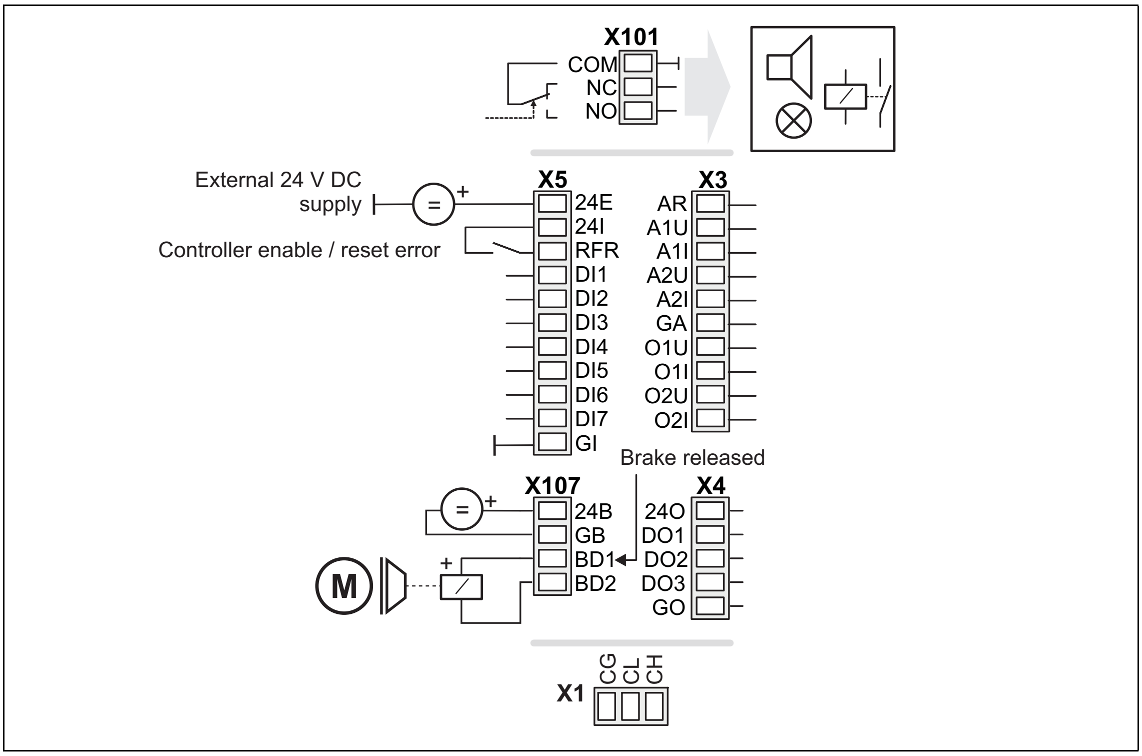
Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	LA_SwitchPos: bFailReset		**X3/A1U**	LA_SwitchPos.nMainSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	LA_SwitchPos: bJogCtrlJog1		**X3/A1I**	
X5/DI2	LA_SwitchPos: bJogCtrlJog2		**X3/A2U**	-
X5/DI3	LA_SwitchPos: bRLQCw		**X3/A2I**	-
X5/DI4	LA_SwitchPos: bRLQCcw		**X3/O1U**	-
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut		**X4/DO1**	-
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.5.3.5 Keypad



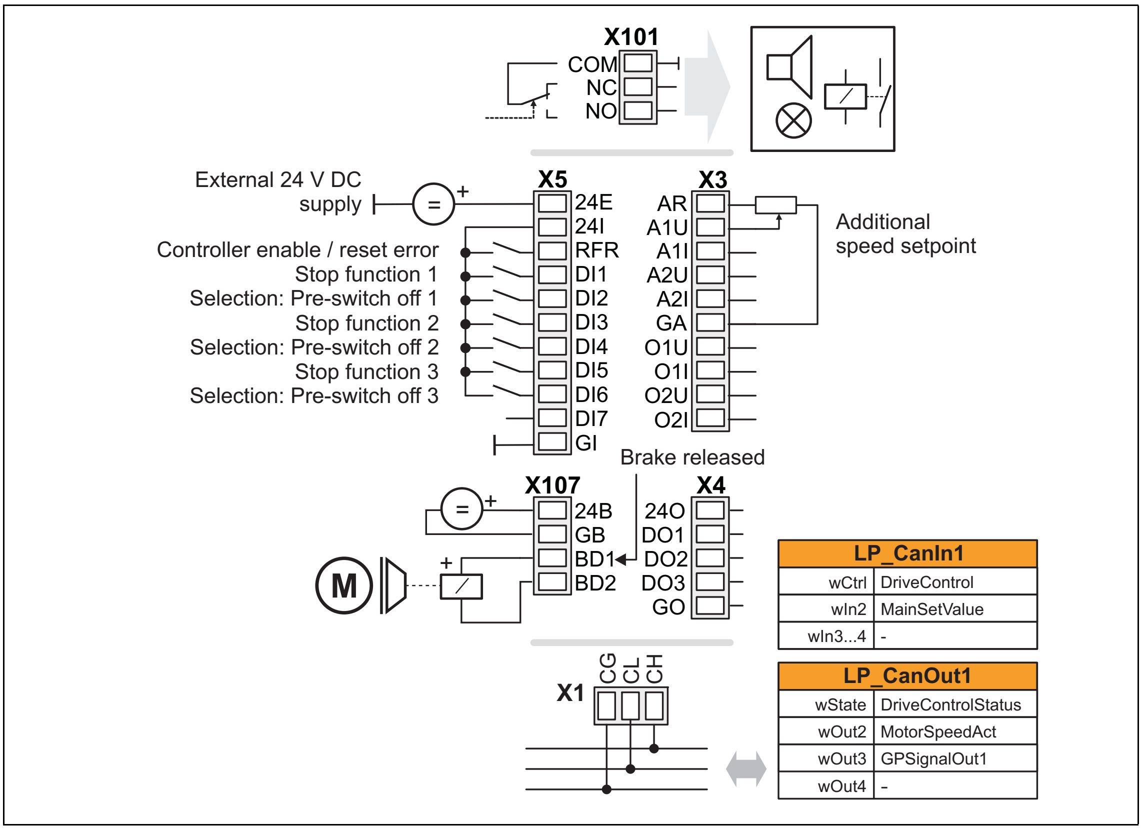
Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	-		**X3/A1U**	-
X5/DI1	-		**X3/A1I**	-
X5/DI2	-		**X3/A2U**	-
X5/DI3	-		**X3/A2I**	-
X5/DI4	-		**X3/O1U**	-
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut		**X4/DO1**	-
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.5.3.6 PC



Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	-		**X3/A1U**	-
X5/DI1	-		**X3/A1I**	-
X5/DI2	-		**X3/A2U**	-
X5/DI3	-		**X3/A2I**	-
X5/DI4	-		**X3/O1U**	-
X5/DI5	-		**X3/O1I**	-
X5/DI6	-		**X3/O2U**	-
X5/DI7	-		**X3/O2I**	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut		**X4/DO1**	-
X107/BD2	-		**X4/DO2**	-
			X4/DO3	-

8.5.3.7 CAN



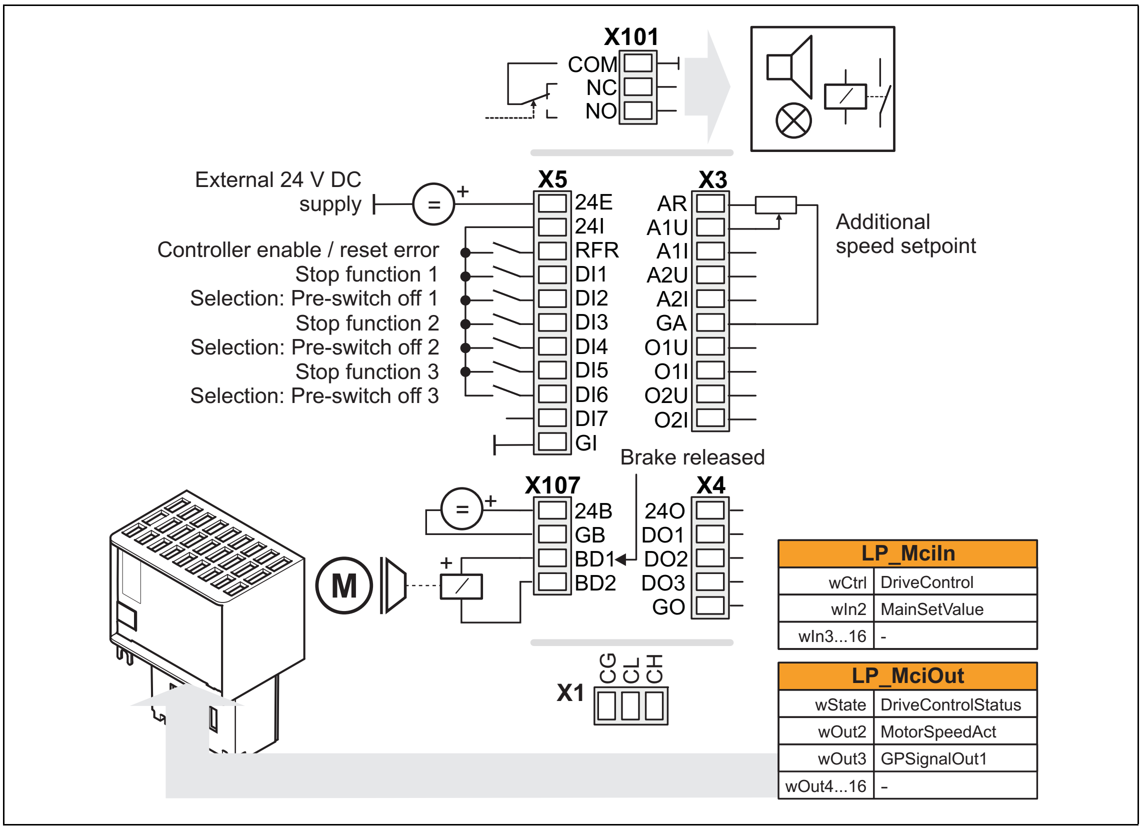
Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	LA_SwitchPos: bFailReset		X3/A1U	LA_SwitchPos.nAuxSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	LA_SwitchPos: bJogCtrlStop1		X3/A1I	
X5/DI2	LA_SwitchPos: bJogCtrlSlowDown1		X3/A2U	-
X5/DI3	LA_SwitchPos: bJogCtrlStop2		X3/A2I	-
X5/DI4	LA_SwitchPos: bJogCtrlSlowDown2		X3/O1U	-
X5/DI5	LA_SwitchPos: bJogCtrlStop3		X3/O1I	-
X5/DI6	LA_SwitchPos: bJogCtrlSlowDown3		X3/O2U	-
X5/DI7	-		X3/O2I	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut		X4/DO1	-
X107/BD2	-		X4/DO2	-
			X4/DO3	-

▸ Process data assignment for fieldbus communication (📖 564)

Note!

You must set the setpoint arithmetic in C00190 to "1: NOut = NSet + NAdd" so that the additional speed setpoint selected via the analog input A1U has an additive effect.

8.5.3.8 MCI



Connection	Assignment		Connection	Assignment
X101/NC-NO	-			
X5/RFR	LA_SwitchPos: bFailReset		X3/A1U	LA_SwitchPos.nAuxSetValue_a 10 V ≡ 100 % reference speed (C00011)
X5/DI1	LA_SwitchPos: bJogCtrlStop1		X3/A1I	
X5/DI2	LA_SwitchPos: bJogCtrlSlowDown1		X3/A2U	-
X5/DI3	LA_SwitchPos: bJogCtrlStop2		X3/A2I	-
X5/DI4	LA_SwitchPos: bJogCtrlSlowDown2		X3/O1U	-
X5/DI5	LA_SwitchPos: bJogCtrlStop3		X3/O1I	-
X5/DI6	LA_SwitchPos: bJogCtrlSlowDown3		X3/O2U	-
X5/DI7	-		X3/O2I	-
X107/BD1	LA_SwitchPos.bMBrakeReleaseOut		X4/DO1	-
X107/BD2	-		X4/DO2	-
			X4/DO3	-

▸ Process data assignment for fieldbus communication (📖 564)

Note!

You must set the setpoint arithmetic in C00190 to "1: NOut = NSet + NAdd" so that the additional speed setpoint selected via the analog input A1U has an additive effect.

8.5.4 Process data assignment for fieldbus communication

The fieldbus communication is connected (preconfigured) to the previously selected technology application by selecting the corresponding control mode in C00007:

- "30: CAN" for the connection to the system bus (CAN)
- "40: MCI" for the connection to a plugged-on communication module (e.g. PROFIBUS)

The assignment of the process data words depends only on the application, not on the bus system used:

Input words	Name	Assignment
Word 1	DriveControl	Control word • For bit assignment see the table below.
Word 2	MainSetValue	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	-	Not preconfigured
Word 4	-	Not preconfigured
Words 5 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Control word	Name	Function
Bit 0	SwitchOn	1 ≡ Change to the "SwitchedOn" device status • This bit must be set in the CAN/MCI control word to ensure that the device changes to the "SwitchedOn" device status after mains connection without the need for a master control specifying this bit via fieldbus. • If control via a bus system is not wanted (e.g. in the case of control via terminals), the *wDriveCtrl* output signal of the LS_ParFix system block can be connected to the control word inputs.
Bit 1	DisableVoltage	1 ≡ Inhibit inverter control (pulse inhibit)
Bit 2	SetQuickStop	1 ≡ Activate quick stop (QSP). ▶ Activate/deactivate quick stop (📖 114)
Bit 3	EnableOperation	1 ≡ Enable inverter (RFR) • If control via terminals is performed, this bit must be set both in the CAN control word and in the MCI control word. Otherwise, the controller is inhibited. ▶ Enable/inhibit inverter (📖 113)
Bit 4	ModeSpecific_1	Reserved (currently not assigned)
Bit 5	JogCtrlInputSel1	Binary coded selection of the switch-off position 1 ... 3 • Activation of the signal pairs *bJogCtrlSlowDown1/bJogCtrlStop1*, *bJogCtrlSlowDown2/bJogCtrlStop2* or *bJogCtrlSlowDown3/bJogCtrlStop3* according to the Truth table for activating the pre-switch off.
Bit 6	JogCtrlInputSel2	
Bit 7	ResetFault	1 ≡ Reset fault (trip reset) • Acknowledge error message (if the error cause has been eliminated). ▶ Reset error (📖 117)
Bit 8	bJogCtrlRfgIn	Ramping down of the setpoint generator in the downstream L_NSet FB according to the Truth table for activating the pre-switch off
Bit 9	reserved_1	Reserved (currently not assigned)
Bit 10	reserved_2	

Control word	Name	Function
Bit 11	MBrkRelease	Holding brake control: 0 ≡ Apply brake 1 ≡ Release brake • In conjunction with the operating mode selected in C02580 (Lenze setting: "Brake control off").
Bit 12	JogCtrlJog1	Binary coded selection of the fixed setpoints (JOG setpoints)
Bit 13	JogCtrlJog2	
Bit 14	SetFail	1 ≡ Set error (trip set)
Bit 15	SetSpeedCcw	0 ≡ Direction of rotation to the right (Cw) 1 ≡ Direction of rotation to the left (Ccw)

Output words	Name	Assignment
Word 1	DriveControlStatus	Status word • For bit assignment see the table below.
Word 2	MotorSpeedAct	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	GPSignalOut1	Analog signal monitor: Output signal 1 • The selection of the signal source to output is executed in C00410/1. • Gain and offset for the output signal can be parameterised in C00413/1 and C00413/2. • For a detailed functional description see the L_SignalMonitor_a FB.
Word 4	-	Not preconfigured
Words 5 ... 16	-	Not preconfigured • Only available in control mode "40: MCI".

Status word	Name	Status
Bit 0	DriveFail	1 ≡ Inverter in the error status • "Fault" device status is active.
Bit 1	PowerDisabled	1 ≡ Inverter control inhibited (pulse inhibit is active)
Bit 2	DriveReady	1 ≡ Inverter is ready for operation • "SwitchedOn" device status is active. • The drive is in this device status if the DC bus voltage is applied and the inverter is still inhibited by the user (controller inhibit).
Bit 3	SpeedCcw	0 ≡ Direction of rotation to the right (Cw) 1 ≡ Direction of rotation to the left (Ccw)
Bit 4	QSPIsActive	1 ≡ Quick stop is active
Bit 5	BrakeReleased	1 ≡ Brake released (after the brake opening time has elapsed)
Bit 6	ActSpeedIsZero	During open-loop operation: 1 ≡ Speed setpoint < Comparison value (C00024)
		During closed-loop operation: 1 ≡ Actual speed value < Comparison value (C00024)
Bit 7	ControllerInhibit	1 ≡ Inverter is inhibited (controller inhibit is active)
Bit 8	StatusCodeBit0	Bit coded display of the active device status ▸ Device state machine and device states (see table [4-1])
Bit 9	StatusCodeBit1	
Bit 10	StatusCodeBit2	
Bit 11	StatusCodeBit3	
Bit 12	Warning	1 ≡ a warning is indicated

Status word	Name	Status
Bit 13	Trouble	1 ≡ Inverter is in the "Trouble" device status • E.g. if an overvoltage has occurred.
Bit 14	JogCtrlInputSel1	Binary coded selection of the switch-off position 1 ... 3 • Bit 5 and bit 6 of the control word.
Bit 15	JogCtrlInputSel2	

8.5.5 Setting parameters (short overview)

Parameters	Info	Lenze setting	
		Value	Unit
C00011	Appl.: Reference speed	1500	rpm
C00012	Accel. time - main setpoint	2.000	s
C00013	Decel. time - main setpoint	2.000	s
C00105	Decel. time - quick stop	2.000	s
C00039/1	Preset setpoint 1	40.00	%
C00039/2	Preset setpoint 2	60.00	%
C00039/3	Preset setpoint 3	80.00	%
C00039/4...15	Fixed setpoint 4 ... 15	0.00	%
C00101/1...15	Add. accel. time 1 ... 15	0.000	s
C00103/1...15	Add. decel. time 1 ... 15	0.000	s
C00105	Decel. time - quick stop	2.000	s
C00106	Auto-DCB: Hold time	0.500	s
C00107	DCB braking: Hold time	999.000	s
C00134	L_NSet_1: Ramp smoothing	0: Off	
C00182	L_NSet_1: S-ramp time PT1	20.00	s
C00190	L_NSet_1: Setpoint arithmetic	0: Out = Set	
C00220	L_NSet_1: Acceleration time - add. setpoint	0.000	s
C00221	L_NSet_1: Deceleration time - add. setpoint	0.000	s
C00241	L_NSet_1: Hyst. NSet reached	0.50	%
C00488/1	InputSens.SlowDown1	0: Level	
C00488/2	InputSens.Stop1	0: Level	
C00488/3	InputSens.SlowDown2	0: Level	
C00488/4	InputSens.Stop2	0: Level	
C00488/5	InputSens.SlowDown3	0: Level	
C00488/6	InputSens.Stop3	0: Level	
C00632/1	L_NSet_1: Blocking speed 1 max	0.00	%
C00632/2	L_NSet_1: Blocking speed 2 max	0.00	%
C00632/3	L_NSet_1: Blocking speed 3 max	0.00	%
C00633/1	L_NSet_1: Blocking speed 1 min	0.00	%
C00633/2	L_NSet_1: Blocking speed 2 min	0.00	%
C00633/3	L_NSet_1: Blocking speed 3 min	0.00	%
C00635	L_NSet_1: nMaxLimit	199.99	%
C00636	L_NSet_1: nMinLimit	-199.99	%
C00670	L_OffsetGainP_1: Gain	1.0000	
C00671	L_OffsetGainP_2: Gain	1.0000	

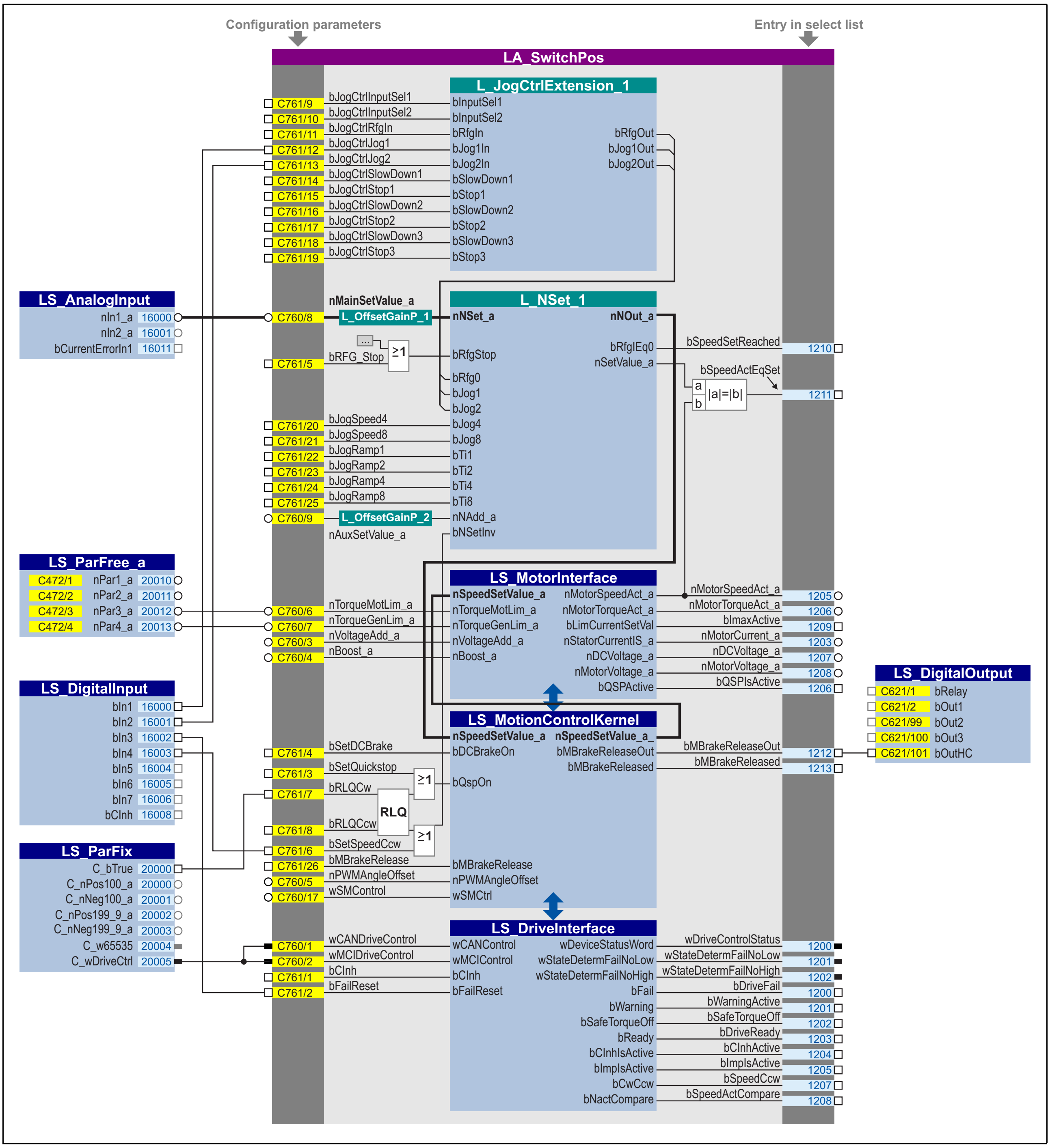
Parameters	Info	Lenze setting	
		Value	Unit
C00672	L_OffsetGainP_3: Gain	1.0000	
C00696	L_OffsetGainP_1: Offset	0.00	%
C00697	L_OffsetGainP_2: Offset	0.00	%
C00698	L_OffsetGainP_3: Offset	0.00	%
C00800	L_MPot_1: Upper limit	100.00	%
C00801	L_MPot_1: Lower limit	-100.00	%
C00802	L_MPot_1: Acceleration time	10.0	s
C00803	L_MPot_1: Deceleration time	10.0	s
C00804	L_MPot_1: Inactive fct.	0: Retain value	
C00805	L_MPot_1: Init fct.	0: Load last value	
C00806	Use of motor potentiometer	0: No	
C02610/2	MCK: Ramp time synchr. setpoint	2.000	s
C02611/1	MCK: Pos. max. speed	199.99	%
C02611/2	MCK: Pos. min. speed	0.00	%
C02611/3	MCK: Neg. min. speed	0.00	%
C02611/4	MCK: Neg. max. speed	199.99	%

Related topics:

▶ "GeneralPurpose" functions (📖 571)

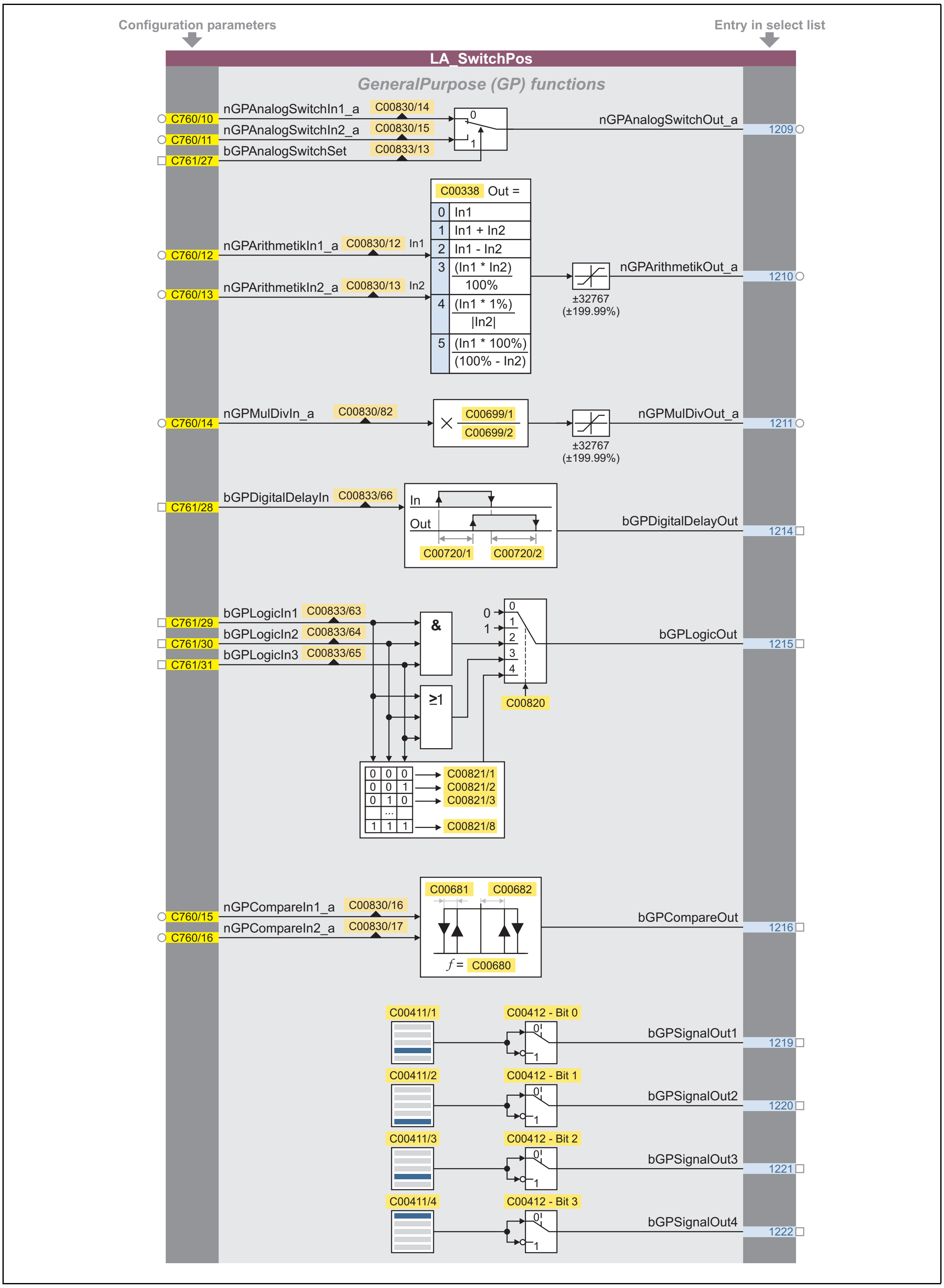
8.5.6 Configuration parameters

If required, the subcodes of C00760 and C00761 serve to change the pre-configured assignment of the application inputs:

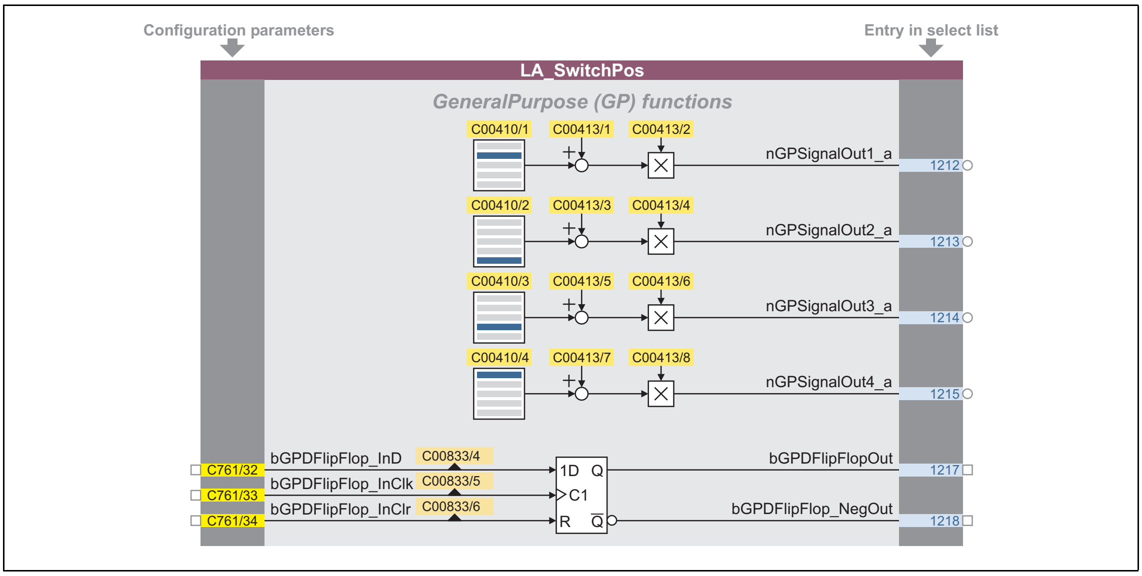


[8-19] Pre-assignment of the "Switch-off positioning" application in the "Terminals 0" control mode

Configuration parameters for "GeneralPurpose" functions



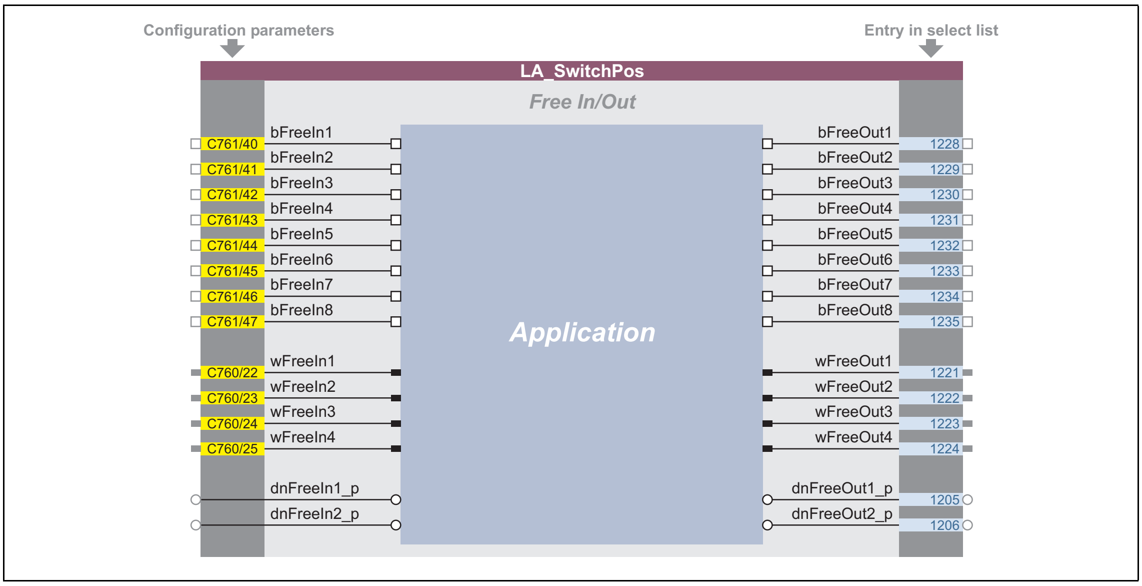
[8-20] "GeneralPurpose" functions



[8-21] "GeneralPurpose" functions (continuation)

Free inputs and outputs

These inputs can be freely interconnected in the application level. They can be used to transfer signals from the I/O level to the application level and vice versa.



[8-22] Free inputs/outputs

Related topics:

- User-defined terminal assignment (📖 445)
- "GeneralPurpose" functions (📖 571)

8.6 "GeneralPurpose" functions

Each technology application provides different free logic and arithmetic functions, so-called "GeneralPurpose" functions.

For the interconnection of these functions, the application block features inputs and outputs on the I/O level, which are linked to the logic/arithmetic function.

Note!

In the Lenze setting, the connectors for the "GeneralPurpose" functions are hidden in the function block editor.

- These connections can be shown via the **Connector visibilities** command in the *Context menu* of the application block.

Tip!

The inputs of the "GeneralPurpose" functions can also be linked to other output signals via the configuration parameters of the technology application.

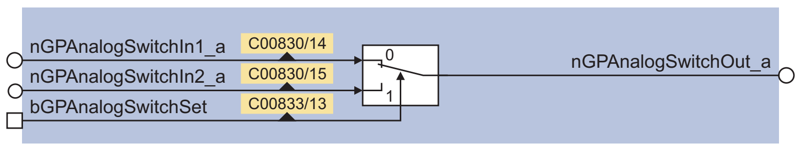
On the other hand, the outputs of the "GeneralPurpose" functions can be selected in the configuration parameters of other inputs.

Related topics:

▸ User-defined terminal assignment (📖 445)
▸ TA "Actuating drive speed": Configuration parameters (📖 478)
▸ TA "Table positioning": Configuration parameters (📖 541)
▸ TA "Switch-off positioning": Configuration parameters (📖 568)

8.6.1 Analog switch

This function switches between two analog input signals. The switch-over is controlled by a boolean input signal.

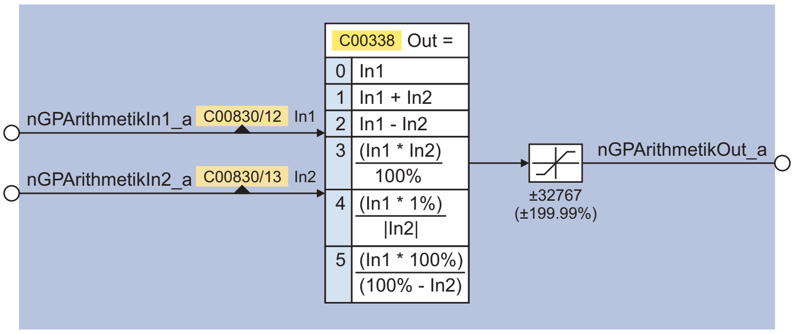


[8-23] GeneralPurpose function "Analog switch"

- For a detailed functional description see FB L_AnalogSwitch.

8.6.2 Arithmetic

This function links two analog signals arithmetically. The arithmetic function can be parameterised.



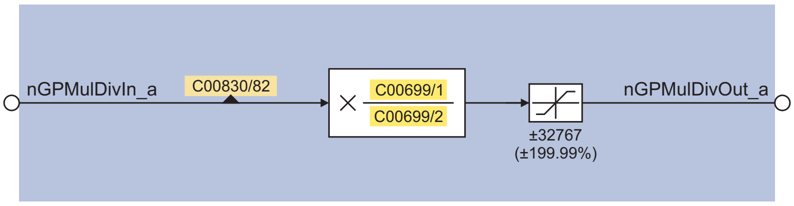
[8-24] GeneralPurpose function "Arithmetic"

Parameters	Info	Lenze setting Value	Unit
C00338	L_Arithmetik_1: Function	0: nOut_a = nIn1_a	

- For a detailed functional description see the L_Arithmetik FB.

8.6.3 Multiplication/Division

This function multiplies an analog input signal with a parameterisable factor. The factor must be selected in the form of a quotient (numerator and denominator).



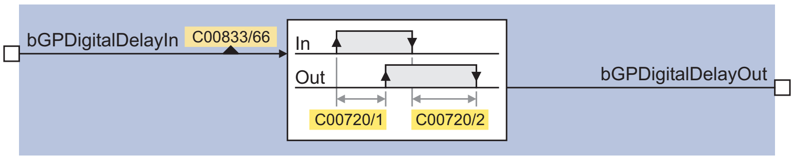
[8-25] GeneralPurpose function "Multiplication/division"

Parameters	Info	Lenze setting Value	Unit
C00699/1	L_MulDiv_1: Numerator	0	
C00699/2	L_MulDiv_1: Denominator	10000	

- For a detailed functional description see FB L_MulDiv.

8.6.4 Binary delay element

This function timely delays binary signals. On-delay and off-delay can be parameterised separately.



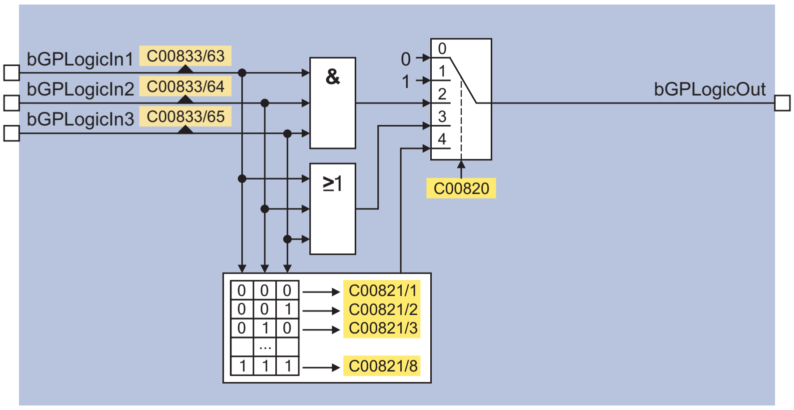
[8-26] GeneralPurpose function "Binary delay element"

Parameters	Info	Lenze setting Value	Unit
C00720/1	L_DigitalDelay_1: On delay	0.000	s
C00720/2	L_DigitalDelay_1: Off delay	0.000	s

- For a detailed functional description see FB L_DigitalDelay.

8.6.5 Binary logic

This function provides a binary output signal which is formed by a logic operation of the input signals. Alternatively, you can also select a fixed binary value which is independent of the input signals.



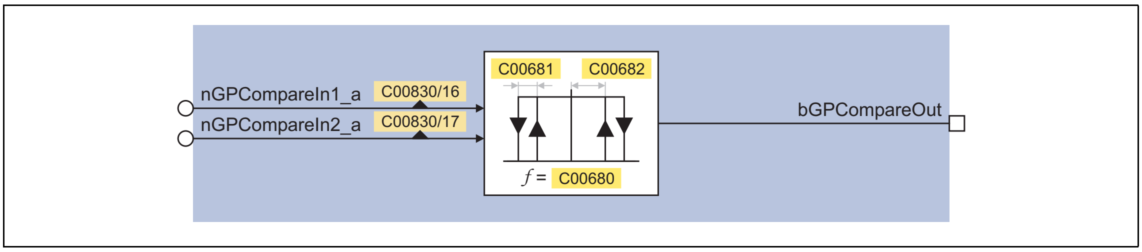
[8-27] GeneralPurpose function "Binary logic"

Parameters	Info	Lenze setting Value	Unit
C00820	L_DigitalLogic_1: Function	0: bOut = 0	
C00821/1	bIn1=0/bIn2=0/bIn3=0	0: FALSE	
C00821/...	...	...	
C00821/8	bIn1=1/bIn2=1/bIn3=1	0: FALSE	

- For a detailed functional description see FB L_DigitalLogic.

8.6.6 Analog comparison

This function compares two analog signals and can be used e.g. to realise a trigger. The comparison operation, hysteresis and window size can be parameterised.



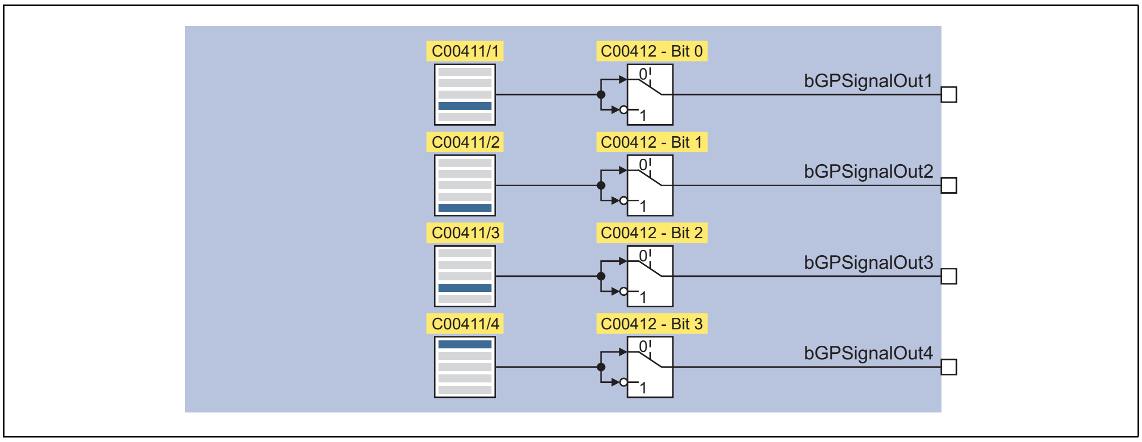
[8-28] GeneralPurpose function "Analog comparison"

Parameters	Info	Lenze setting Value	Unit
C00680	L_Compare_1: Fct.	6: |In1| < |In2|	
C00681	L_Compare_1: Hysteresis	0.50	%
C00682	L_Compare_1: Window	2.00	%

- For a detailed functional description see FB L_Compare.

8.6.7 Binary signal monitor

This function serves to output four binary signals selected from a list of all binary output signals available in the inverter. You can set an inversion of the output signals.



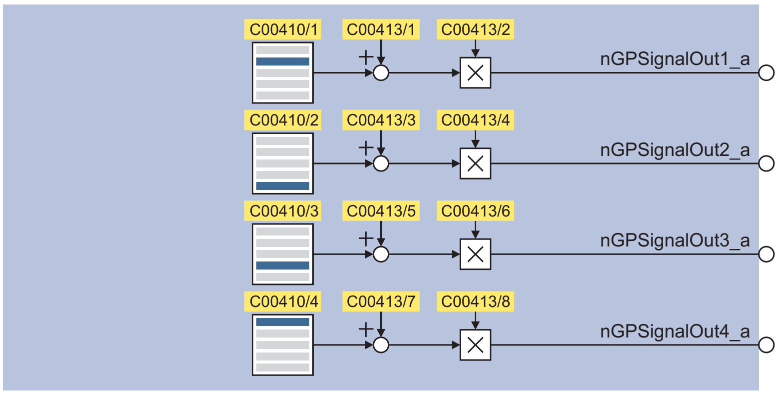
[8-29] GeneralPurpose function "Binary signal monitor"

Parameters	Info	Lenze setting Value	Unit
C00411/1...4	L_SignalMonitor_b: Signal 1 ... 4	0: Not connected	
C00412	L_SignalMonitor_b: Inversion	Bit coded	

- For a detailed functional description see FB L_SignalMonitor_b.

8.6.8 Analog signal monitor

This function serves to output four analog signals selected from a list of all analog output signals available in the inverter. Offset and gain of the source signals can be adjusted.



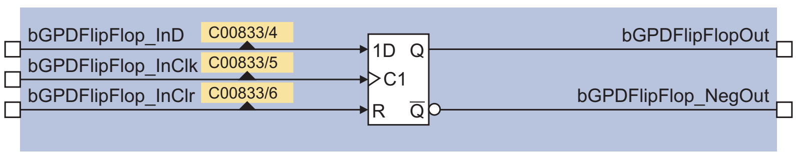
[8-30] GeneralPurpose function "Analog signal monitor"

Parameters	Info	Lenze setting	
		Value	Unit
C00410/1...4	L_SignalMonitor_a: Signal 1 ... 4	0: Not connected	
C00413/1	L_SignalMonitor_a: Signal 1 offset	0.00	%
C00413/2	L_SignalMonitor_a: Signal 1 gain	100.00	%
C00413/3	L_SignalMonitor_a: Signal 2 offset	0.00	%
C00413/4	L_SignalMonitor_a: Signal 2 gain	100.00	%
C00413/5	L_SignalMonitor_a: Signal 3 offset	0.00	%
C00413/6	L_SignalMonitor_a: Signal 3 gain	100.00	%
C00413/7	L_SignalMonitor_a: Signal 4 offset	0.00	%
C00413/8	L_SignalMonitor_a: Signal 4 gain	100.00	%

- For a detailed functional description see the L_SignalMonitor_a FB.

8.6.9 D-FlipFlop

This function saves the logic status of the data input (1D) in case of an active clock edge at the clock input (C1) and puts out its value in sequence at the output Q. If there is no active clock edge, the input value is not accepted.

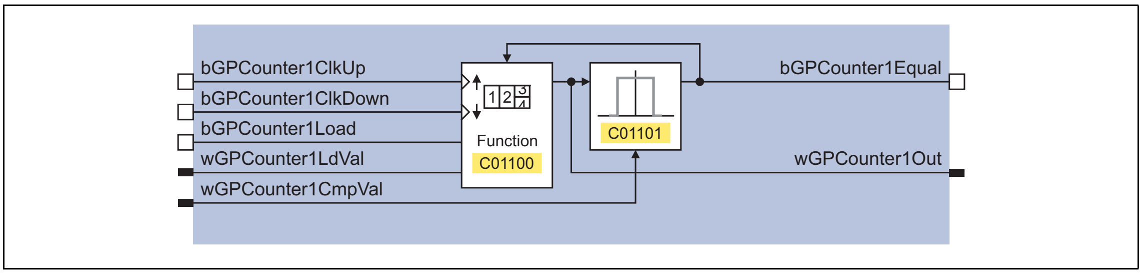


[8-31] GeneralPurpose function "D-FlipFlop" (clock-edge controlled)

- For a detailed functional description see FB L_DFlipFlop.

8.6.10 Counter

This function is a digital upcounter and downcounter with a comparison operation.



[8-32] GeneralPurpose function "Counter"

Parameters	Info	Lenze setting Value | Unit
C01100/1	L_Counter_1: Function	0: Normal counting
C01101/1	L_Counter_1: Comparison	0: Greater than or equal to

- Only available with TA "Table positioning".
- For a detailed functional description see FB L_Counter.

9 Basic drive functions (MCK)

In this chapter, the standard and basic drive functions integrated in the **Motion Control Kernel** (MCK) of the 8400 TopLine are described to which the active technology application can gain access via defined internal interfaces. As a result, the time-consuming creation of individual FB interconnections is avoided and the amount of work and complexity involved in the implementation of standard functions is minimised.

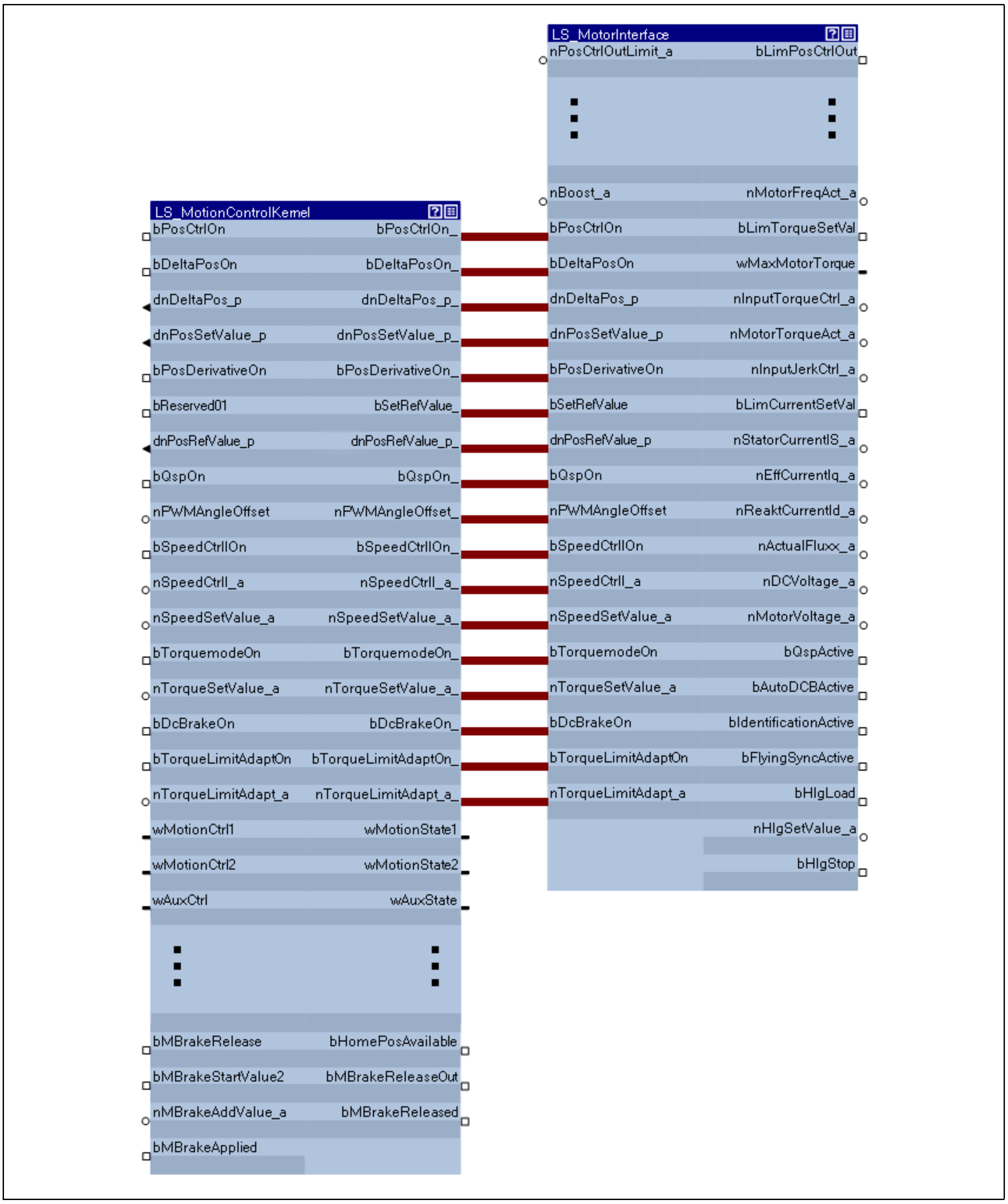
In the **Motion Control Kernel**, for example, an automatic holding brake control function is integrated which controls the holding brake in relation to the speed setpoint and various other internal control signals. Due to integrated automatic brake operation, the user is relieved of the task of managing these control signals.

Other standard functions integrated in the **Motion Control Kernel** for the 8400 TopLine inverter are, for example:

- Manual jog, e.g. for manual setting-up operation
- Homing for positioning functions of the drive
- Position profile generator for positioning functions of the drive

9.1 Basic signal flow

The **Motion Control Kernel** is connected between setpoint generator (e.g. ramp generator, PID process controller, etc.) and the motor control function in the case of the available technology applications. For problem-free interaction of the **Motion Control Kernel** and motor control function, the two associated system blocks LS_MotionControlKernel and LS_MotorInterface have interfaces with corresponding inputs/outputs. These are visible in the FB Editor for monitoring purposes and must be connected to each other:



[9-1] Interconnection of Motion Control Kernel and motor control function

In the interconnection previously shown, the **Motion Control Kernel** monitors every interface. Some of the signals such as a quick stop request or a DC-injection braking request are directly passed through to the motor control. However, other signals are passed through or modified depending on the operating mode (e.g. synchronising a setpoint selection via ramp function).

9.2 Internal interfaces | System block "LS_MotionControlKernel"

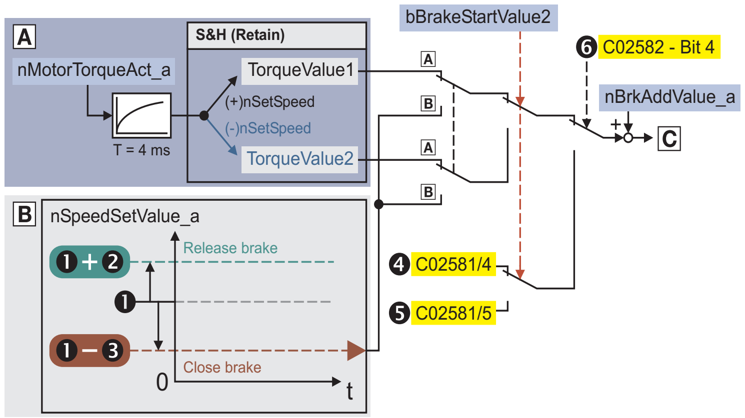
In the Function Block editor, the system block **LS_MotionControlKernel** provides the interfaces to the **Motion Control Kernel**.

inputs

Designator	Data type	Information/possible settings	
Control and setpoint signals for motor control The purpose of the following inputs is to transfer control signals and setpoints to the internal motor control function (LS_MotorInterface).			
bPosCtrlOn	BOOL	Activation of position/angle control	
		FALSE	Position/angle control deactivated.
		TRUE	Position/angle control activated.
bDeltaPosOn	BOOL	Activate position difference as setpoint selection • In order to position the motor shaft, the position control function can work within the motor control function with the absolute position setpoint *dnPosSetValue_p* or alternatively with the speed setpoint *nSpeedSetValue_a* and the position difference *dnDeltaPos_p*.	
		FALSE	Positioning with position setpoint *dnPosSetValue_p*.
		TRUE	Positioning with speed setpoint *nSpeedSetValue_a* and position difference *dnDeltaPos_p*. **Note:** In this case, no position encoder should be set to ensure a correct function (C00490 = "No encoder: nSpeedSetValue_a").
dnDeltaPos_p	DINT	Position difference (following error input) • Difference between setpoint position and actual position in [increments] • Is used for position control if *bDeltaPosOn* = TRUE. • Scaling: A revolution is displayed with 65536 increments or steps.	
dnPosSetValue_p	DINT	Absolute position setpoint in [increments] • Is used for position control if *bDeltaPosOn* = FALSE. • Scaling: A revolution is displayed with 65536 increments or steps.	
bPosDerivativeOn	BOOL	Create a setpoint for the speed controller from the position setpoint • For highly dynamic control systems, the setpoint for the speed controller can be created from the absolute position setpoint *dnPosSetValue_p* instead of the speed setpoint *nSpeedSetValue_a* ▶ Position control/additive speed specification	
		TRUE	Create a speed setpoint from the position setpoint. • The absolute position setpoint *dnPosSetValue_p* is differentiated and a speed value is created which is the setpoint for the speed controller. • Internal limitation of 65536 increments/ms.
bReserved01	BOOL	Reserved	
dnPosRefValue_p	DINT	Home position in [increments] • Value is adopted when control bit 9 ("HomeSetPos") is set from "0" to "1" in the MCK control word and is sent to the motor control function via the *dnPosRefValue_p_* output. • Transfer is possible in any operating mode ("Homing on the fly"). • For this function, the *dnPosRefValue_p_* output must be connected to the input of the same name of the SB LS_MotorInterface.	

Designator Data type		Information/possible settings
bQspOn BOOL		Trigger quick stop (QSP) via the MCK • Also see device command "Activate/deactivate quick stop".
	TRUE	Activate quick stop • Motor control is decoupled from the setpoint selection and, within the deceleration time parameterised in C00105, the motor is brought to a standstill (n_{act} = 0). • A pulse inhibit is set if the auto-DCB function has been activated via C00019. • The motor is kept at a standstill during closed-loop operation (function in preparation).
	FALSE	Deactivate quick stop • The quick stop is deactivated if no other source for the quick stop is active. • C00159 displays a bit code of active sources/causes for the quick stop.
nPWMAngleOffset INT		Angular offset input • Scaling: 16384 ≡ 100 % • Setting range: 0 ... 199.99 %
bSpeedCtrlIOn BOOL		Directly set the I-component of speed controller • In order to statically specify a minimum torque, e.g. when a load is being lifted.
	TRUE	Se the I-component of the speed controller to the value *nSpeedCtrlI_a*.
nSpeedCtrlI_a INT		I-component of the speed controller • Value is adopted in the case of a FALSE-TRUE edge at the input *bSpeedCtrlIOn*.
nSpeedSetValue_a INT		Rotation speed/velocity setpoint
bTorquemodeOn BOOL	TRUE	Switch on torque-controlled operation
nTorqueSetValue_a INT		Torque setpoint
bDcBrakeOn BOOL		Manual DC-injection braking (DCB) • For this function, the *bDcBrakeOn_* output signal must be connected to the *bDcBrakeOn* input of the same name of the LS_MotorInterface system block. • Detailed information on DC-injection braking is provided in the motor control chapter, subchapter "DC-injection braking".
		Note! Holding braking is not possible when this braking mode is used! Use the basic "Holding brake control" function for controlling the holding brake with a low rate of wear.
	FALSE	Deactivate DC-injection braking.
	TRUE	Activate DC-injection braking, i.e. the drive is brought to a standstill by means of DC-injection braking. • The braking effect stops when the rotor is at standstill. • After the hold time (C00107) has expired, the controller sets the pulse inhibit.
bTorqueLimitAdaptOn BOOL		Adaptation of torque limitation
	TRUE	Activate adaptation of torque limitation.
nTorqueLimitAdapt_a INT		Value for adaptation of torque limitation • Scaling: 16384 ≡ 100 % of the torque limit specified at the LS_MotorInterface system block via the *nTorqueMotLimit_a* and *nTorqueGenLimit_a* inputs.

Designator Data type	Information/possible settings
Control words	
MCK: wMotionCtrl1 wMotionCtrl2 WORD	MCK control word 1 & 2 • The two control words together form a 32-bit double control word with which the entire Motion Control Kernel is controlled. • All motion profiles in the different operating modes can be operated via this interface. • See the "MCK control word" subchapter for a detailed description of the individual control bits. • Display parameter: C01240
wAuxCtrl WORD	For future extensions - Input has no function at present time! Additional control word
wSMCtrl WORD	Interface to the optional safety system. • Setting control bit 0 ("SafeStop1") in this control word causes e.g. the automatic deceleration of the drive to standstill within this application (in the **Motion Control Kernel**). • See the "Interface to safety system" subchapter for a detailed description of the individual control bits.
Control and setpoint signals for Motion Control Kernel function	
dnProfilePosition_p DINT	Profile position in [increments] • Position which is to be entered into a profile data set selected via the MCK control word. • In the "absolute" mode, this position is a target position. However, in the "relative" mode it is a relative traverse path. ▸ Consideration of residual value in case of external profile calculation
nSpeedAddValue_v INT	Additive speed setpoint in [inc/ms]
nSpeedOverride_a INT	Value for Speed override • Percentage multiplier (0 ... 199.99 %) for the currently active speed. • 16384 ≡ 100 % of the maximum traversing speed (display in C01211/1). • If the override value is 0 %, the drive is brought to a standstill.
nAccOverride_a INT	Value for Acceleration override • Percentage multiplier (0 ... 199.99 %) for the currently active acceleration. • 16384 ≡ 100 % of the parameterised acceleration of the corresponding operating mode. • If the override value is 0 %, acceleration ceases.
nSRampOverride_a INT	Value for S-ramp smoothing override • Percentage multiplier (0 ... 100 %) for the currently active acceleration. • 16384 ≡ 100 % of the parameterised S-ramp time (C01306/1...15). • Values > 16384 are ignored.
	Note! If the *nSRampOverride_a* input remains unconnected or if an override value of "0 %" is selected, activation of the S-ramp override causes deactivation of the S-ramp time. • Deactivation of the S-ramp time before the start of a profile with S-ramp time causes linear ramp generation. • Deactivation of the S-ramp time during a traversing process, however, is not accepted immediately in the profile generator, but the profile generator checks automatically when an online change of the ramp form can be carried out and then initiates it automatically.
bLimitSwitchPos BOOL	Input for Hardware limit switches (positive)
bLimitSwitchNeg BOOL	Input for Hardware limit switches (negative)
bHomingMark BOOL	Input for pre-stop mark/pre-stop signal for homing • Relevant for homing modes "4" ... "7".

<table>
<tr><th>Designator
Data type</th><th colspan="2">Information/possible settings</th></tr>
<tr><td rowspan="3">bMBrakeRelease
BOOL</td><td colspan="2">Holding brake control:
Releasing/applying the brake in connection with the selected operating mode</td></tr>
<tr><td>FALSE</td><td>Apply brake.
• During automatic operation, the internal brake logic controls the brake.</td></tr>
<tr><td>TRUE</td><td>Release brake manually (forced release).
• Note!
The brake can also be released when the controller is inhibited!
• During automatic operation, the internal brake logic is deactivated and the brake is released (supervisor operation). If a controller inhibit has been set by the brake control, it will be deactivated.
• In semi-automatic operation, the brake is released including feedforward control.</td></tr>
<tr><td rowspan="4">bMBrakeStartValue2
BOOL</td><td colspan="2">Holding brake control:
Selection of the torque feedforward control value
▸ Feedforward control of the motor before release</td></tr>
<tr><td>FALSE</td><td>Starting value 1 is active (see the following figure).</td></tr>
<tr><td>TRUE</td><td>Starting value 2 is active (see the following figure).</td></tr>
<tr><td colspan="2">Creation of the feedforward control value for the release process of the brake:
A Signal path for motor control <u>with</u> feedback
B Signal path for motor control <u>without</u> feedback
C Feedforward control value


❶ C02581/1: Switching threshold
❷ C02581/2: Hysteresis for release
❸ C02581/3: Hysteresis for application
❹ C02581/4: Starting value 1 for manual feedforward control
❺ C02581/5: Starting value 2 for manual feedforward control
❻ C02582/Bit 4: Selection of the feedforward control value (automatic/manual selection)</td></tr>
<tr><td>nMBrakeAddValue_a
INT</td><td colspan="2">Holding brake control:
Additive feedforward control value (speed or torque) in [%] for torque feedforward control when the respective control mode is started
• For speed control: 100 % ≡ reference speed (C00011)
• For torque control: 100 % ≡ maximum torque (C00057)
▸ Feedforward control of the motor before release</td></tr>
<tr><td rowspan="3">bMBrakeApplied
BOOL</td><td colspan="2">Holding brake control:
Input for status detection via switching contacts at the brake
• Only effective if bit 5 in C02582 is set to "1".</td></tr>
<tr><td>FALSE</td><td>Brake is released.</td></tr>
<tr><td>TRUE</td><td>Brake is applied.</td></tr>
</table>

Designator Data type	Information/possible settings
dnPosSetCycle_p DINT (from version 17.00.00)	Define the cycle length for cycle-related position setpoint • The clock signal at *dnPosSetValue_p* is converted into a angular difference signal by the cycle length.
Process data interface for entering profile data	
dnProcessIn1_p .. dnProcessIn4_p DINT	Profile position as process data in [increments]
nProcessIn1_a .. nProcessIn8_a DINT	Speed, acceleration, deceleration, S-ramp time, final speed as process data in [%] • The 16 bit value has an adaptive effect on the corresponding parameters

outputs

Designator Data type	Value/meaning	
Control and setpoint signals for motor control The following outputs are used to transfer control signals and setpoints to the internal motor control function (LS_MotorInterface).		
bPosCtrlOn_ BOOL	TRUE	Activate position control.
bDeltaPosOn_ BOOL	TRUE	Control to adjust for following errors.
dnDeltaPos_p_ DINT	Following error input	
dnPosSetValue_p_ DINT	Absolute position setpoint	
bPosDerivativeOn_ BOOL	TRUE	Activate precontrol function of speed controller.
bSetRefValue_ BOOL (from version 18.00.00)	TRUE	Set actual position to home position. • Trigger signal when reference setting is requested or the reference signal is detected during the referencing procedure.
dnPosRefValue_p_ DINT	Home position	
bQspOn_ BOOL	TRUE	Activate quick stop
nPWMAngleOffset_a_ INT	Angular offset input	
bSpeedCtrlIOn_ BOOL	TRUE	Set I-component of speed controller.
nSpeedCtrlI_a_ INT	I-component of the speed controller	
nSpeedSetValue_a_ INT	Main setpoint of speed	
bTorqueModeOn_ BOOL	TRUE	Switch on torque-guided operation.
nTorqueSetValue_a_ INT	Torque setpoint	
bDcBrakeOn_ BOOL	TRUE	Activate DC-injection braking.
bTorqueLimitAdaptOn_ BOOL	TRUE	Activate adaptation of torque limitation.
nTorqueLimitAdapt_a_ INT	Value for adaptation of torque limitation	

Designator Data type	Value/meaning	
Status words		
wMotionState1 wMotionState2 WORD	MCK status word 1 & 2 • For a detailed description of the individual status bits, see subchapter entitled "MCK status word. • Display parameter: C01241	
wAuxState WORD	For future extensions - Output has no function at present time! Additional status word	
Status signal and actual-value signals from Motion Control Kernel functions		
nSpeedSet_v INT	Speed setpoint selection in [increments/ms] • 16384 ≡ 15000 rpm	
dnPosCycle_p DINT (from version 17.00.00)	Master cycle length (master value) in [increments]	
dnPosTarget_p DINT	Target position in [increments] • 65535 ≡ 1 revolution of the motor shaft	
dnPosSet_p DINT	Modulo position in [increments] • 65535 ≡ 1 revolution of the motor shaft ▸ Activation of the modulo measuring system	
dnPosSetRelative_p DINT (from version 02.00.00)	Relative feed in positioning processes in [increments] • 65535 ≡ 1 revolution of the motor shaft	
wActProfileNo WORD	Number of current profile	
wFollowProfileNo WORD	Number of sequence profile	
bReadyToOperate BOOL (From version 16.00.00)	TRUE	Readiness to process setpoint signals and setpoint commands is active
bBusy BOOL	TRUE	Setpoint profile generation is active.
bDone BOOL	TRUE	Setpoint profile generation is completed (set position = target position).
bHomingDone BOOL	TRUE	Homing has been carried out. • The *bHomingDone* output, in contract to the *bHomePosAvailable* output, remains set to TRUE even if a travel command that resulted in a reset of the home position has been carried out.
bHomePosAvailable BOOL	TRUE	Home position is known. ▸ Status bit "HomPosAvailable"
bMBrakeReleaseOut BOOL	Trigger signal for switching element holding brake control via a digital output • Use bit 0 in C02582 to activate inverted switching element triggering. ▸ Holding brake control	
	FALSE	Apply brake.
	TRUE	Release brake.
bMBrakeReleased BOOL	"Brake released" status signal considering the brake release time • When the holding brake is triggered to close, *bMBrakeReleased* is immediately set to FALSE even if the brake closing time has not yet elapsed! ▸ Holding brake control	
	TRUE	Brake released (after the brake release time has expired).
bFollowErrLim1 BOOL (from version 12.00.00 onwards)	TRUE	The current following error has exceeded the limit 1 for the following error set in C01215/1 via the time set in C01244/2. ▸ Following error monitoring system

<table>
<tr><th>Designator
Data type</th><th colspan="2">Value/meaning</th></tr>
<tr><td>bFollowErrLim2
BOOL
(from version 12.00.00 onwards)</td><td>TRUE</td><td>The current following error has exceeded the limit 2 for the following error set in C01244/3 via the time set in C01215/2.
▶ Following error monitoring system</td></tr>
<tr><td>wGearNum
wGearDenom
WORD
(from version 02.00.00)</td><td colspan="2">Output of the gearbox factor set in C01202/1 and C01202/2
• These outputs can be connected to the GearNum-/GearDenom inputs of the FBs which process a gearbox factor (FB L_PhiIntegrator_1, FB L_DFSET_1, FB L_CalcDiameter_1).
• C01067/1...3 can be used to inform these FBs that the gearbox direction of rotation is inverted is mostly done by selecting "motor mounting position inverted").</td></tr>
</table>

9.2.1 MCK control word

The motion control function implemented in the LS_MotionControlKernel system block can be controlled via the *wMotionCtrl1* and *wMotionCtrl2* control words. Together, both control words form a 32-bit control double word which serves to control the entire MotionControlKernel. All motion profiles in the various operating modes can be operated via this interface.

For direct control via a fieldbus system, the two control words can be triggered by the field bus via a port block. In addition to a few other signals (e.g. limit switch, speed override) that are connected to the digital inputs of the inverter, all control signals can therefore be activated/deactivated via the fieldbus used (CAN, PROFIBUS, etc.).

As an alternative to this, changes to the individual control bits or bit fields (e.g. for specifying the profile number) can also be carried out via separate process inputs provided by the L_MckCtrlInterface function block. The control words *wOutMckPosCtrl_1* and *wOutMckPosCtrl_2* output by this function block constitute the input information for the LS_MotionControlKernel system block in this case.

MCK control word 1 (wMotionCtrl1)

<table>
<tr><th>Bit</th><th>Designation</th><th colspan="5">Description</th></tr>
<tr><td rowspan="8">0
...
3</td><td rowspan="8">OpMode_Bit0
...
OpMode_Bit3</td><td>Operating mode</td><td>Bit 3</td><td>Bit 2</td><td>Bit 1</td><td>Bit 0</td></tr>
<tr><td>Speed follower</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Homing</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Manual jog</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>Positioning</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>Stop</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>Position follower</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td colspan="5">All other possible settings are reserved for future extensions!</td></tr>
<tr><td rowspan="5">4
5</td><td rowspan="5">ManJogPos
ManJogNeg</td><td colspan="3">Manual jog</td><td>Bit 5</td><td>Bit 4</td></tr>
<tr><td colspan="3">Stop manual jog</td><td>0</td><td>0</td></tr>
<tr><td colspan="3">Manual jog CW</td><td>0</td><td>1</td></tr>
<tr><td colspan="3">Manual jog in counter-clockwise direction</td><td>1</td><td>0</td></tr>
<tr><td colspan="3">No change from previous status</td><td>1</td><td>1</td></tr>
<tr><td>6</td><td>ManExecute 2ndSpeed</td><td colspan="5">"1" ≡ Change over to manual speed 2</td></tr>
<tr><td>7</td><td>ReleaseLimitSwitch</td><td colspan="5">"1" ≡ Retract operated hardware limit switch</td></tr>
<tr><td>8</td><td>HomStartStop</td><td colspan="5">"1" ≡ Start/stop homing process</td></tr>
<tr><td>9</td><td>HomSetPos</td><td colspan="5">"1" ≡ Set homing position</td></tr>
<tr><td>10</td><td>HomResetPos</td><td colspan="5">"1" ≡ Reset the "Reference known" status
• bHomePosDone and bHomePosAvailable are reset to FALSE.
• The positions remain unaffected.</td></tr>
<tr><td>11</td><td>EnableSpeedOverride</td><td colspan="5">"1" ≡ Activate speed override</td></tr>
<tr><td>12</td><td>EnableAccOverride</td><td colspan="5">"1" ≡ Activate acceleration override</td></tr>
<tr><td>13</td><td>Enable SRampOverride</td><td colspan="5">"1" ≡ Activate S-ramp override</td></tr>
<tr><td>14</td><td>PosTeachSetPos</td><td colspan="5">"1" ≡ Teach MCK set position into the selected profile</td></tr>
<tr><td>15</td><td>PosTeachActPos</td><td colspan="5">"1" ≡ Teach current position into the selected profile</td></tr>
</table>

MCK control word 2 (wMotionCtrl2)

<table>
<tr><th>Bit</th><th>Designation</th><th colspan="9">Description</th></tr>
<tr><td>16</td><td>PosExecute</td><td colspan="9">"0↗1" ≡ Start travelling</td></tr>
<tr><td>17</td><td>PosFinishTarget</td><td colspan="9">"0↗1" ≡ Complete cancelled profile</td></tr>
<tr><td>18</td><td>PosDisable FollowProfile</td><td colspan="9">"1" ≡ Do not travel sequence profile</td></tr>
<tr><td>19</td><td>PosStop</td><td colspan="9">"1" ≡ Cancel travelling
From version 02.00.00, more travel requests will be inhibited ("PosExecute" will be blocked).</td></tr>
<tr><td rowspan="13">20
...
23</td><td rowspan="13">PosModeBit0
...
PosModeBit3</td><td colspan="5">Positioning mode</td><td>Bit 3</td><td>Bit 2</td><td>Bit 1</td><td>Bit 0</td></tr>
<tr><td colspan="5">Positioning mode = setting in C01300/1...15</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="5">Absolute (shortest path)</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td colspan="5">Continuous</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td colspan="5">Relative</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="5">absolute (Cw)</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td colspan="5">absolute (Ccw)</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td colspan="5">Absolute (shortest path) to TP</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="5">Continuous to TP</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td colspan="5">Relative to TP</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td colspan="5">Absolute (Cw) on TP</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="5">Absolute (Ccw) on TP</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td colspan="9">All other possible settings are reserved for future extensions!</td></tr>
<tr><td rowspan="7">24
...
31</td><td rowspan="7">ProfileNo_Bit0
...
ProfileNo_Bit7</td><td>Profile</td><td>Bit 7</td><td>Bit 6</td><td>Bit 5</td><td>Bit 4</td><td>Bit 3</td><td>Bit 2</td><td>Bit 1</td><td>Bit 0</td></tr>
<tr><td>Profile 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Profile 1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Profile 2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>...</td><td colspan="8">...</td></tr>
<tr><td>Profile 15</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td colspan="9">All other possible settings are reserved for future extensions!</td></tr>
</table>

Note!

The profile 0 is no valid profile for the "Positioning" operating mode.

If a driving request with an invalid profile number is started, the response set in C00595/12 occurs (Lenze setting: "WarningLocked").

Tip!

Travel requests/profiles can also be started while the drive is running. The drive does not need to be at standstill.

9.2.2 MCK status word

MCK status word 1 (wMotionState1)

Bit	Designation	Description				
0 ... 3	ActOpModeBit00 ... ActOpModeBit03	**Active operating mode**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
		Speed follower	0	0	0	0
		Homing	0	0	0	1
		Manual jog	0	0	1	0
		Positioning	0	0	1	1
		Stop or Safe stop 1 (SS1)	0	1	0	0
		Position follower	0	1	0	1
		StandBy (internal operating mode in the event of quick stop, pulse inhibit and DC-injection braking)	1	1	1	1
4	Busy	"1" ≡ Internal profile generation is active. A speed profile is being generated.				
5	Done	"1" ≡ Generation of a speed profile with the selected position has been completed.				
6	AcceleratingActive	"1" ≡ Profile generation phase is in the acceleration process.				
7	ConstSpeedDuty	"1" ≡ Profile generation phase at constant speed active.				
8	DeceleratingActive	"1" ≡ Profile generation phase is in the deceleration process.				
9	S_ShapingActive	From version 12.00.00: "1" ≡ Rounding during acceleration/deceleration active. ▸ Status bit "S_ShapingActive"				
10	Pos. HW-Limit Detected	"1" ≡ Positive limit switch has triggered. • Reset only possible via"Manual jog" mode! ▸ Hardware limit switches				
11	Neg. HW-Limit Detected	"1" ≡ Negative limit switch has triggered. • Reset only possible via"Manual jog" mode! ▸ Hardware limit switches				
12	HomPosDone	"1" ≡ Homing has been completed.				
13	HomPosAvailable	"1" ≡ The home position has been detected and is known in the drive. ▸ Status bit "HomPosAvailable"				
14	Pos. SW limit detected	"1" ≡ Positive software limit position overtravelled. ▸ Software limit positions				
15	Neg. SW limit detected	"1" ≡ Negative software limit position overtravelled. ▸ Software limit positions				

Note!

The internal "StandBy" operating mode is assumed if controller inhibit, pulse inhibit, quick stop and/or DC-injection braking are activated.

- In this operating mode, no setpoint generation takes place through the **Motion Control Kernel**.
- If the holding brake control sets a controller inhibit when the holding brake is closed, the internal "StandBy" operating mode is not assumed.

MCK status word 2 (wMotionState2)

<table>
<tr><th>Bit</th><th>Designation</th><th colspan="9">Description</th></tr>
<tr><td>16</td><td>DwellTime</td><td colspan="9">"1" ≡ Dwell time after reaching the setpoint position is active.
Target position monitoring (status "drive in target")</td></tr>
<tr><td>17</td><td>InTarget</td><td colspan="9">"1" ≡ Dwell time has expired and current actual position is in the set target window.
Target position monitoring (status "drive in target")</td></tr>
<tr><td>18</td><td>PosDone</td><td colspan="9">"1" ≡ Positioning profile has been completed in the "Positioning" or "Homing" mode. Setpoint position of a profile data set is in target.</td></tr>
<tr><td>19</td><td>ReadyToOperate</td><td colspan="9">"1" ≡ Readiness to process setpoint signals and setpoint commands.
Status bit "bReadyToOperate"</td></tr>
<tr><td rowspan="12">20
...
23</td><td rowspan="12">ActPosMode_Bit00
...
ActPosMode_Bit03</td><td colspan="5">Active positioning mode</td><td>Bit 3</td><td>Bit 2</td><td>Bit 1</td><td>Bit 0</td></tr>
<tr><td colspan="5">Absolute (shortest path)</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td colspan="5">Continuous</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td colspan="5">Relative</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="5">absolute (Cw)</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td colspan="5">absolute (Ccw)</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td colspan="5">Absolute (shortest path) to TP</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="5">Continuous to TP</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td colspan="5">Relative to TP</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td colspan="5">Absolute (Cw) on TP</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="5">Absolute (Ccw) on TP</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td colspan="9">Note: The display of the active positioning mode depends on further factors. See subchapter "Override of the parameterised positioning mode".</td></tr>
<tr><td rowspan="6">24
...
31</td><td rowspan="6">ActProfileNo_Bit00
...
ActProfileNo_Bit07</td><td>Active profile</td><td>Bit 7</td><td>Bit 6</td><td>Bit 5</td><td>Bit 4</td><td>Bit 3</td><td>Bit 2</td><td>Bit 1</td><td>Bit 0</td></tr>
<tr><td>Profile 0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Profile 1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Profile 2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>...</td><td colspan="8">...</td></tr>
<tr><td>Profile 15</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table>

Note!

The effective positioning mode is generated via the MCK control word (bit 20 ... 23) and/or the parameters C01300/1...15 for defining the positioning mode in the profile data. Here, the setting in the MCK control word superimposes the mode selection via the corresponding parameter. This means that the mode selection and the touch probe enable are possible via process data.

For generating the effective positioning mode, the following applies:

- Valid PosMode in the MCK control word:
  → Use positioning mode of the MCK control word
- PosMode in the MCK control word = 0:
  → Use positioning mode set in C01300/x
- Invalid PosMode in the MCK control word:
  → Error message "Ck09: Positioning mode invalid"

9.2.2.1 Status bit "HomPosAvailable"

Bit 13 ("HomPosAvailable") in the MCK status word 1 shows that the reference position has been detected and is known in the drive.

An available reference information (bit 13 = "1") and thus a valid measuring system is required for the following functions:

- Positioning in the "Positioning" operating mode in the following (C01300/1...15) positioning modes:
  - 1: absolute (beeline)
  - 4: absolute (Cw)
  - 5: absolute (Ccw)
  - 8: absolute (shortest path) to TP
  - 11: absolute (Cw) to TP
  - 12: absolute (Ccw) to TP
- Response to set software limit positions
- Stopping on breakpoints in the "Manual jog" operating mode
- Traversing of a sequence profile with absolute measuring reference in the "Homing" mode after homing is completed

Resetting the reference information

The following events reset bit 13 ("HomPosAvailable") in the MCK status word 1:

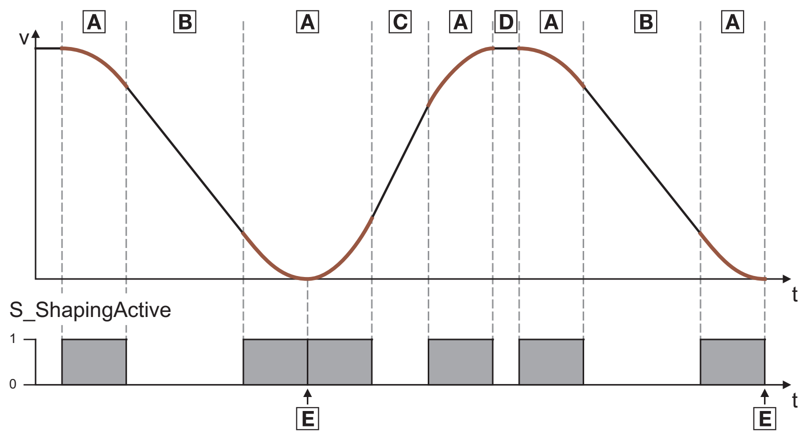
- Setting bit 10 ("HomResetPos") in the MCK control word 1
- Overtravelling the 32-bit position display area at the limit ±2147483647 increments when the cycle (C1201/1) = "0 units".
- Change of the position encoder selection in C00490
- Change of the cycle in C1201/1
- Writing on the machine parameters if bit 1 ("delete reference when machine data is changed") is set in C2652.
  - Machine parameters are cycle (C1201/1), gearbox factors (C1202/1...2, C1203/1...2) and feed constant (C1204).
  - The mere writing of these parameters (even when the previous value is used) causes the reference to be deleted!
- Encoder error when a position encoder is used at the multi-encoder interface
- Resolver error when a resolver is used as position encoder
- Mains recovery at a cycle set unlike "0 units" in C1201/1 and if the position difference at reference recovery is higher than 1000 cycles.

9.2.2.2 Status bit "S_ShapingActive"

This function extension is available from version 12.00.00!

Bit 9 ("S_ShapingActive") in the MCK status word 1 indicates in the operating modes "Homing", "Manual jog", "Positioning", "Stop" and "Position follower" when rounding is effective while S-shaped profiles are being carried out.

- Bit 9 is usually reset in case of profile restarts and after profiles have been completed.
- The following illustration shows this by means of a real temporal characteristic:



A Rounding active
B Constant deceleration
C Constant acceleration
D Constant speed
E Profile completed (bit 9 is reset in case of profile restart and profile completion)

[9-2] Example: Display of the rounding generated via the "S_ShapeActive" status bit

9.2.2.3 Status bit "bReadyToOperate"

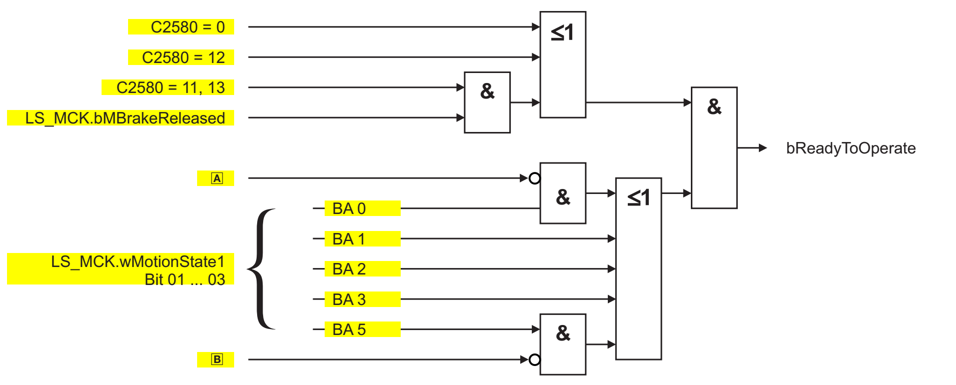
This function extension is available from version 16.00.00!

The **LS_MotionControlKernel** with status bit 19 in the status word 2 (wMotionState2) provides the signalling for the readiness to process the setpoint signals or setpoint commands. The state of bit *bReadyToOperate* provides TRUE if, for instance, a defined speed setpoint or a command (e.g. "manual jog positive") can be directly processed by the **LS_MotionControlKernel**.

The status of this bit depends on three factors:

- Operating mode of the MotionControlKernel
  - *bReadyToOperate* = FALSE if operating mode **Standby** or **Stop**
- Synchronisation status
  - *bReadyToOperate* = FALSE if **LS_MotionControlKernel** executes a synchronisation process in the operating modes **Speed follower** or **Position follower**.

The figure shows the logical relationship:



A Speed follower: Synchronisation active
B Position follower: Synchronisation active
BA 0 ... 5: Operating mode 0 ... 5

[9-3] Creation of the status bit *bReadyToOperate*

9.2.3 MCK diagnostic word

The MCK diagnostic word provides information derived by parameter settings (e.g. effective limitations etc.).

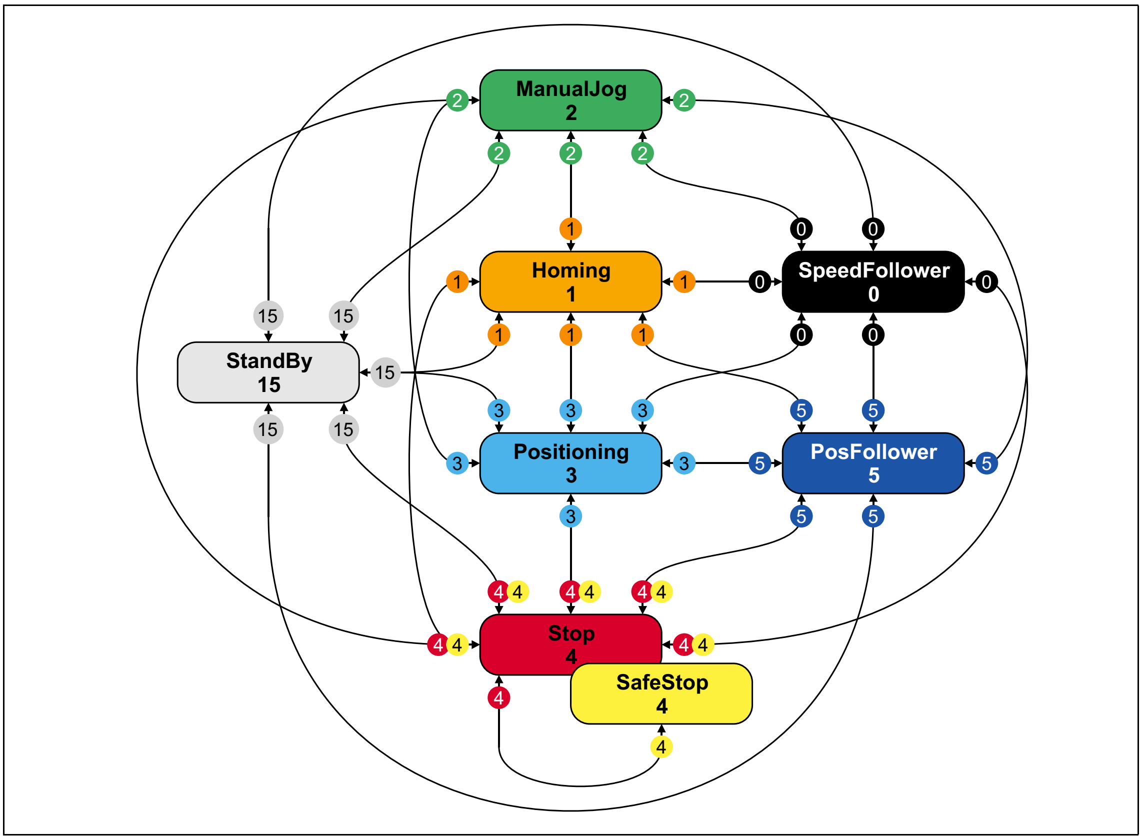
MCK: Diagnostic word

Bit	Designation	Description
Bit 0	Traversing range: Limited	1 ≡ The traversing range is limited: • The software limit positions are set. • The reference is known. • There is no modulo measuring system available (C01201/1 = 0). **Note:** This status does not consider whether the software limit positions are activated or deactivated by the settings in the prevailing operating mode.
Bit 1	Traversing range: Modulo	1 ≡ A modulo measuring system is available as a cycle length has been set in C01201/1.
Bit 2	Pos. HW limit switch active	Evaluation of travel range limit switch active in positive direction
Bit 3	Neg. HW limit switch active	Evaluation of travel range limit switch active in negative direction
Bit 4	Pos. SW-Limit active	Evaluation of software limit positions active in positive direction
Bit 5	Neg. SW-Limit active	Evaluation of software limit positions active in negative direction
Bit 6	Reserved	
Bit 7	Reserved	
Bit 8	SpeedNormConstant : Low-Limit	1 ≡ Due to the set machine parameters, an internal calculation constant is limited to a possible minimum. Remedy: Increase reference speed in C00011.
Bit 9	SpeedNormConstant : Up-Limit	1 ≡ Due to the set machine parameters, an internal calculation constant is limited to a possible maximum. Remedy: Reduce reference speed in C00011.
Bit 10	Max. speed limit	1 ≡ Due to the set machine parameters, the maximum speed for specifications in [units/s] are limited to the factor 150 %.
Bit 11	Quick stop is active	1 ≡ Quick stop function is active.
Bit 12	Quick stop standstill is active	1 ≡ Standstill during active quick stop function (no ramp generation).
Bit 13	Max. speed is pending	1 ≡ The speed setpoint has reached or exceeded the limit of ± 199.99 %.
Bit 14	Position resolution limit	1 ≡ The position resolution calculated by machine data entries has been limited to the maximum display area of 214748.3647 [Incr./Unit] with reference to the display in C01205 . The real value is higher!
Bit 15	Modulo buffer overflow	1 ≡ The internal position memory for creating the saw tooth could not be emptied in one cycle. The speed provides a position difference per cycle that is larger than the cycle length.
Bit 16	Ck10: position > cycle length	Positioning mode: • The position selected for the positioning profile with process data interface is higher than the cycle length when the Modulo measuring system is activated. Homing mode: • The position selected for the sequence profile with process data interface is higher than the cycle length when the Modulo measuring system is activated.
Bit 17	Ck10: invalid speed	Cause 1: • When an infinite profile starts at a current speed = 0, the parameterised profile speed is specified with 0. Cause 2: • When a positioning profile is started for approaching a target position, the parameterised profile speed is specified with 0.

Bit	Designation	Description
Bit 18	Ck10: invalid acceleration	When a profile is started, the parameterised deceleration is specified with 0.
Bit 19	Ck10: invalid deceleration	When a profile is started, the parameterised deceleration is specified with 0. While a positioning profile is executed to target position (current speed ≠ 0), the profile deceleration is written to 0, e.g. by Acc/Dec override.
Bit 20	Ck10: invalid final speed	If C02868/1 Bit 2 = FALSE, it is detected while crossing the finish line of a positioning process that the final profile speed is higher than the current speed. The final profile speed is not reached.
Bit 21	Ck10: reversal at overchange	If C02868/1 Bit 2 = FALSE, the reversing phase is requested during a positioning process to the target position with final profile speed ≠ 0.
Bit 22	Ck10: distance calculation error	During the profile start of a positioning profile, the profile data results in a longer distance than the defined distance (optimised by C02868/1 bit 6).
Bit 23 .... Bit 31	Reserved	

9.2.4 MCK state machine

Prio		Condition				
Global conditions:						
1	15	Setpoint generation through Motor control (MCTRL): • DCB = DC-injection braking • QSP = quick stop • CINH = controller inhibit	DCB, QSP, CINH → ≥1 → 15			
2	4	"Safe stop 1" (SS1) requested Interface to safety system	DCB, QSP, CINH, SafeStop1 → & → 4			
Conditions requested by MCK control word:			**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
3	2	Manual jog requested	0	0	1	0
4	1	Homing requested	0	0	0	1
5	3	Positioning requested	0	0	1	1
6	0	Speed follower requested	0	0	0	0
7	5	Position follower requested	0	1	0	1
8	4	Stop requested	0	1	0	0



[9-4] MCK state machine

9.2.4.1 "StandBy" operating mode

The internal "StandBy" operating mode is implicitly assumed if controller inhibit, pulse inhibit, quick stop and/or DC-injection braking are activated. Thus, there is no setpoint generation by the **Motion Control Kernel** in the "StandBy" operating mode.

- If the holding brake control sets a controller inhibit when the holding brake is closed, the internal "StandBy" operating mode is not assumed.
- The "StandBy" operating mode cannot be activated via the MCK control word.
- When the "StandBy" operating mode is active, bits 0 ... 3 are set in the MCK status word.

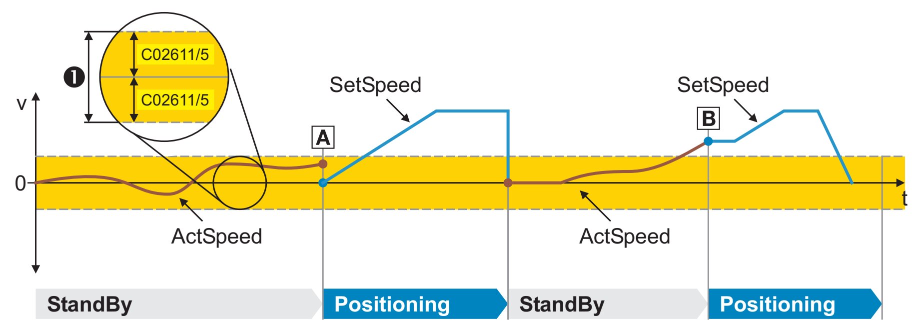
Speed acceptance when the "StandBy" operating mode is quit

In addition to cancelling the controller inhibit, pulse inhibit, quick stop and/or DC injection braking, a change to another operating mode from "StandBy" also requires a termination of the magnetisation of the motor (not in case of synchronous motors). If there is a change to a setpoint-controlled operating mode of the **Motion Control Kernel** the current speed is transferred to the speed setpoint.

Note!

Due to a motor magnetisation of a motor normally standing still, the current speed can also be non-zero rpm when "StandBy" is quit. If the operating mode now changes to "Positioning", the transfer of this speed to the speed setpoint may cause a permanent drifting of the motor shaft, depending on the activated option in C01216.

From version V12.00.00 onwards, a sensitivity threshold can be set for preventing a drifting of the motor shaft in C2611/5. If the absolute value of the current speed is below the sensitivity threshold, the value "0" is transferred to the speed setpoint instead of the current speed. The following illustration shows the issue:



A Current speed below sensitivity threshold → speed setpoint = 0
B Current speed above sensitivity threshold → speed setpoint = current speed
❶ C02611/5: Sensitivity threshold (in [%] with regard to reference speed)

[9-5] Example: Sensitivity threshold for speed transfer from the "StandBy" operating mode

In the Lenze setting "0.5 %", the sensitivity threshold corresponds to approx. 7.5 rpm at a reference speed of 1500 rpm set in C00011.

9.2.5 Interface to safety system

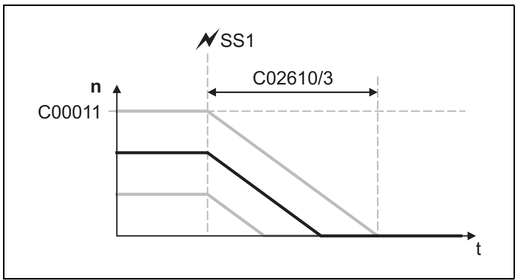
For operation with optional safety system, the LS_MotionControlKernel system block has the *wSMCtrl* input, This interface is used to transfer a control word by means of which the **Motion Control Kernel** can be supplied with information on requested or active safety functions. The **Motion Control Kernel** then initiates the necessary motion sequence (e.g. braking).

At the moment, only bit 0 in the *wSMCtrl* control word has a function. Additional functions are in preparation:

Bit	Designation	Description
0	SafeStop1	"1" ≡ Request for "Safe Stop 1" (SS1).
1 ... 15	Reserved	In preparation - Still without function!

Behaviour in case of request for "Safe Stop 1" (SS1)

The drive is brought to a standstill with the stopping ramp set in C02610/3.



[9-6] Ramping down to standstill

- The time set in C02610/3 refers to the down-ramping of the reference speed set in C00011.
- If the current speed is lower, the time to standstill is accordingly lower as well.

If the request is reset during the down-ramping process (bit 0 = "1↘0"), the behaviour depends on the active operating mode:

- In the "Speed follower" operating mode, direct synchronisation with the target speed takes place with the ramp time set in C02610/2.
- In the "Homing" operating mode, the ramp-down is continued with the deceleration for stop set in C01251 if the bit 8 ("HomStartStop") in the MCK control word = "0". If bit 8 is set, the homing process starts immediately in the selected homing mode.
- In the "Manual jog" operating mode, the behaviour depends on whether the manual jog initiator (*bManJogPos* or *bManJogNeg*) is still set:

Bit 0 (SafeStop1)	bManJogPos bManJogNeg	Behaviour
1	TRUE	Ramping down to standstill
0	TRUE	Accelerating to manual speed
0	FALSE	Ramping down to standstill

- In the "Positioning" operating mode, the behaviour depends on the setting in C01216.
- In the "Position follower" operating mode, (forward) positioning to the signalled absolute position always takes place if this position differs from the internal position.

9.2.6 Consideration of residual value in case of external profile calculation

The *dnProfilePosition_p* input serves to transfer a profile position in [increments] to the SB LS_MotionControlKernel. This will then be entered into the active profile data set selected in the MCK control word via the bits 24 ... 31.

For (external) calculations to be carried out before, the following blocks can be used in the FB Editor.

- From version 12.00.00 onwards, these blocks automatically consider residual values and modulo positions in case of a cycle length definition and correctly forward these internally:

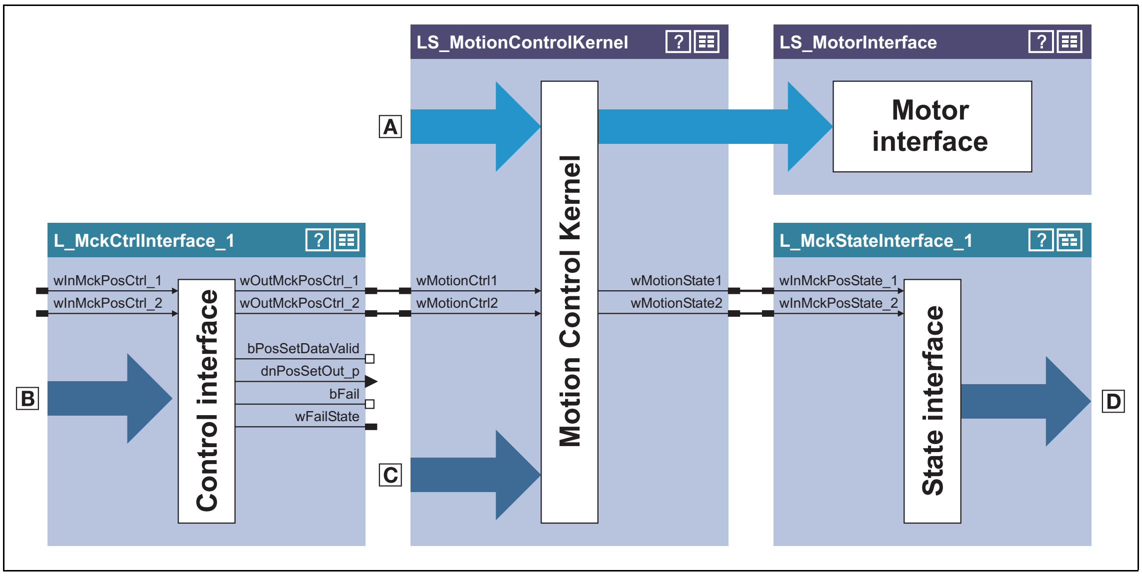
Block	Function
L_ConvUnitsToIncr_1 L_ConvUnitsToIncr_2 L_ConvUnitsToIncr_3	... converts a position value provided in the real unit of the machine into an internal 32-bit position value. • These FBs are available from version 12.00.00.
L_MckCtrlInterface_1	... provides the application with process inputs for controlling various basic functions of the Motion Control Kernel.
LS_ParFreeUnit LS_ParFreeUnit_2	Output of 16 parameterisable position signals with internal conversion of [unit] in [increments] • These SBs are available from version 02.00.00.

The following blocks also automatically consider residual values and module positions and can be used for forwarding the increment results of the previously mentioned blocks:

Block	Function
L_Mux_1	... switches one of eight selectable input signals to the output.
L_SignalSwitch32_1 L_SignalSwitch32_2 L_SignalSwitch32_3	... .switches between two input signals of "DINT" data type. • These FBs are available from version 02.00.00.
LA_SwitchPos / LA_TabPos: • dnFreeIn1_p • dnFreeIn2_p	These two inputs serve to transfer 32-bit signals from the I/O level to the application level. The signals are available at the correspondent outputs of the application block LA_SwitchPosIn or LA_TabPosIn.
LA_TabPos: • dnPosProfilePosition	Selection of the target position in [increments] for the "table positioning" application. The signal is available in the application level at the correspondent output of the LA_TabPosIn application block.

9.3 MCKInterface

The so-called "MCK interface" described in this chapter consists of the two function blocks L_MckCtrlInterface and L_MckStateInterface, which are connected upstream respectively downstream of the LS_MotionControlKernel system block:



[9-7] Detail of the interconnection architecture for the "table positioning" technology application

A Control and setpoint signals for motor control

B Additional process inputs for controlling the **Motion Control Kernel**, e.g.:
- Selection of the operating mode
- Selection of the profile number
- Override of the positioning mode
- Control inputs for manual jogging, homing, positioning
- Control inputs for speed/acceleration/S-ramp override

C Control and setpoint signals for the **Motion Control Kernel** such as
- Selection of speed setpoint for speed follower
- Selection of positioning setpoint for position follower
- Selection of override values
- Connection for limit switch & pre-stop mark for homing
- Holding brake control

D Output of status signals of the **Motion Control Kernel**

Control of the Motion Control Kernel

Control of the basic drive functions implemented in the Motion Control Kernel is carried out by means of

- direct specification of the control words, for example via a master control unit also connected to the fieldbus.
  - For this purpose, the control word inputs can be directly connected to the field bus interface **LP_MciIn** respectively **LP_CanIn**.
  - See the "MCK control word" chapter for a detailed description of the individual control bits.
- the specification of individual process signals at the L_MckCtrlInterface FB which are then ORed with the control word.

Plausibility check

The process signals applied to the L_MckCtrlInterface FB are logically linked by means of an OR logic operation to the specified control word via the two control word inputs *wInMckPosCtrl_1* and *wInMckPosCtrl_2* and, after a plausibility check, are output via the two control word outputs *wOutMckPosCtrl_1* and *wOutMckPosCtrl_2* .

- The profile number, the operating mode, and the positioning mode are checked for plausibility.
- If implausibility is detected, only control bit information is output and the *bFail* output is set to TRUE.
- The result of the plausibility check is provided as the *wFailState* output word and displayed in C01299.

The control words that are output, namely *wOutMckPosCtrl_1* and *wOutMckPosCtrl_2*, constitute the input information for the LS_MotionControlKernel system block.

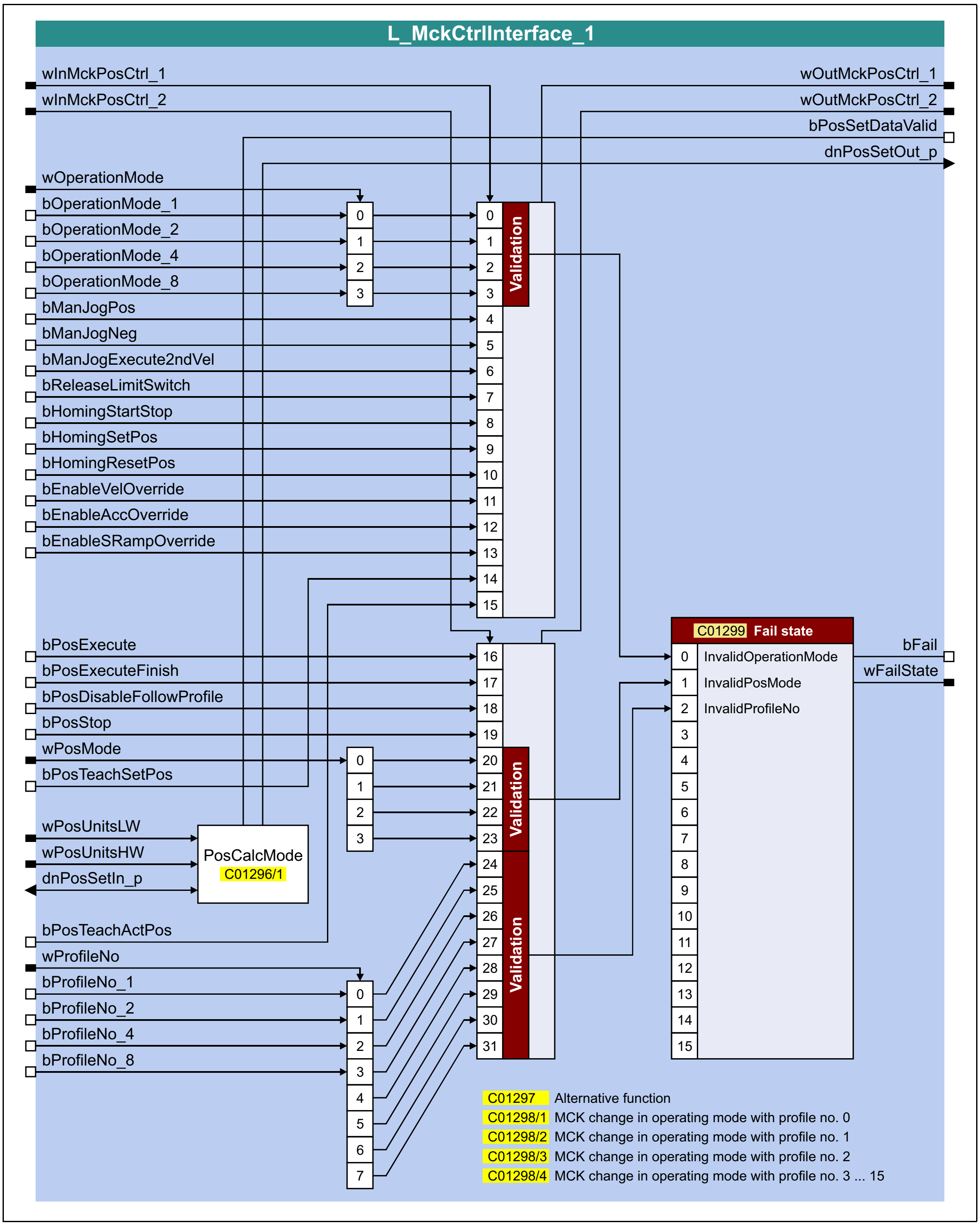
Status information of the Motion Control Kernel

Output by the LS_MotionControlKernel, the status words *wMotionState1* and *wMotionState2* also constitute input information for the downstream L_MckStateInterface FB, which provides this information to the application in the form of process signals.

9.3.1 Control inputs | "L_MckCtrlInterface" function block

This FB provides process inputs for controlling different basic functions of the **Motion Control Kernel**.

In addition to the ORing of discrete input signals with the control word inputs, the FB has further functions that are described in the following subchapters.



inputs

Designator Data type	Information/possible settings	
wInMckPosCtrl_1 wInMckPosCtrl_2 WORD	Direct specification of the MCK control word 1 & 2 • For example, via a master control unit that is also connected to the fieldbus. For this purpose, the control word inputs can be directly connected to the **LP_MciIn** respectively **LP_CanIn1** fieldbus interface. • The two control words together form a 32-bit double control word with which the entire Motion Control Kernel is controlled. • All motion profiles in the different operating modes can be operated via this interface. • See the "MCK control word" chapter for a detailed description of the individual control bits.	
wOperationMode WORD	Selection of the operating mode of the **Motion Control Kernel** • Only bit 0 ... bit 3 of *wOperationMode* are evaluated. • If an invalid operating mode is selected, the response set in C00595/11 is activated (Lenze setting: "Warning"). • The current operating mode is displayed in C01243.	
	0	Speed follower
	1	Homing
	2	Manual jog
	3	Positioning
	4	Stop
	5	Position follower
	6 ... 15	Reserved for future extensions
bOperationMode_1 ... bOperationMode_8 BOOL	Binary-coded selection of the operating mode of the **Motion Control Kernel** • See the "MCK control word" chapter for a detailed description of the individual control bits. • If an invalid operating mode is selected, the response set in C00595/11 is activated (Lenze setting: "Warning"). • The current operating mode is displayed in C01243.	
bManJogPos bManJogNeg BOOL	Manual jog: *bManJogPos* = TRUE: Manual jog right *bManJogNeg* = TRUE: Manual jog left Both inputs = TRUE: No change compared to previous state Both inputs = FALSE: Stop manual jog	
bManJogExecute2ndVel BOOL	Manual jog: Changeover to speed 2	
	FALSE	Speed 1 (C01231/1) active
	TRUE	Speed 2 (C01231/2) active
bReleaseLimitSwitch BOOL	Manual jog: Retract operated limit switch	
	TRUE	Retract operated limit switch (in opposite direction)

Designator (Data type)		Information/possible settings
bHomingStartStop (BOOL)		[Homing](#): Start/stop homing • Only possible in the "referencing" operating mode.
	TRUE	If one of homing modes "4" ... "15" in [C01221](#) is selected: **Start reference search** • The current status of the reference search is indicated via the status outputs *bHomingDone* and *bHomePosAvailable*.
		If homing mode "100: SetRef" is selected: **Setting the home position manually** • The home position is set manually with the drive at a standstill. The current actual position now corresponds to the reference position set in [C01227/2](#) in the machine measuring system.
	TRUE↘FALSE	Stop homing. • If the *bHomingStartStop* input is reset to FALSE during active referencing, homing is cancelled and the drive is brought to a standstill.
bHomingSetPos (BOOL)		[Homing](#): Set home position ("on the fly" homing) • With referencing "on the fly", the home position of a machine can be set during ongoing movement. Jerking and compensating movements do not occur.
	FALSE↗TRUE	The position at the *dnPosRefValue_p* input of the [LS_MotionControlKernel](#) SB at the instant of activation is the set home position.
bHomingResetPos (BOOL)		[Homing](#): Delete home position **Note:** With this function, positions are not deleted but only the status signals *bHomePosAvailable* and *bHomingDone* are reset. Setpoints and actual positions remain untouched until a renewed reference setting or homing.
	FALSE↗TRUE	The internal status "reference known" is reset. • The inverter is no longer referenced. • The process outputs *bHomePosAvailable* and *bHomingDone* are reset to FALSE.
bEnableVelOverride (BOOL)		[Speed override](#)
	TRUE	Activate speed override
bEnableAccOverride (BOOL)		[Acceleration override](#)
	TRUE	Activate acceleration override
bEnableSRampOverride (BOOL)		[S-ramp smoothing override](#) **Note!** If the *nSRampOverride_a* input on the LS MotionControlKernel remains unconnected or if "0 %" is specified as the override value, activation of the S-ramp override results in deactivation of the S-ramp time. • Deactivation of the S-ramp time before the start of a profile with S-ramp time causes linear ramp generation. • Deactivation of the S-ramp time during a traversing process, however, is not accepted immediately in the profile generator, but the profile generator checks automatically when an online change of the ramp form can be carried out and then initiates it automatically.
	TRUE	Activate S-ramp smoothing override
bPosExecute (BOOL)		[Positioning](#): Start travelling
	FALSE↗TRUE	Execute selected profile
bPosExecuteFinish (BOOL)		[Positioning](#): Complete cancelled profile
	FALSE↗TRUE	A positioning process previously cancelled, e.g. by *bPosStop* or due to a device error, is resumed by travelling to the original target.
bPosDisableFollowProfile (BOOL)		[Positioning](#): Do not execute sequence profile (switch-off profile linkage)
	TRUE	Evaluation of the sequence profile number parameterised in [C01307/1...15](#) for the selected profile is suppressed.

Designator Data type	Information/possible settings	
bPosStop BOOL	Positioning: Cancel travelling	
	TRUE	Stop positioning From version 02.00.00, more travel requests will be inhibited ("PosExecute" will be blocked).
wPosMode WORD	Override of the positioning mode set in the profile data • Via this input, an override of the positioning mode parameterised in C01300/1...15 for the selected profile is possible. • The value set in C01300/1...15 is not overwritten in this case. • Only bit 0 ... bit 3 of *wPosMode* are evaluated. • If *wPosMode* = 0 is selected, the positioning mode set in C01300/1...15 is used.	
	0	Positioning mode = setting in C01300/1...15
	1	Absolute (shortest path)
	2	Continuous
	3	Relative
	4	absolute (Cw)
	5	absolute (Ccw)
	8	Absolute (shortest path) to TP
	9	Continuous to TP
	10	Relative to TP
	11	Absolute (Cw) on TP
	12	Absolute (Ccw) on TP
	All other possible settings are reserved for future extensions!	
bPosTeachSetPos BOOL	Position teaching: MCK setpoint position	
	FALSE↗TRUE	Teach MCK setpoint position into the selected profile.
wPosUnitsLW wPosUnitsHW WORD	Selection of the target position in [units] • *wPosUnitsLW* = LOW word, *wPosUnitsHW* = HIGH word • The mode for calculating the position is selected in C01296/1.	
dnPosSetIn_p DINT	Selection of the target position in [increments] • The mode for calculating the position is selected in C01296/1.	
bPosTeachActPos BOOL	Position teaching: Current position	
	FALSE↗TRUE	Teach current position into the selected profile.

wProfileNo
WORD
 	
bProfileNo_1
...
bProfileNo_8
BOOL
	Stipulation of the profile to be executed
	• Optionally as a data word or binary coded.
	• When the profile is stipulated, this FB carries out a mode change in the Lenze setting at the same time:
	  • If profile 0 is selected: Activation of "Speed follower" operating mode
	  • If profile 1 is selected: Activation of "Homing" operating mode
	  • If profile 2 is selected: Activation of "Manual jog" operating mode
	  • If profile 3 ... 15 is selected: Activation of "Positioning" operating mode	

outputs

Designator Data type	Value/meaning	
wOutMckPosCtrl_1 wOutMckPosCtrl_2 WORD	Output of the MCK control word 1 & 2 • For transfer to the LS_MotionControlKernel system block. • For a detailed description of the individual control bits, see chapter "MCK control word". (📖 586)	
bPosSetDataValid BOOL	Status signal "Position conversion completed, data consistent" • This output is permanently set to TRUE if the g"0: dnPosOut_p=dnPosIn_p" mode has been set for the position calculation in C01296/1 and no modulo measuring system exists. • If a cycle length has been set in C01201/1, a calculation is made under the following conditions without an immediate setting of *bPosSetDataValid*: - Position specification is higher than or equal to the cycle length. - Position specification is negative. For a correct positioning process, the *bPosSetDataValid* = TRUE status is required.	
	TRUE	Conversion of the target position from [units] into [increments] has been completed. • The travel profile data are valid and the profile is ready to start.
dnPosSetOut_p DINT	Output of the target position in [increments] • Observe the *bPosSetDataValid* status output!	
wFailState WORD	Result of the plausibility check • Display parameter: C01299 • Result is bit coded:	
	Bit 0	Invalid operating mode selection • "1" ≡ The selected operating mode is not defined/invalid.
	Bit 1	Invalid positioning mode selection • "1" ≡ The selected positioning mode is not defined/invalid.
	Bit 2	Invalid profile number selection • "1" ≡ The selected profile number refers to a profile data set that does not exist.
	Bit 3	Reserved
	...	
	Bit 15	
bFail BOOL	FALSE	Okay, no error
	TRUE	• Plausibility check error or • control information error (in this case after ORing the individual signals with the control words)

9.3.1.1 Alternative functions for control bit "PosExecute"

In C01297, alternative functions for bit 16 (PosExecute) in the MCK control word can be selected with bit-coding.

PosStop with PosExecute = FALSE

If bit 0 has been set in C01297, positioning can only be started/cancelled with the "PosExecute" control bit.

- If the "Positioning" operating mode has been set on the MCK, the "Pos-Execute" control bit has the following effects when being activated:

Signals at the input	Signals in the control word to the MCK
bPosExecute = TRUE	bPosExecute = TRUE bPosStop = FALSE bHomingStartStop remains unchanged
bPosExecute = FALSE	bPosExecute = FALSE bPosStop = TRUE bHomingStartStop remains unchanged

HomingStartStop with PosExecute

If bit 1 has been set in C01297, homing can only be started/stopped with the "PosExecute" control bit.

- If the "Homing" operating mode has been set on the MCK, the "PosExecute" control bit has the following effect when activated:

Signals at the input	Signals in the control word to the MCK
bPosExecute = TRUE	bPosExecute = TRUE bPosStop = FALSE bHomingStartStop = TRUE
bPosExecute = FALSE	bPosExecute = FALSE bPosStop remains unchanged bHomingStartStop = FALSE

SetProfilPosition with PosExecute

If bit 2 has been set in C01297 and control bit ("PosExecute") is being set, the setpoint position being applied is incorporated in the currently selected profile and then the profile is started immediately.

- If the "Positioning" operating mode has been set on the MCK, the "PosExecute" control bit has the following effect when activated:

Signals at the input	Signals in the control word to the MCK
bPosExecute = TRUE	bPosExecute = TRUE bPosStop = FALSE bHomingStartStop remains unchanged bPosTeachSetPos = TRUE (edge)
bPosExecute = FALSE	bPosExecute = FALSE bPosStop remains unchanged bHomingStartStop remains unchanged bPosTeachSetPos = FALSE

SetProfilPosition at position change

If bit 3 is set in C01297, the setpoint positions at the MCKInterface are automatically accepted into the profile with the applied profile number if a change of data is detected at the corresponding input for the setpoint position.

- If the "0: dnPosOut_p=dnPosIn_p" mode has been set in C01296/1 for converting the position, automatic acceptance is executed if the incremental position selection at the *dnPosIn_p* input was changed.
- If another mode (>0) has been set in C01296/1, automatic acceptance is executed if the incremental position selection at the *wPosUnitsLW* and *wPosUnitsHW* inputs was changed.
- From version 02.00.00 a hysteresis can be set in C01245/3 for the position change.

PosExecute at position change

(from version 02.00.00)

If bit 4 is set in C01297, an automatic "PosExecute" takes place if the incremental position selection changes and this change is higher than set in the hysteresis for position change (C01245/3).

If the selection is made in units, so that in a first step it has to be converted into increments, and if this automatic function is then activated, the "PosExecute" will only be created automatically when the internal conversion is completed (*bPosSetDataValid*).

Tip!

This option serves to start travel requests very easily by simply defining the new target position.

9.3.1.2 Operating mode change with profile number

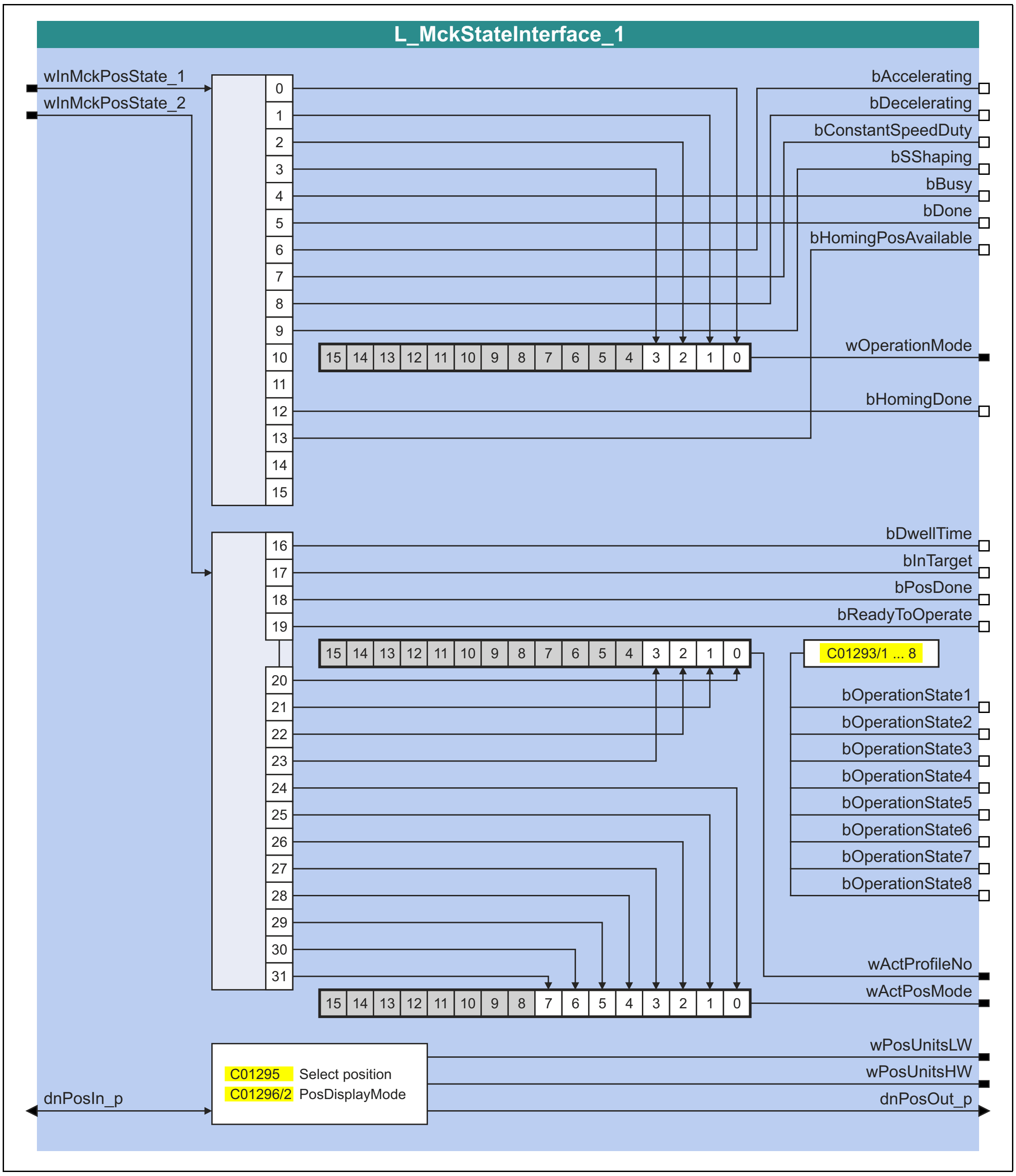
One of the operating modes of the **LS_MotionControlKernel** system block can be assigned to the positioning profiles via the four subcodes of C01298:

Parameters	Info	Lenze setting
C01298/1	Operating mode if profile 0 is selected	Speed follower
C01298/2	Operating mode if profile 1 is selected	homing
C01298/3	Operating mode if profile 2 is selected	Manual jog
C01298/4	Operating mode if profile 3 ... 15 is selected	Positioning

- When a changeover to the corresponding profile is carried out, the set operating module is requested at the same time.
- With the setting "0: Op.Mode inputs", no operating modes are changed when the profile is changed. Instead, the operating mode requested "externally" applies (optionally via the inputs *wInMckPosCtrl_1*, *wOperationMode* or *bOperationMode_1...8* of the FB L_MckCtrlInterface).
  Note: These inputs are internally connected via OR-logic. Only connect one of these inputs to ensure non-ambiguous operating mode changeovers!

9.3.2 Status outputs | FB "L_MckStateInterface"

This FB provides the application with different status information of the **Motion Control Kernel** via process outputs.



Note!

When C01296/2 > 0, the position selected in C01295 is calculated in [units]. In this case, the output values *wPosUnitsLW*, *wPosUnitsHW* and *dnPosOut_p* are not updated in a 1-ms cycle (not in real time). However, they are applied consistently to each other. For this reason, we recommend the use of these outputs only for diagnostic purposes where an update in real time is not important.

inputs

Designator / Data type	Information/possible settings
wInMckPosState_1 wInMckPosState_2 WORD	Inputs for accepting the status words from the LS_MotionControlKernel system block.
dnPosIn_p DINT	Position in [increments] • Is converted into [units] if C01295 = "0: dnPosIn_p" and output at the *wPosUnitsLW* and *wPosUnitsHW* outputs. • The mode for calculating the position is selected in C01296/2.

outputs

Designator / Data type	Value/meaning	
bAccelerating BOOL	TRUE	Acceleration phase active.
bDecelerating BOOL	TRUE	Braking phase active.
bConstantSpeedDuty BOOL	TRUE	Constant phase active.
bSShaping BOOL	From version 12.00.00:	
	TRUE	Profile rounding active.
bBusy BOOL	TRUE	Setpoint generation active.
bDone BOOL	TRUE	Target position (setpoint) has been approached.
bHomingPosAvailable BOOL	TRUE	Home position is known.
wOperationMode WORD	Active setpoint-generating state of the **Motion Control Kernel**. • Bit B0 ... B3 contain the information of the MCK status word. • Bits B4 ... B15 are fixed at "0".	
bHomingDone BOOL	TRUE	Homing has been executed.
bDwellTime BOOL	TRUE	Settling in target position is active
bInTarget BOOL	TRUE	Target position (actual value) is in the target window.
bPosDone BOOL	TRUE	Target position from the profile has been approached.
bReadyToOperate BOOL	TRUE	Readiness to process setpoint signals and setpoint commands.
bOperationState1 ... bOperationState8 BOOL	TRUE	Configurable status display from the MCK status word 1 ▸ Configurable status display bOperationState
wActProfileNo WORD	Current traversing profile number • Bit B0 ... B7 contain the information of the MCK status word. • Bits B8 ... B15 are fixed at "0".	
wActPosMode WORD	Current positioning mode • Bit B0 ... B3 contain the information of the MCK status word. • Bits B4 ... B15 are fixed at "0".	

Designator Data type	Value/meaning
wPosUnitsLW wPosUnitsHW WORD	Output of the position selected in C01295 in [units] • *wPosUnitsLW* = LOW word, *wPosUnitsHW* = HIGH word • The mode for calculating the position is selected in C01296/2. • These values are not updated in a 1-ms cycle (not in real time)!
dnPosOut_p DINT	Output of the position selected in C01295 in [increments] • The mode for calculating the position is selected in C01296/2. • When C01296/2 > 0, *dnPosOut_p* is updated isochronously with the outputs *wPosUnitsLW* and *wPosUnitsHW* and hence not in a 1-ms cycle (not in real time)!

9.3.2.1 Configurable status display *bOperationState*

This function extension is available from version 16.00.00!

The status display from the MCK status word 1 can be configured with C01293/1 ... C01293/8 and applies to the following states:

Status bit	Status
Bit00	ActOpModeBit00
Bit01	ActOpModeBit01
Bit02	ActOpModeBit02
Bit03	ActOpModeBit03
Bit04	Busy
Bit05	Done
Bit06	HomPosAvailable
Bit07	ReadyToOperate

Example

The *bOperationState5* output is to provide the TRUE status if the drive

- is in the current **positioning** mode
- has been referenced
- and is ready for start signals.

The code C01293/5 has to be parameterised as follows:

Status bit	Status
Bit00	ActOpModeBit00 = TRUE
Bit01	ActOpModeBit01 = TRUE
Bit02	ActOpModeBit02 = FALSE
Bit03	ActOpModeBit03 = FALSE
Bit04	Busy = FALSE
Bit05	Done = FALSE
Bit06	HomPosAvailable = TRUE
Bit07	ReadyToOperate = TRUE

9.4 Basic settings

9.4.1 Machine parameters

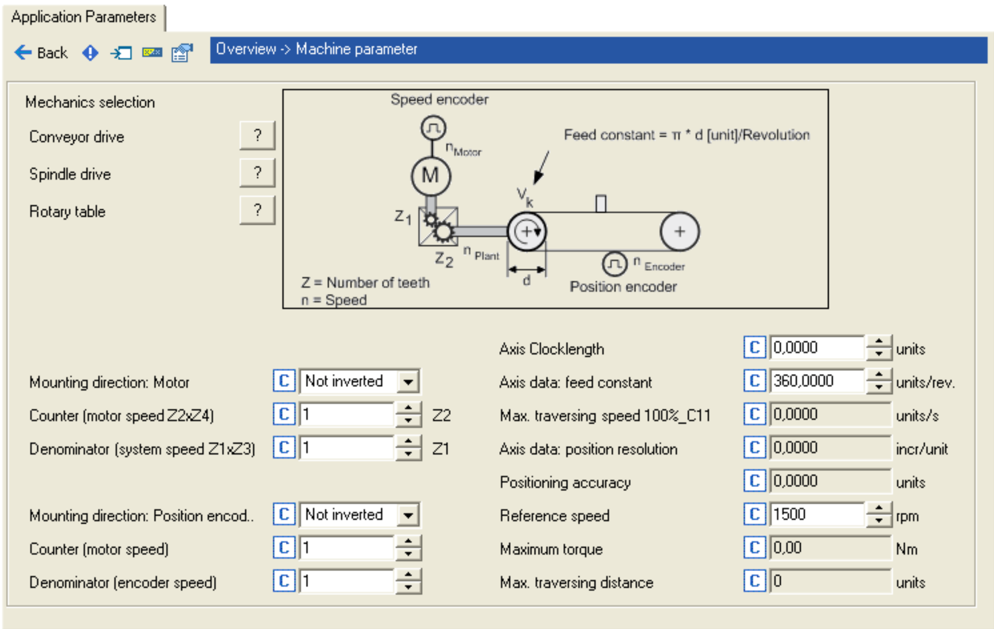
The motor end, among other things, is described by the machine parameters indicated below in respect of the mechanics used.

Note!

Setting the machine parameters is a basic prerequisite for the operating modes "Homing", "Manual jog" and "Positioning".

The more precisely the machine parameters are set, the better the results of positioning!

For TA "Table positioning", you can set the machine parameters in the »Engineer» on the tab headed **Application Parameters** on the dialog level *Overview* ➞ *Machine parameters*:



Tip!

You are provided with more detailed information on the machine parameters "Gearbox ratio" and "Feed constant" in the following subchapters.

Short overview of machine parameters:

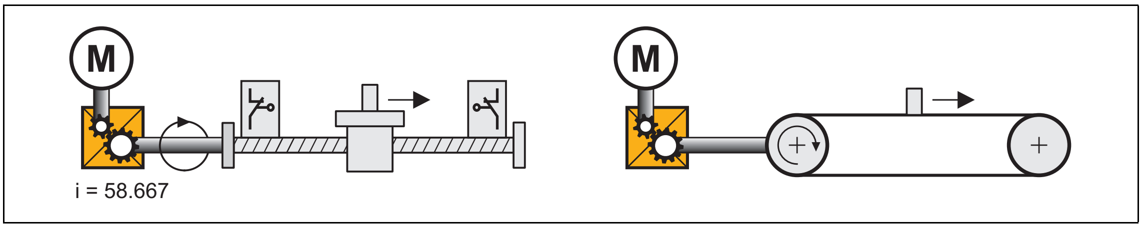
Parameters	Info	Lenze setting Value	Unit
C01206/1	Mounting direction: Motor	0: Not inverted	
C01202/1	iM: Nominator of gearbox factor Z2	1	
C01202/2	iM: Denominator of gearbox factor Z1	1	
C01206/2	Mounting direction: Position encoder	0: Not inverted	
C01203/1	iG: Numerator (motor speed)	1	
C01203/2	iG: Denominator (encoder speed)	1	
C01201/1	Axis data: Axis cycle ▶ Activation of the modulo measuring system	0.0000	units
C01204	Axis data: Feed constant	360.0000	units/rev.
C00011	Appl.: Reference speed	1500	rpm

After the machine parameters have been entered, the inverter sends back application-relevant data via the following display parameters:

Parameters	Info	Lenze setting Value	Unit
C01211/1	Max. traversing speed 100%_C11 • Is used for orientation when the velocity is being set in the profile sets.	-	units/s
C01205	Axis data: Position resolution • Is needed for the incremental specification of positions.	-	incr/unit
C01210/5	MCK: Positioning accuracy • Theoretical accuracy during positioning with account being taken of the machine data and encoder data.	-	units
C00057	Maximum torque	-	Nm
C01213/1	MCK: Max. traversing distance	-	units
Greyed out = display parameter			

9.4.1.1 Gearbox ratio

The gearbox ratio indicates the number of revolutions of the motor axis it takes for exactly one revolution of the load axis (e.g. spindle or drive roll) to take place.

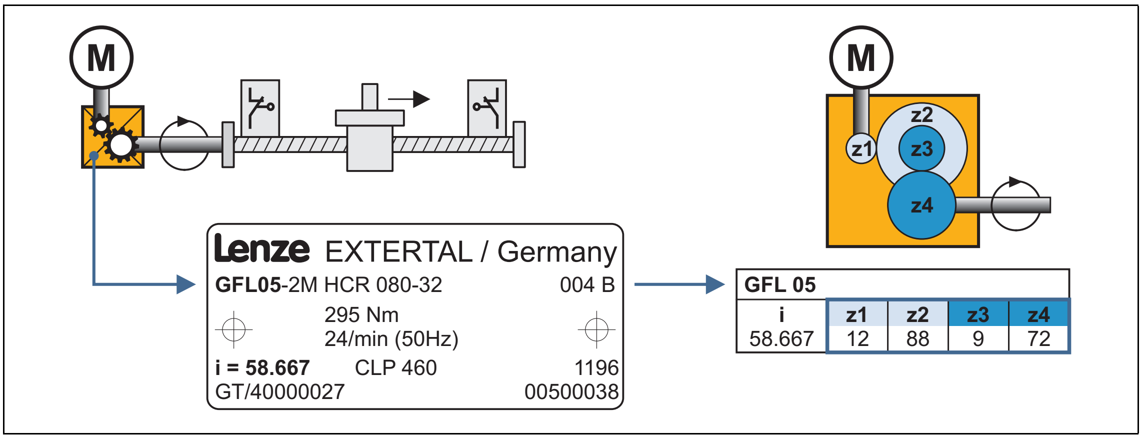


[9-8] Schematic diagram of gearbox ratio

- In the example shown in illustration [9-8] one revolution of the spindle is carried out at exactly 58,667 revolutions of the motor axis.

Specification of the gearbox ratio

- The gearbox ratio is to be defined in the form of a quotient (numerator/denominator); the data required can be found in the technical data for the gearbox:



[9-9] Example: Technical data relating to the gearbox (from gearbox catalogue)

Tip!

In order to specify the gearbox ratio exactly, use the number of teeth indicated on the data sheet or in the catalogue, if possible, instead of the information on the nameplate (see following calculation).

Example calculation on the basis of the technical gearbox data:

$$\text{Gearbox factor numerator (C01202/1)} = z2 \times z4 = 88 \times 72 = 6336$$

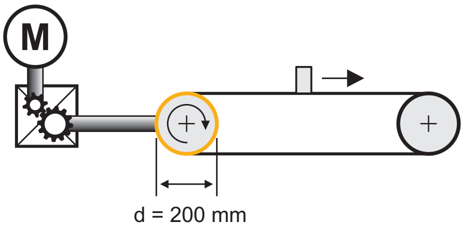
$$\text{Gearbox factor denominator (C01202/2)} = z1 \times z3 = 12 \times 9 = 108$$

[9-10] Calculation example (for 2-stage gearbox)

9.4.1.2 Feed constant

The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft.

- The entry in the **Feed constant** (C01204) field is made in the user unit [units] in respect of the revolution.
- In the case of a conveyor drive, the feed constant is obtained from the drive roll's circumference, which, in the following example, is calculated on the basis of the indicated diameter:

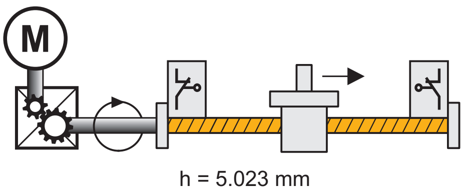


d = diameter

$$\text{Feed constant} = \pi \cdot d \frac{[\text{Unit}]}{\text{Revolution}} = \pi \cdot 200 \frac{\text{mm}}{\text{Revolution}} = 628.3185 \frac{\text{mm}}{\text{Revolution}}$$

[9-11] Schematic diagram: Feed constant for a conveyor driver

- In the case of a spindle drive (linear axis), the feed constant is derived from the leadscrew pitch. The feed constant indicates the distance the slide travels during one revolution of the spindle (in the following example: 5.023 mm).



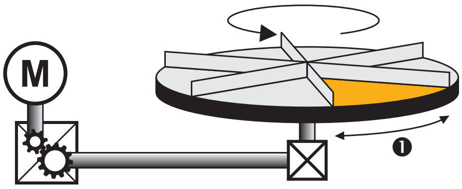
h = leadscrew pitch (can be obtained from the technical data of the linear axis)

[9-12] Schematic diagram: Feed constant for a spindle drive

- In the case of a rotary table and its specification as an angle, the feed constant is = 360°/revolution.

9.4.1.3 Activation of the modulo measuring system

The modulo measuring system is also called "rotary table application".



❶ C01201/1: Cycle

[9-13] Example: Rotary table application

- The measuring system is repeated.
- If the set cycle is exceeded, a defined overflow takes place.
  - In a rotary system, the cycle length typically corresponds to one rotation or one tool distance.
- For positioning, the home position must be known.
  - Exception: Positioning modes relative (TP) and continuous (TP)
- Software limit positions are not effective.
- Absolute targets can be approached by exceeding the measuring system limit, e.g. from 10° via 0° to 350°.

Activating the modulo measuring system

The Modulo system is activated by setting a cycle (C01201/1) > 0 units.

- The setting of the cycle is possible when the inverter is enabled.
- When the cycle (C01201/1) is set to 0 units (Lenze setting), the traversing range is unlimited (classical measuring system).

Creation of the modulo measuring system

When the Modulo measuring system is active, it is displayed internally via an integrator. The Modulo position is provided at the *dnPosSet_p* process output of the SB LS_MotionControlKernel and displayed in C01210/7. When the Modulo measuring system is not active, the continuous (*dnPosSetValue_p*) setpoint position is output instead.

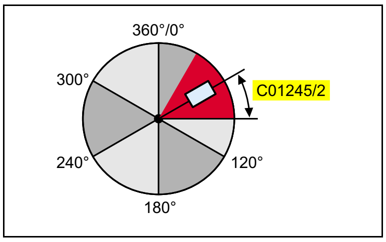
Blocking zone for "absolute (Cw)" and "absolute (Ccw)" positioning modes

Set a blocking zone in C01245/2 for the "absolute (Cw)" and "absolute (Ccw)" positioning modes in which no target positions are approached. This blocking zone around the current setpoint position serves to consider drift motions of the shaft in case of controller inhibit/enable cycles in order to prevent e.g. an unwanted motion by one cycle.

Stop!

With the "absolute (Cw)" and "absolute (Ccw)" positioning modes, target positions that are located in the set blocking zone are always approached by the shortest possible path!

If the blocking zone is set too large, the drive may travel in the opposite direction!



[9-14] Definition of the blocking zone

- Target positions outside the blocking zone are approached with the selected positioning mode.
- The blocking zone is internally limited to half the cycle.

Display of the target position in the "positioning" mode

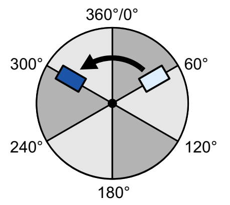
Positioning mode	Target position displayed in C01210/6 in [units]	
Absolute (shortest path)	Defined position	
absolute (Cw)	Defined position	Multiple cycles are possible (target position ≥ cycle length). The blocking zone in C01245/2 is taken into consideration if C02868/1: bit 5 (optimal procedure for multiple cycles) is set.
absolute (Ccw)	Defined position	
Continuous	214748.3647	
Relative	Modulo position + defined position	

Case studies for modulo positioning

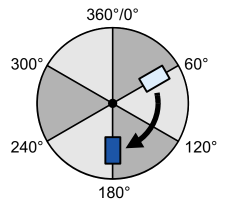
In the following, some case studies with the different positioning modes are represented. The target positions are defined by the user.

Positioning mode "absolute (shortest path)"

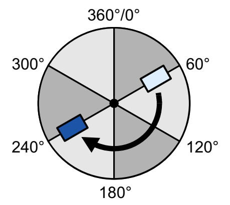
Case 1



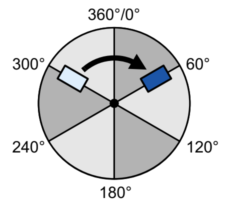
Case 2



Case 3

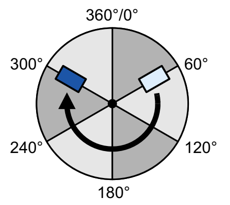


Case 4

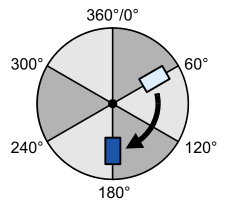


Positioning mode "absolute (Cw)"

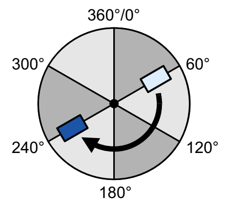
Case 1



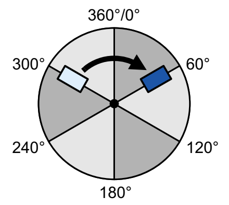
Case 2



Case 3

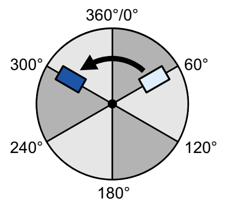


Case 4

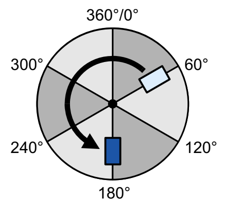


Positioning mode "absolute (Ccw)"

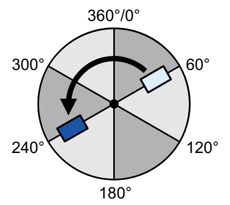
Case 1



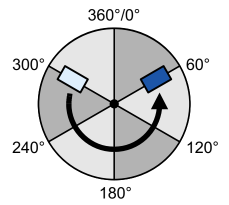
Case 2



Case 3



Case 4



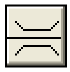
9.4.2 Min/Max speed

For the "Speed follower" operating mode, yo can initially limit the speed setpoint by means of the following parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C02610/2	MCK: Ramp time synchr. setpoint	2.000	s
C02611/1	MCK: Pos. max. speed	199.99	%
C02611/2	MCK: Pos. min. speed	0.00	%
C02611/3	MCK: Neg. min. speed	0.00	%
C02611/4	MCK: Neg. max. speed	199.99	%

In the »Engineer», you can set the initial limit by means of the dialog box *Min/Max speed*.

- Open the *Min/max speed* dialog box by opening the **Application Parameters** tab and clicking on the following button on the dialog level *Overview* ➞ *Signal flow*:

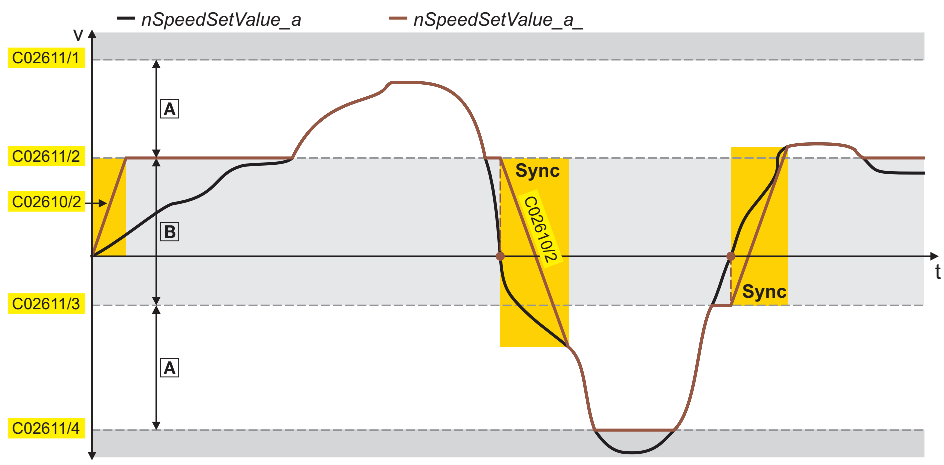


Speed setpoint generation

When the speed limit values are set, the **Motion Control Kernel** influences the setpoint generation with a synchronisation mode The synchronisation mode serves to travel the synchronisation range dynamically with the synchronisation ramp set in C02610/2.

For the synchronisation process it is irrelevant whether you are already in the "Speed follower" operating mode or whether you branch to this operating mode. If you branch to the "Speed follower" operating mode, the current speed setpoint is accepted and processed with regard to the input limitation parameterised. The transition to the "Speed follower" operating mode may cause a speed jump if the synchronisation period is set too short. This speed jump is avoided by extending the ramp time with C02610/2.

Synchronisation starts in the zero crossing of the selected speed in each case:



A Setpoint range
B Synchronisation range

[9-15] Example: Speed setpoint generation in the "Speed follower" operating mode (with nSpeedAddValue_v = 0)

9.4.3 Limit position monitoring

9.4.3.1 Software limit positions

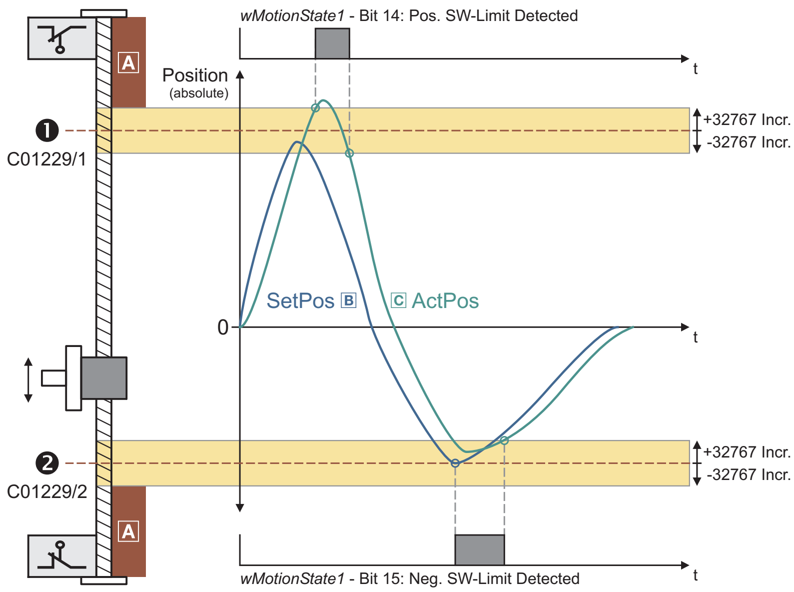
The parameterisable software limit positions serve to limit the traversing range by the software.

The positive software limit position is set in C01229/1 and the negative software limit position is set in C01229/2.

Evaluation and monitoring of the software limit positions are only carried out if

- C01229/1 - C01229/2 > 32767
- no modulo measuring system is set (C01201/1 = 0.0000 units)
- the drive knows the home position (*bHomePosAvailable* = TRUE) and the software limit positions for the respective operating mode have been activated (see the following table)!

Operating mode	Software limit positions active (if home position is known)
Speed follower	Yes (adjustable in C01219 - bit 3)
Homing	Yes
Manual jog	No (adjustable in C01230 - bit 3)
Positioning	Yes
Position follower	Yes (adjustable in C01218 - bit 3)



A Error range (actual position) B Set position C Actual position
❶ C01229/1: Positive software limit position
❷ C01229/2: Negative software limit position

[9-16] Definition of the software limit position trigger limits

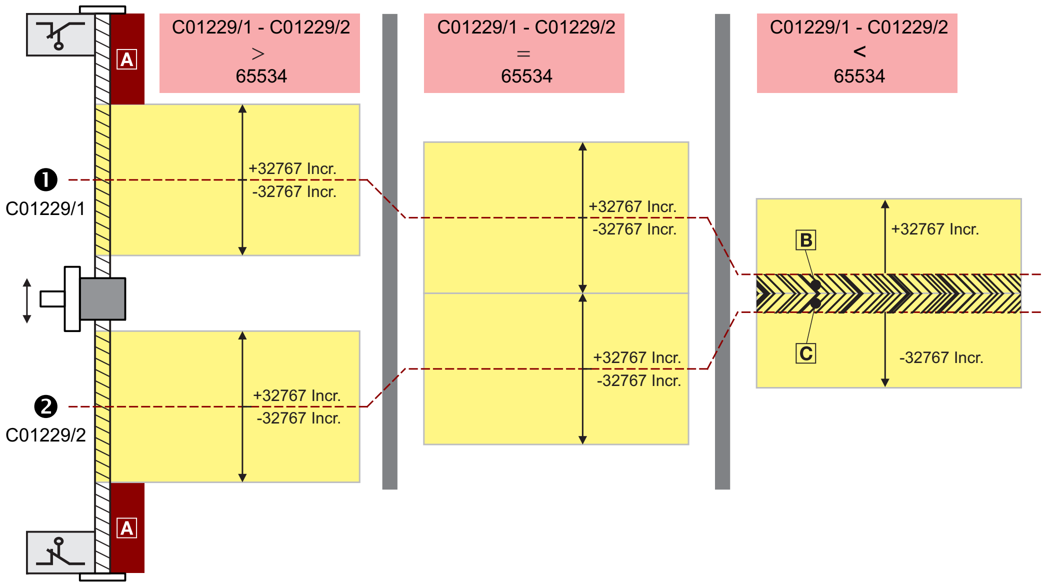
Exception: hysteresis of both software limit positions meets or overlaps

In the description of the previous section, the hysteresis areas of both software limit positions are separated from each other. The following applies to this standard case (also see the left-hand diagram below):

- C01229/1 - C01229/2 > 65534 incr

The following applies to applications in which the hysteresis areas meet or overlap (also see the diagram in the middle below/right-hand diagram below):

- C01229/1 - C01229/2 ≤ 65534 incr



A Error area (actual position)
B Reduced hysteresis area in positive direction
C Reduced hysteresis area in negative direction
❶ C01229/1: Positive software limit position
❷ C01229/2: Negative software limit position

Up to version 15.xx.xx, the maximum travel range will be set if hysteresis areas are overlapping (C01229/1 - C01229/2 ≤ 65534 incr.).

Danger!

The drive traverses without taking the software limit positions set into consideration!

The software limit positions are therefore deactivated. In order to prevent the system from being damaged, the software limit positions have to be corrected so that the following conditions apply:

1. SW_Limit_POS[incr] > SW_Limit_NEG[incr]
2. SW_Limit_POS[incr] - SW_Limit_NEG[incr] > 32767

From version 16.00.00 onwards, overlapping of the inner hysteresis areas of the software limit positions is avoided by scaling down the hysteresis width, see hatched area:

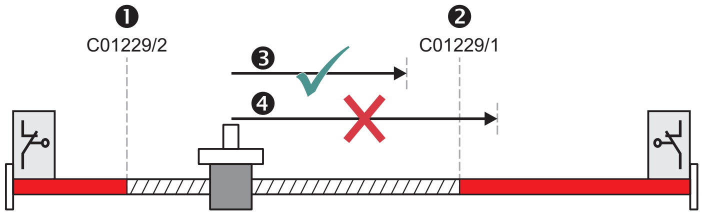
Hysteresis width [incr] in pos. direction B	Hysteresis width in neg. direction C
$\text{C01229/1} - \left(\frac{\text{C01229/1} - \text{C01229/2}}{2}\right)$	$\text{C01229/2} + \left(\frac{\text{C01229/1} - \text{C01229/2}}{2}\right)$

Behaviour in the case of active software limit positions

Note!

The"travel commands" mentioned in the following description are no speed setpoint selections. In the operating modes "Speed follower" and "Position follower", an acknowledged software limit position error ensures that traversing into the impermissible travel range remains possible afterwards. This is because in these two operating modes, there is no preview whether a software limit position is approached or not with a setpoint selection.

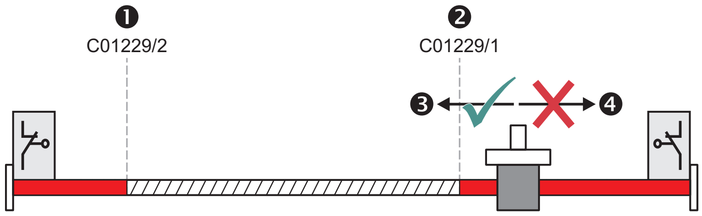
If the software limit positions are active, travelling commands that would result in exiting from the permissible travel range can no longer be executed:



❶ C01229/2: Negative software limit position
❷ C01229/1: Positive software limit position
❸ Travel command possible
❹ Travel command not possible since the target position is outside the permissible traversing range.

[9-17] Example: Traversing range limitation by means of software limit positions

If the drive is already outside the permissible travel range and the software limit positions have been activated, only travel commands that result in the drive moving back into the permissible travel range can be executed:



❶ C01229/2: Negative software limit position
❷ C01229/1: Positive software limit position
❸ Traversing in "retracting direction" possible
❹ Traversing opposite to "retracting direction" not possible

[9-18] Example: Permissible traversing direction if software limit positions active

If the software limit positions are active and a software limit position is passed ("overtravel"):

- The error response "TroubleQuickStop" takes place in the Lenze setting, i.e. the drive is brought to a standstill in the deceleration time set for the quick stop function and does so irrespective of the setpoint selection. The error response can be parameterised in C00595/3 and C00595/4.
- The error message "Ck03: Pos. SW limit position" or "Ck04: Neg. SW limit position" is entered in the logbook of the inverter.
- Depending on the parameterised fault response, the drive cannot traverse until the error has been acknowledged.
- C01247/Bit4 and C01247/Bit5 show if a software limit position is active.

Homing in case of software limit positions

From version 16.00.00 onwards:

Software limit position monitoring responds (error message "Ck04: neg. SW limit position") if homing takes place in the limited measuring system at reasonably parameterised and effective software limit positions and the hardware limit switch is used as reference signal. The sequence profile for positioning in the "homing" mode in the valid software limit position range is not started.

- By changing over to the "manual jog" mode and error acknowledgement, the drive can be travelled back into the valid software limit position range.

From version 18.00.00:

If bit 8 is set in C01220 ("SW limits inactive after Ref Ok"), the software limit positions are directly deactivated after the reference signal is detected. Thus, an optionally set sequence profile can start and moves the drive into the valid software limit position range.

9.4.3.2 Hardware limit switches

The travel range limits are monitored by means of limit switches via the inputs *bLimitSwitchPos* and *bLimitSwitchNeg* of the LS_MotionControlKernel system block.

- The two inputs react to the TRUE state.
- In TA "Table positioning", the two inputs are connected to the digital inputs provided for connection of the limit switches.

Stop!

The limit switches are only evaluated if the limit switches for the respective operating mode have been activated (see the following table)!

Operating mode	Hardware limit switch effective
Speed follower	Yes (adjustable in C01219 - bit 2)
Homing	Depending on the selected homing mode (see description of the homing modes)
Manual jog	No (adjustable in C01230 - bit 2)
Positioning	Yes
Stop	Yes
Position follower	Yes (adjustable in C01218 - bit 2)

Note!

If the digital inputs used for the connection of the limit switches are to be designed in a fail-safe manner (activation at LOW level), you simply change the terminal polarity of the corresponding digital inputs in C00114.

Behaviour when hardware limit switches are active

If one of the two monitoring inputs is set to TRUE:

- The error response "TroubleQuickStop" takes place in the Lenze setting, i.e. the drive is brought to a standstill in the deceleration time set for the quick stop function and does so irrespective of the setpoint selection. The error response can be parameterised in C00595/1 and C00595/2.
- The error message "Ck01: Pos. HW limit switch" or. "Ck02: Neg. HW limit switch" is entered in the logbook of the inverter.
- Bit 10 ("Pos. HW-Limit Detected") or bit 11 ("Neg. HW-Limit Detected") is set in the MCK status word.
- Depending on the parameterised fault response, the drive cannot traverse until the error has been acknowledged.

Reading out code C01247/0, bit 2 /bit 3 serves to check whether the travel range limit switches are active in positive or negative direction.

Note!

An activated limit switch can be retracted again by manual jog in the opposed direction or with the "Retract limit switch" function. ▶ Retracting of an operated limit switch (📖 665)

Only in the "Manual jog" operating mode, retracting of the limit switch resets bit 10 ("Pos. HW-Limit Detected") or bit 11 ("Neg. HW-Limit Detected") in the MCK status word.

Re-activation after acknowledging the error

When the error has been acknowledged and the limit switch is still active, the following action is required for a renewed activation of the monitoring depending on the operating mode:

Operating mode	Action for (re) activation
Speed follower	Limit switch is activated and setpoint in direction of the activated limit switch is pending **Note:** If the setpoint e.g. is specified via an analog potentiometer, potentiometer noise can mean an unintentional setpoint causing the error to be set again.
Homing	Limit switch is activated and *bHomStartStop* = TRUE
Manual jog	Limit switch is activated and manual jog in direction of the activated limit switch
Positioning	Limit switch is activated and setpoint command has been transmitted
Stop	Setting of the operating mode
Position follower	Limit switch is activated and setpoint in direction of the activated limit switch is pending

9.4.4 Target position monitoring (status "drive in target")

The target position monitoring detects whether the drive has reached the target.

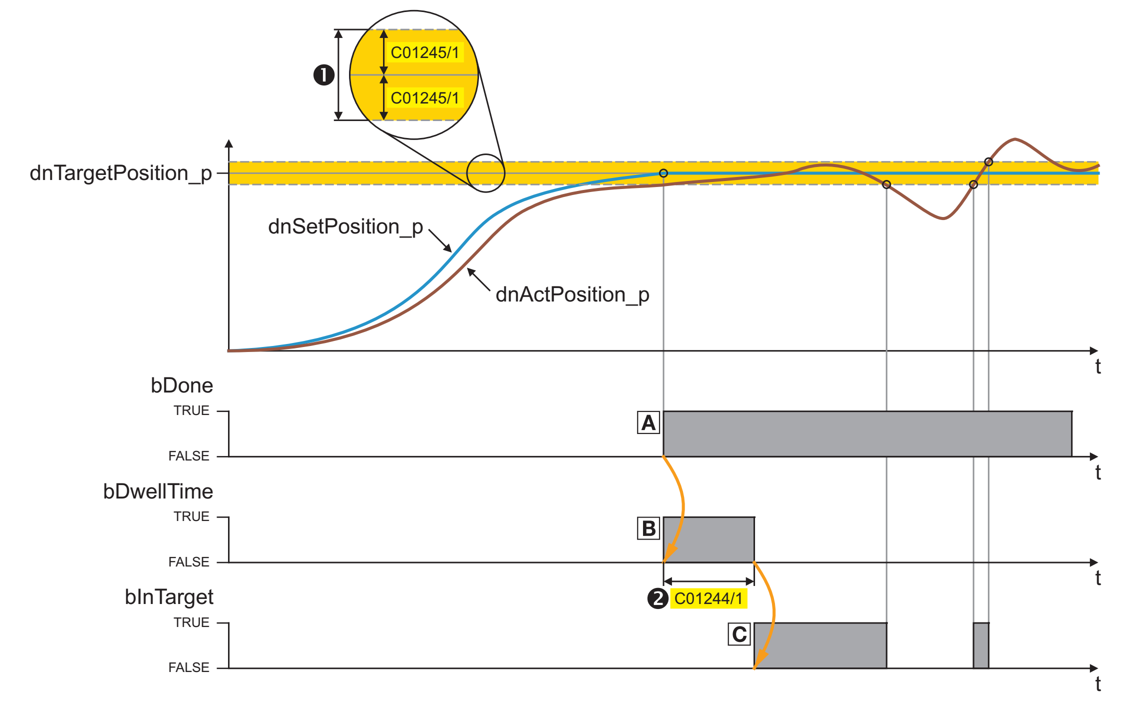
Note!

- The target position detection is active with the following operating modes/functions:
  - Operating mode "Positioning": Absolute and relative positioning mode
  - Operating mode "Homing": After approaching the starting position via start profile (if start profile has been selected)
  - "Manual jog" operating mode: when approaching the breakpoints and the software limit positions, respectively
- The target position monitoring is active until the operating mode changes or a new travel command influences the setpoint.

Short overview of the parameters for target position monitoring:

Parameters	Info	Lenze setting	
		Value	Unit
C01244/1	MCK: Dwell time - target position • For considering mechanical transient phenomenons of the tool when the target position is reached.	100	ms
C01245/1	MCK: Window target position • Symmetrical window around the target position.	1.0000	units

Principal procedure of the position detection



❶ target position window (symmetrically around the target position; 2 x C01245/1)
❷ dwell time

[9-19] Signal characteristics

Status "Done" (A)

- The position setpoint has reached the target position.

Status "DwellTime" (B)

- Bit 16 in the MCK status word or the *bDwellTime* output of the L_MckStateInterface FB is set when the setpoint position has been reached after the dwell time set in C01244/1 has expired.
- The status detects the time after the *bDone* signal when settling to target position is executed and actual position detection is not active yet.

Status "InTarget" (C)

- If the actual position is within the symmetrical target position window after the dwell time has expired, bit 17 is set to "1" in the MCK status word or the *bInTarget* output of the L_MckStateInterface FB is set to TRUE.

Note!

- The target position including the target position window has to be within the maximum traversing range.
- Please observe the following if you use *bInTarget* as stepping condition in a step sequence:
  The *bInTarget* signal needs the set position to be located in the target. A profile restart immediately deletes the setpoint position in the first cycle and thus also resets *bInTarget*.

- The "InTarget" status is reset if:
  - the motor shaft leaves the target position window,
  - a new profile or a new motion process is started,
  - *bInTarget* has been set via manual jog (intermediate stops) and the "Manual jog" mode is left,
  - the target position window has been left in the "Speed follower" mode and a speed setpoint not equal to "0" is injected into the process,
  - a new homing process starts,
  - the reference is set,
  - the device is switched off/on.

9.4.5 Monitoring of the maximum travel distance

Continuous travel requests in the "Positioning" operating mode (or relative positioning with feed in the same direction) cause an overflow of the position integrators when the max. display area of the position is reached and the reference is set. The same behaviour takes place in the "Speed follower" and "Manual jog" operating modes. For this reason, a monitoring mode of the maximum travel distance is implemented in the **Motion Control Kernel**.

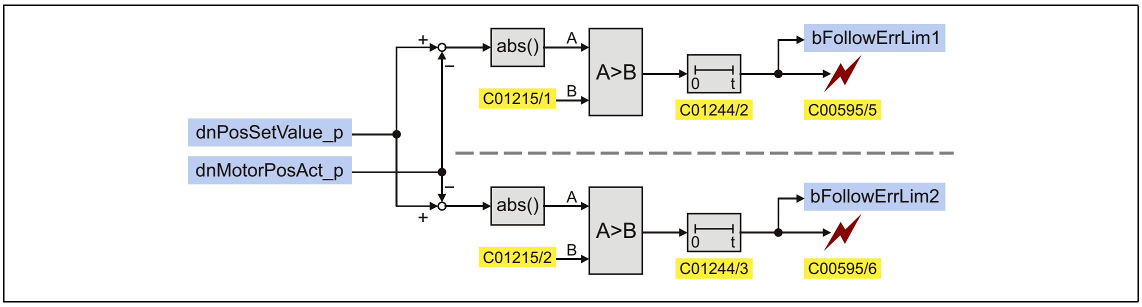
- If the maximum travel distance (C01213/1) is exceeded:
  - The error response set in C00595/7 will be carried out (Lenze setting: "TroubleQuickStop").
  - The "Ck07: Travel range limit exceeded" error message is entered into the logbook.
  - The internal status "reference known" is reset (the inverter is no longer referenced).
- A setting of C00595/7 = "0: No Reaction" deactivates the monitoring.

9.4.6 Following error monitoring system

The difference between set position and actual position is called the following error. Ideally, the following error should be "0". The set position is created by the internal definition of the traversing profiles of the **Motion Control Kernel**. The actual position is created by the integration of the speed supplied by the position encoder. If the position control is adjusted optimally, only a minimum following error arises which is always compensated dynamically and not increases continuously.

Certain processes, however, require that a defined limit as a difference between set position and actual position is not exceeded. If it is exceeded, it may have been caused by mechanical blocking in the machine and the system part is not situated at the position defined at that time. In such a case, it makes sense to activate the "Fault" error response to make the motor torqueless.

In the 8400 TopLine inverter, two independent following error monitoring systems can be parameterised:



[9-20] Two-channel following error monitoring system

Parameters	Info	Lenze setting	
		Value	Unit
C01215/1	MCK: Following error limit 1	0.0000	units
C01215/2	MCK: Following error limit 2	0.0000	units
C01244/2	MCK: Following error deceleration 1	0.000	s
C01244/3	MCK: Following error deceleration 2	0.000	s
C00595/5	MCK: Resp. to following error 1	Warning	
C00595/6	MCK: Resp. to following error 2	Warning	

Note!

If the limit for the following error is set to "0.0000 units" (Lenze setting, the following error monitoring system is not active.

Tip!

In certain situations (e.g. dynamic acceleration of the load), higher system-dependent following errors occur than while approaching the target position.

In order that no error is triggered during acceleration and a close tolerance limit can be monitored all the same at standstill in the target, the addressing of the following error monitoring system can be decelerated. Thus, dynamic processes or torque impulses occurring for short periods can be "masked out".

Operating principle

If the limit for the following error in C01215/1..2 is set higher than "0.0000 units" and if the current following error exceeds this limit over the time set in C01244/x:

- The error response set in C00595/5 or C00595/6 will be carried out (Lenze setting: "Warning").
- The error message "Ck05: Error, following error 1" or "Ck06: Error, following error 2" will be entered into the logbook.
- The status output *bFollowErrLim1* or *bFollowErrLim2* at the SB LS_MotionControlKernel is set to TRUE.

9.4.6.1 Monitoring of the following error in case of controller inhibit

Up to and including version 13.xx.xx, the setpoint position is always set = actual position in case of controller inhibit.

From version 14.00.00 onwards, the setpoint position is only set = actual position if the current following error reaches the limit value set in C01215/3.

- In the Lenze setting C01215/3 = "0.0000 units" the behaviour is as before, i.e. the setpoint position is immediately set = actual position in case of controller inhibit.

Parameters	Info	Lenze setting	
		Value	Unit
C01215/3	MCK: Following error SetPos=ActPos	0.0000	units

Note!

The limit value set in C01215/3 should not be set higher than the drift of the actual position at standstill of the drive plus a possible safety option. Otherwise, the position controller carries out an intermittent compensation after the controller inhibit is deactivated due to the pending system deviation!

Application example

In case ofa continuous relative positioning by 100 units and a setting of the controller inhibit after each positioning process (e.g. by means of the Holding brake control in the "automatically controlled" mode), it is expected that after 10 positioning processes, an absolute target of 10 x 100 units = 1000 units has been reached.

Without setting a limit value (C01215/3 =0), the continuous setting of the setpoint position to the actual position in case of controller inhibit causes a drifting of the setpoint position due to the drifting of the actual position at standstill. After ten positioning processes, only e.g. 999.5 units have been reached instead of 1000 units. If, however, the setpoint position is not set to the actual position (C01215/3 <> 0) in case of controller inhibit, the setpoint position is exactly in the target after 10 positioning processes and 9 x controller inhibits (in case of 10 x relative traverse paths).

9.4.7 Setpoint holdback for bus runtime compensation

This function extension is available from version 12.00.00!

Basics of the "setpoint holdback" function

In case of a master/slave application in which a slave drive must follow a master drive with a precise angle, the data coupling from axis to axis is mostly effected via real-time capable fieldbuses (e.g. CAN).

- Here, the data is always transferred in a time-controlled way. The axes are synchronised, i.e. the reading of the process data and its internal processing are effected at the same time.
- For such applications, the 8400 TopLine inverter provides the Axis bus which is very easy to configure.

During a synchronous run between a master and several slaves, mostly a position and speed setpoint generated by the master are transferred to the slaves.

- The setpoints are generated in the master, usually by the LS_MotionControlKernel system block.
- The setpoints are transferred to the slaves via PDOs.
- The output to the respective fieldbus is effected via the corresponding port blocks (e.g. LP_CanOut1 or LP_MciOut) or in case of the axis bus via the LS_AxisBusOut system block.

Due to the bus cycle time (e.g. 1 ms or 2 ... 4 ms for buses with a higher load) and the telegram runtimes (approx. 260 µs per PDO @500 kbits), the setpoints generated by the master arrive in the slave axes with a delay. At this time, the master has already transferred the setpoints to the control loops of the motor control. This offset causes an error in the synchronism of the axes.

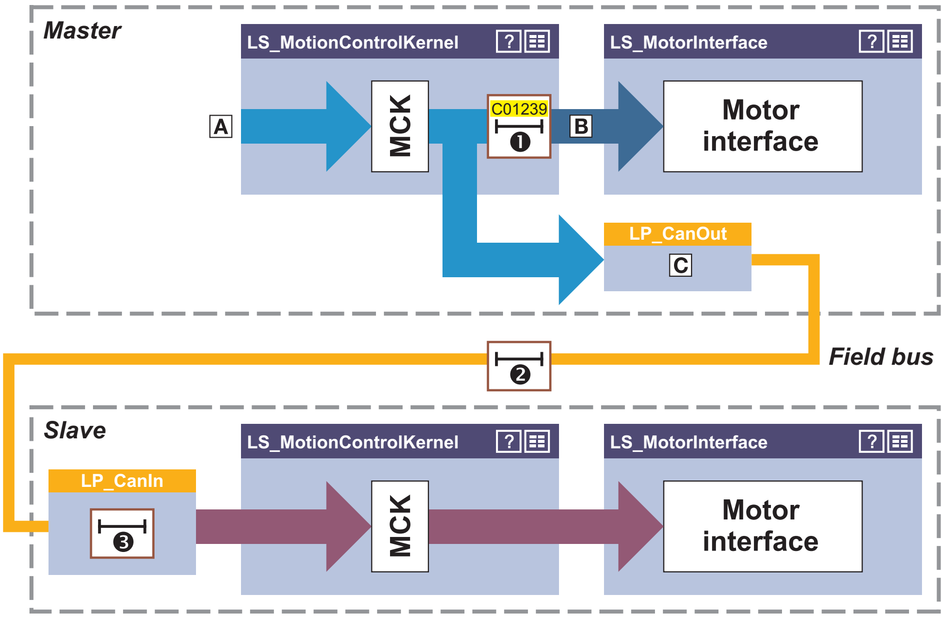
- In case of a mechanical coupling, e.g. a portal feed with two drives, an inclination of the two feed drives is caused.
- This behaviour is more or less distinctive for all serial bus systems depending on the transmission speed and cycle time.

A further delay between the reading of the received data and the forwarding to the control loops is caused by the internal cycle time of the inverters (1 ms for the 8400 device series).

Parameterising the "setpoint holdback" function

For compensating the runtimes described before, a setpoint holdback can be parameterised in C01239. The function directly affects the setpoint forwarding from the **Motion Control Kernel** to the motor control.

- Internally, the setpoints are transmitted with a delay by the time set in C01239 from the LS_MotionControlKernel system block to the motor control.
- Setting rule: C01239 = bus cycle time + 1 ms
- The Lenze setting "0 ms" corresponds to the previous behaviour (no setpoint holdback).



A Setpoints
B Transfer of the setpoints to the motor control
C Transfer of the setpoints to the slave drives via fieldbus
❶ Setpoint holdback: Parameterisable runtime for the transfer of the setpoints to the motor control
❷ Runtime on the bus due to the bus cycle time and telegram runtimes
❸ Runtime in the slave inverter due to the internal cycle time (1 ms) of the inverter

[9-21] Principle of the setpoint holdback (here only in the master for "setpoint line" topology)

Note!

In which drives the function is to be parameterised, depends on the bus topology:

- Setpoint line (1 master, many slaves):
  Only in the master, the setpoint holdback (C01239) has to be set.
- Setpoint cascade (setpoints from one to drive to the other):
  In all drives, the setpoint holdback (C01239) has to be set.

As the setpoint holdback is implemented in the **Motion Control Kernel**, the LS_MotionControlKernel system block is required for this function independent of the setpoint source.

Behaviour when the "setpoint holdback" function is parameterised online

The setpoint holdback can also be set "online", i.e. without inhibiting the inverter before. This provides for an adjustment during commissioning while the setpoint generation is running, similar to the setting of the gain of the speed or position controller.

Stop!

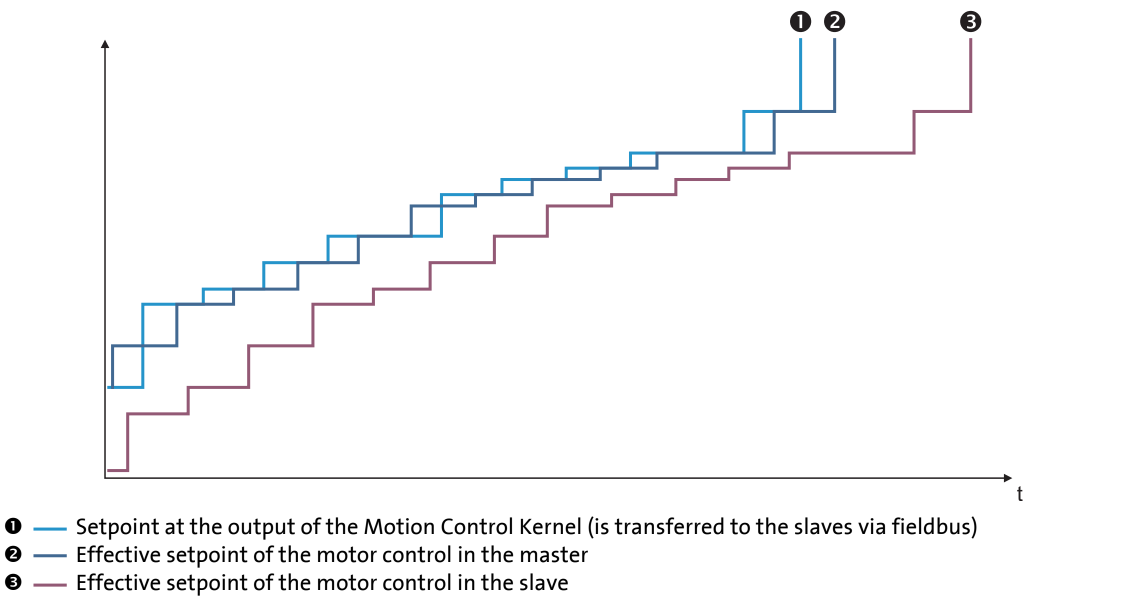
The setting while the setpoint generation is running causes a system-dependent time change of the setpoint processing so that e.g. considerable step-changes or setpoint holding processes occur.

In order to prevent setpoint step-changes which have a negative effect on the mechanics:

- Always change the setting of the setpoint holdback in 1-ms steps only.
- Make the online parameterisation at low speeds only.

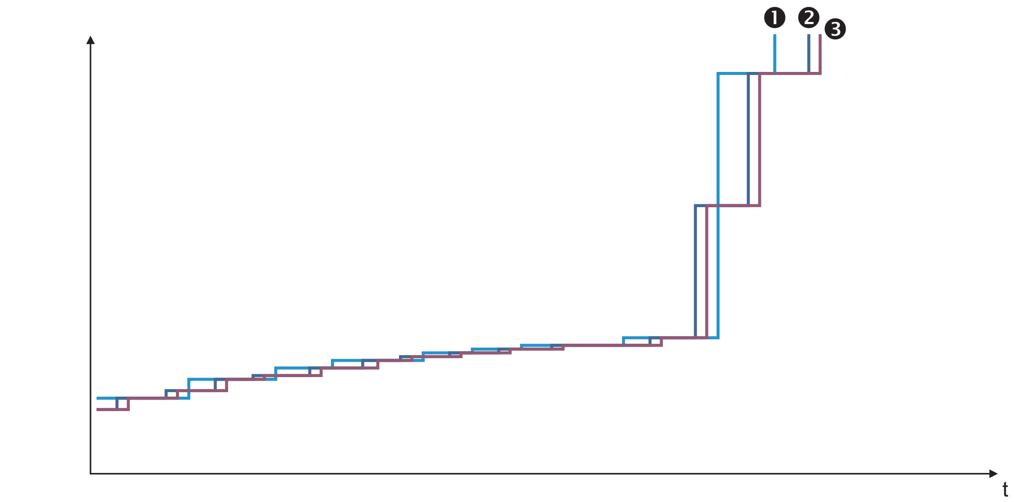
Example: Runtime behaviour without and with setpoint holdback

Example 1 shows the behaviour without setpoint holdback. Due to the runtimes, there is an offset between master and slave of 2 ms.



[9-22] Example 1: Runtime behaviour without setpoint holdback

In example 2, the setpoint holdback in the master is set to 2 ms. This removes the offset between master and slave. Only time differences are visible due to the telegram runtimes on the bus.



❶ Setpoint at the output of the Motion Control Kernel (is transferred to the slaves via fieldbus)
❷ Effective setpoint of the motor control in the master
❸ Effective setpoint of the motor control in the slave

[9-23] Example 2: Runtime behaviour with setpoint holdback = 2 ms

9.4.8 Setting or activation of maximum jerk for traversing profiles

This function extension is available from version 14.00.00!

Basics of the "maximum jerk" function

A change over of a traversing profile within an acceleration or deceleration ramp to a sequence profile which has a lower jerk than the traversing profile from the calculation of the new acceleration and S-ramp time causes an increase of the speed which may be not wanted for process technology reasons. The reason is that the new jerk reduces the acceleration in a too long time period which generates a too high speed.

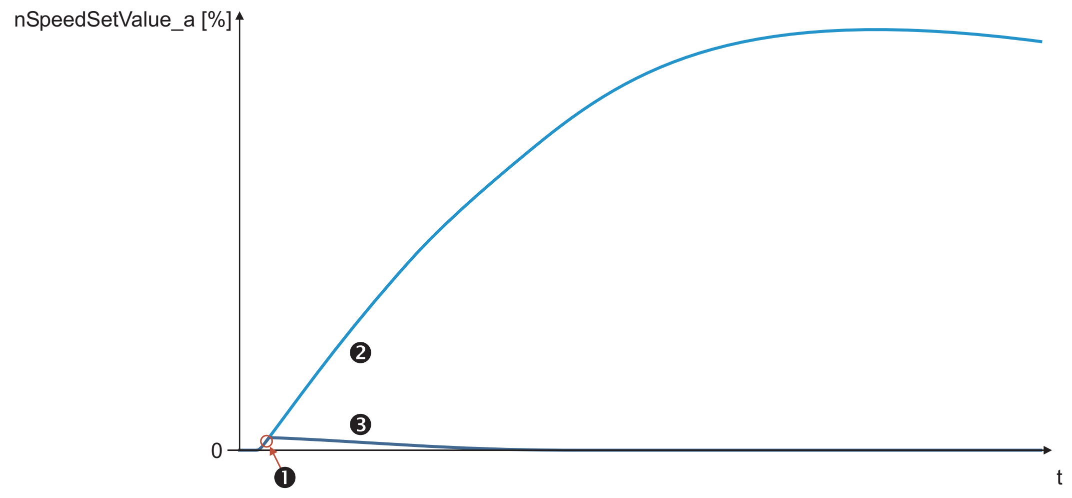
As a remedy, the "maximum jerk" function can be activated which integrates the currently pending acceleration into the acceleration of the new traversing profile or to zero.

Parameterising the "maximum jerk" function

The "maximum jerk" function can be individually activated for the operating modes "Positioning", "Homing" and "Manual jog" via the following setting parameters:

Parameters	Info	Lenze setting
C01216 - bit 5	Maximum jerk for positioning on/off	Off
C01220/1 - bit 5	Maximum jerk for homing on/off	Off
C01230 - bit 5	Maximum jerk for manual jog on/off	Off

When the function is activated, the maximum jerk is calculated from the higher value of both acceleration values and the lower value of both S-ramp time values based on the acceleration and S-ramp time of the current and sequence profile. This maximum jerk is then used for an accelerated drive in order to reduce the acceleration to 0 or to the acceleration of the sequence profile.



❶ Transition to sequence profile
❷ —— Speed characteristic without "maximum jerk" setting
❸ —— Speed characteristic with "maximum jerk" setting

[9-24] Example: Behaviour during transition to sequence profile with accelerated drive

If the sequence profile is started with a non-accelerated drive, the parameters of the sequence profile are continued to be used for the jerk calculation without maximum comparison to the current profile.

9.5 Speed follower

In the "speed follower" operating mode, the drive follows a speed setpoint.

9.5.1 Parameter setting

Short overview of parameters for the "speed follower" operating mode:

Parameters	Info	Lenze setting Value	Unit
C01219	MCK: Speed follower setting	Bit coded	
C02610/2	MCK: Ramp time synchr. setpoint	2.000	s
C02611/1	MCK: Pos. max. speed	199.99	%
C02611/2	MCK: Pos. min. speed	0.00	%
C02611/3	MCK: Neg. min. speed	0.00	%
C02611/4	MCK: Neg. max. speed	199.99	%

9.5.1.1 Functional settings

In C01219, various functional settings for the speed follower can be made in bit-coded form.

Function		Lenze setting
Bit 0	Reserved	Off
Bit 1	Reserved	Off
Bit 2	**HW limit switch on** In the "speed follower" operating mode, a travel range monitoring mode via hardware limit switch is active. ▶ Limit position monitoring (📖 619)	On
Bit 3	**SW limit switch on** In the "speed follower" operating mode, a travel range monitoring mode via parameterised software limit positions. ▶ Limit position monitoring (📖 619)	On
Bit 4	Reserved	Off
Bit 5	Reserved	Off
Bit 6	Reserved	Off
Bit 7	**Position controller off** In the "Speed follower" operating mode, the position controller is deactivated. Thus, the compensation of the following error is switched off.	Off

9.5.2 Requesting the operating mode

Request for "speed follower" operating mode by means of the MCK control word:

MCK control word						
Bit 31	**...**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
			OpMode_Bit3	OpMode_Bit2	OpMode_Bit1	OpMode_Bit0
X	...	X	0	0	0	0
X = Status not significant						

If the **MCKInterface** is connected upstream to the **Motion Control Kernel** and if the operating mode is requested at the L_MckCtrlInterface FB, the *wOperationMode* and *bOperationMode_1...8* process inputs are available.

9.5.3 Setpoint selection

The speed setpoint is selected via the *nSpeedSetValue_a* process input and additively via the *nSpeedAddValue_v* process input.

- Usually, the ramp generator L_NSet and, optionally, the process controller L_PCTRL are upstream of the *nSpeedSetValue_a* process input.
- The speed setpoint is limited internally to the speed limits set in C02611/1...4.

Note!

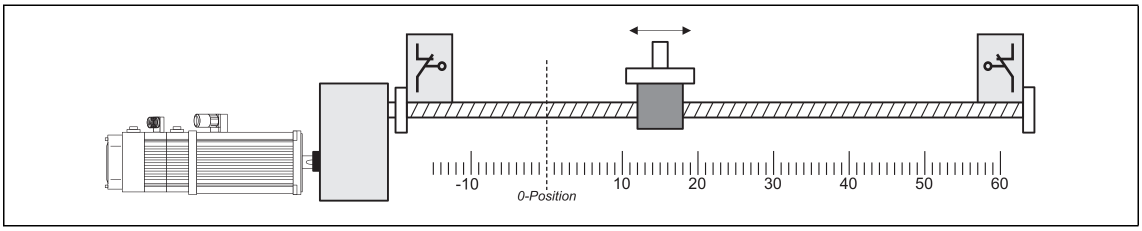
In the "Speed follower operating mode without position control, for an "Electrical shaft" interconnection the actual position value *LS_MotorInterface.dnMotorPosAct_p* must be used as master signal if a drift is to be ruled out.

When the speed limit values are set, the **Motion Control Kernel** influences the setpoint generation with a synchronisation mode The synchronisation mode serves to travel the synchronisation range dynamically with the synchronisation ramp set in C02610/2.

▶ Min/Max speed (📖 618)

9.6 Homing

The measuring system in the machine is selected by means of homing and the 0 position is set within the possible physical travel range:



[9-25] Homing (selection of 0 position)

The zero position (home) can be defined by a reference run or by setting a home position:

- If a reference run is carried out, the drive follows a previously selected path to find the home position.
- When the reference is set, it is selected manually when the drive stands still.

Danger!

During homing operations, specially assigned profile parameters are effective. If these parameters are not set correctly, the drive may execute unexpected movements!

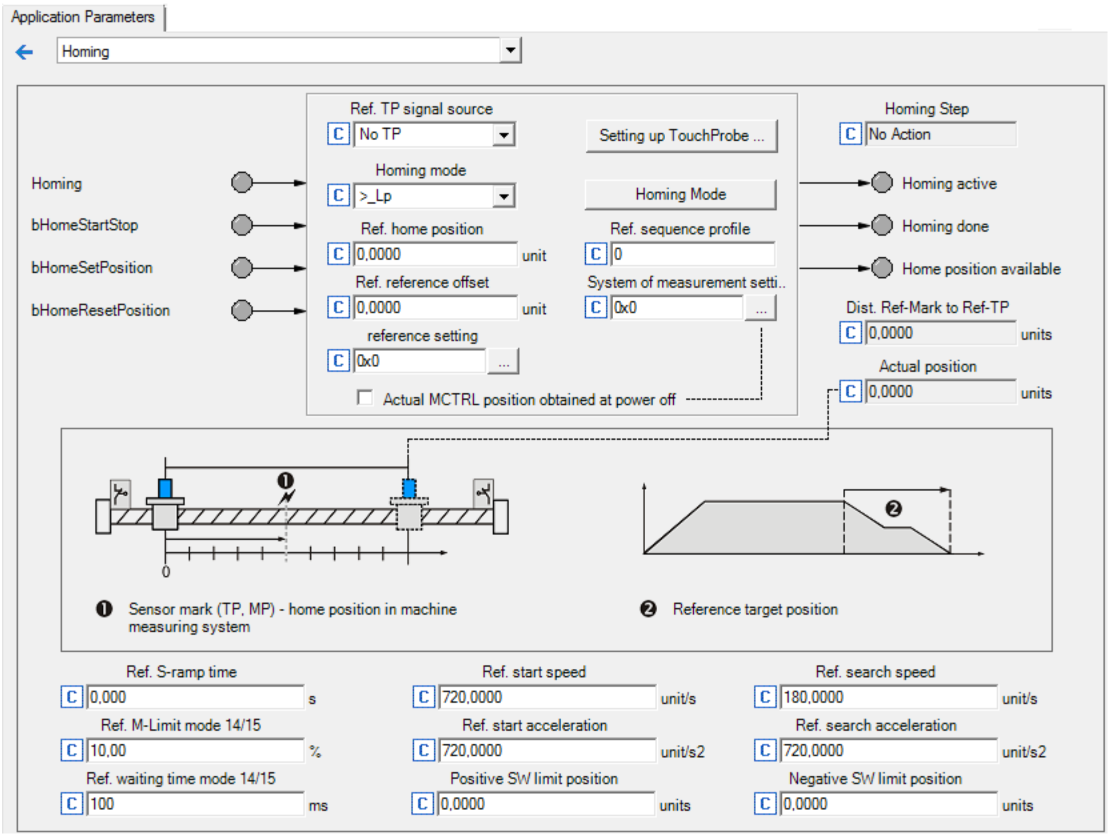
Tip!

A reference run is mainly used for systems with feedback via encoders, resolvers, or single-turn absolute value encoders, as in the case of these systems the home position is lost when the supply voltage is switched off.

Usually the reference is set only once during commissioning or in the event of service (e. g. if drive components are replaced), and it is mostly used for systems with feedback via absolute value encoders.

9.6.1 Parameter setting

Parameterisation dialog in the »Engineer«



Short overview of parameters for "Referencing" operating mode:

Parameters	Info	Lenze setting Value	Unit
C01220/1	MCK: Ref. setting (from version 14.00.00)	Bit coded	
C01221	MCK: Homing mode	12: >_Lp	
C01224/1	MCK: Ref. initial speed	720.0000	unit/s
C01225/1	MCK: Ref. initial acceleration	720.0000	unit/s2
C01224/2	MCK: Ref. search speed	180.0000	unit/s
C01225/2	MCK: Ref. search acceleration	720.0000	unit/s2
C01226/1	MCK: Ref. S-ramp time	0.000	s
C01222	MCK: Ref. M limit mode 14/15	10.00	%
C01223	MCK: Ref. waiting time mode 14/15	100	ms
C01227/1	MCK: Ref. offset reference degree	0.0000	unit
C01227/2	MCK: Ref. home position	0.0000	unit
C01228	MCK: Ref. sequence profile	0	
C01229/1	MCK: Positive SW limit position	0.0000	units
C01229/2	MCK: Negative SW limit position	0.0000	units
C01246/1	MCK: Ref. TP signal source	0: No TP	
C01246/2	MCK: Set.Ref. signal source	0: No TP	
C01248/1	MCK: Reference step	-	units
C01210/9	MCK: Dist. ref. mark and Ref-TP	-	units
C01210/3	MCK: Actual position	-	-

9.6.1.1 Functional settings

From version 14.00.00, various functional settings for homing can be carried out bit-coded in C01220/1.

Function		Lenze setting
Bit 0	Reserved	Off
Bit 1	Flange-sensitive start	Off
Bit 2	Reserved	Off
Bit 3	Reserved	Off
Bit 4	**Start deletes bHomePosAvailable** When homing is started (mode 4 ... 15), the reference information is deleted.	Off
Bit 5	**Maximum jerk** ▶ Setting or activation of maximum jerk for traversing profiles (📖 634)	Off
Bit 6	Reserved	Off
Bit 7	Reserved	Off
Bit 8	**SW limits inactive after Ref Ok** After detecting the reference signal, the software limit positions are deactivated.	Off
Bit 9 ... Bit 15	Reserved	Off

9.6.1.2 Homing mode

Specify the referencing mode in C01221 , i.e. the way in which referencing is to take place.

- For reference setting, the referencing mode "100" is to be selected in C01221.
- For a reference search, C01221 contains referencing modes "4"..."15" which can be selected from.

Referencing mode C01221	**Evaluated signals/sensors** Touch probe sensor (Sensor reference signal)	Travel range limit switch		Pre-stop mark at *bHomingMark*
		Negative limit switch	Positive limit switch	
4 *	☑			☑
5 *	☑			☑
6	☑			☑
7	☑			☑
8	☑			
9	☑			
10	☑		☑	
11	☑	☑		
12			☑	
13		☑		
14	Positive direction of rotation to torque limit.			
15	Negative direction of rotation to torque limit.			
100	Set reference directly.			
* From version 14.00.00				

Internal interfaces

The switches/sensors are evaluated via the following internal interfaces:

Switch/sensor	**Internal interface for digital input signal**
Touch probe sensor (Sensor reference signal)	The touch probe signal source can be selected in C01246/1. • If the reference signal is to follow a real touch probe, configure the touch probe signal accordingly. ▶ Touch probe detection (🕮 435) **Note:** In case of setting "0: No TP" in C01246/1, the digital input DI3 is used as signal source. An inversion of DI3 via C00114 is considered.
Positive travel range limit switch	*bLimitSwitchPos*
Negative travel range limit switch	*bLimitSwitchNeg*
Pre-stop mark/pre-stop signal	*bHomingMark* • This input has to be connected to the corresponding digital input to which the pre-switch off sensor is connected. • The edge sensitivity of this input and the response to the pre-switch off signal depend on the selected homing mode (see the following description of the homing modes).

Note!

For a reference search with touch probe detection:

- The touch probe signal source can be selected in C01246/1.
- Go to C02810/x and select the edge, the digital input used for the connection of the touch-probe sensor is to respond. In the Lenze setting of C02810/x, no touch probe is detected!
- Moreover, no "acceptance window" must be set in C2813/x and C02814/x for accepting the touch probe signal in order that a touch probe will always be detected independent of the position.

▶ Touch probe detection (📖 435)

Start and search profile data set

Certain referencing modes use two different profile data sets for homing in order to shorten the homing time and, at the same time, increase accuracy.

- At first, a quick approach of the limit switch/pre-stop mark (depending on the selected mode) is carried out using the start profile data set.
- After reversing at the limit switch/pre-stop mark, the search profile data set results in slower – but more accurate – approaching of the touch probe sensor.

Start profile data set		Search profile data set	
C01224/1	Starting velocity	C01224/2	Search speed
C01225/1	Start acceleration (deceleration as well)	C01225/2	Search acceleration (deceleration as well)
C01226/1	S-ramp time (identical in the two profile data sets)	C01226/1	S-ramp time (identical in the two profile data sets)

Note!

A changeover to the search profile data set is only carried out of the search speed has been set to (C01224/2) > "0"!

The exact time of the changeover to search profile data set occurs in the respective homing mode is indicated in the process descriptions of the homing modes.

Tip!

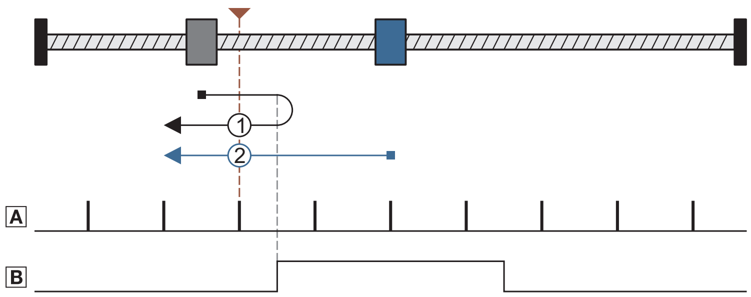
The result of setting a lower search speed in C01224/2 and a high search acceleration in C01225/2 is that deceleration to search speed takes place quickly and position detection is exact (at slower search speed).

Abbreviations used for the referencing modes:

Abbreviation	Meaning
>	Movement in positive direction
Set reference	Movement in negative direction
Lp	Positive travel range limit switch
Ln	Negative travel range limit switch
Rp	Positive edge pre-stop mark/pre-stop signal
Rn	Negative edge - pre-stop mark/pre-stop signal
TP	Touch probe or sensor reference signal/reference switch detected
Mlim	Torque limit value reached

Mode 4: >_Rp_<_TP

(from version 14.00.00)



A Touch probe
B Pre-stop mark

Mode 4	>	Rp	Set reference	TP	Offset path	Sequence profile (optional)
	Starting velocity		Search speed			Profile speed
Lp	Active	Active	Active	Active	Active	Active
Ln	inactive	Active	Active	Active	Active	Active

Process of case 1 ➜ Axis has not yet reached the pre-stop mark (*bHomingMark* = FALSE):

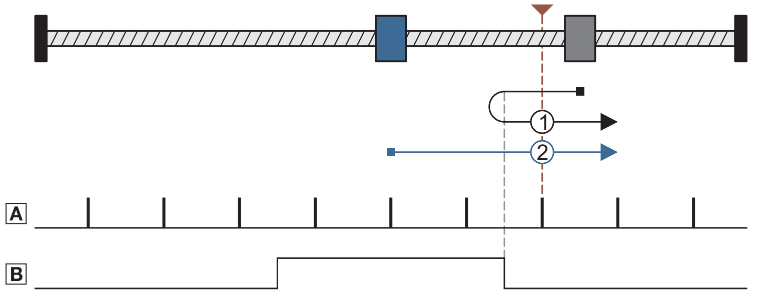
1. Movement in positive direction with start profile data set.
2. Reversing in case of positive edge at *bHomingMark* and, at the same time, activating the search profile data set for continued reference searching.
3. Negative edge at *bHomingMark* enables home position detection.
4. Following edge of the touch probe sensor sets home position.

Process of case 2 ➜ Axis is already on the pre-stop mark (*bHomingMark* = TRUE):

1. Movement in negative direction with search profile data set.
2. Negative edge at *bHomingMark* enables home position detection.
3. Following edge of the touch probe sensor sets home position.

Mode 5: <_Rp_>_TP

(from version 14.00.00)



A Touch probe
B Pre-stop mark

Mode 5	Set reference	Rp	>	TP	Offset path	Sequence profile (optional)
	Starting velocity		Search speed			Profile speed
Lp	inactive	Active	Active	Active	Active	Active
Ln	Active	Active	Active	Active	Active	Active

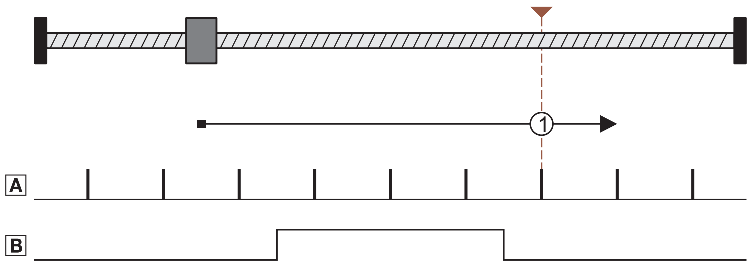
Process of case 1 → Axis has not yet reached the pre-stop mark (*bHomingMark* = FALSE):

1. Movement in negative direction with start profile data set.
2. Reversing in case of positive edge at *bHomingMark* and, at the same time, activating the search profile data set for continued reference searching.
3. Negative edge at *bHomingMark* enables home position detection.
4. Following edge of the touch probe sensor sets home position.

Process of case 2 → Axis is already on the pre-stop mark (*bHomingMark* = TRUE):

1. Movement in positive direction with search profile data set.
2. Negative edge at *bHomingMark* enables home position detection.
3. Following edge of the touch probe sensor sets home position.

Mode 6: >_Rn_>_TP



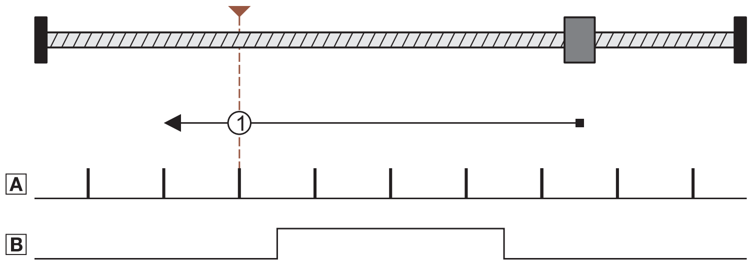
A Touch probe
B Pre-stop mark

Mode 6	>	Rn	>	TP	Offset path	Sequence profile (optional)
	Starting velocity		Search speed			Profile speed
Lp	Active	Active	Active	Active	Active	Active
Ln	inactive	inactive	inactive	inactive	Active	Active

Procedure:

1. Movement in positive direction with start profile data set.
2. Positive edge at *bHomingMark* activates search profile data set for further reference search.
3. Negative edge at *bHomingMark* enables home position detection.
4. Following edge of the touch probe sensor sets home position.

Mode 7: <_Rn_<_TP



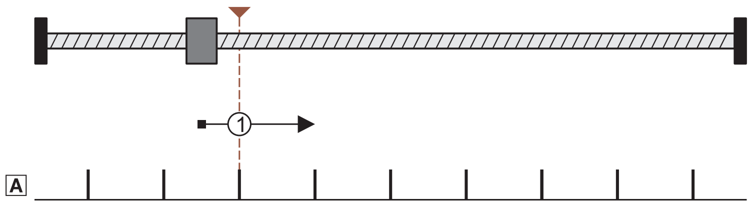
A Touch probe
B Pre-stop mark

Mode 7	Set reference	Rn	Set reference	TP	Offset path	Sequence profile (optional)
	Starting velocity		Search speed			Profile speed
Lp	inactive	inactive	inactive	inactive	Active	Active
Ln	Active	Active	Active	Active	Active	Active

Procedure:

1. Movement in negative direction with start profile data set.
2. Positive edge at *bHomingMark* activates search profile data set for further reference search.
3. Negative edge at *bHomingMark* enables home position detection.
4. Following edge of the touch probe sensor sets home position.

Mode 8: >_TP



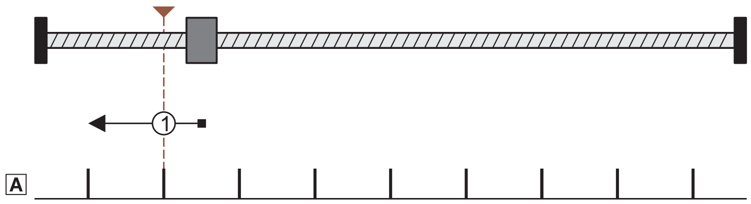
A Touch probe

Mode 8	>	TP	Offset path	Sequence profile (optional)
	Starting velocity			Profile speed
Lp	Active	Active	Active	Active
Ln	inactive	inactive	Active	Active

Procedure:

1. Movement in positive direction with start profile data set.
2. Following edge of the touch probe sensor sets home position.

Mode 9: <_TP



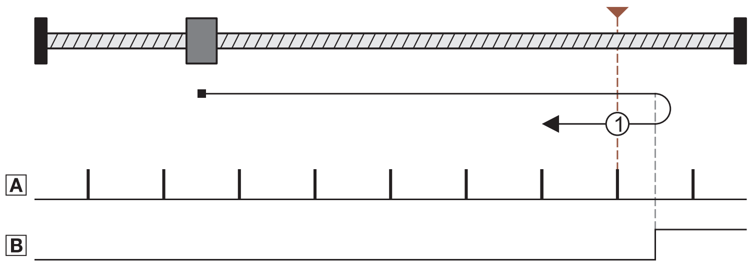
A Touch probe

Mode 9	Set reference	TP	Offset path	Sequence profile (optional)
	Starting velocity			Profile speed
Lp	inactive	inactive	Active	Active
Ln	Active	Active	Active	Active

Procedure:

1. Movement in negative direction with start profile data set.
2. Following edge of the touch probe sensor sets home position.

Mode 10: >_Lp_<_TP



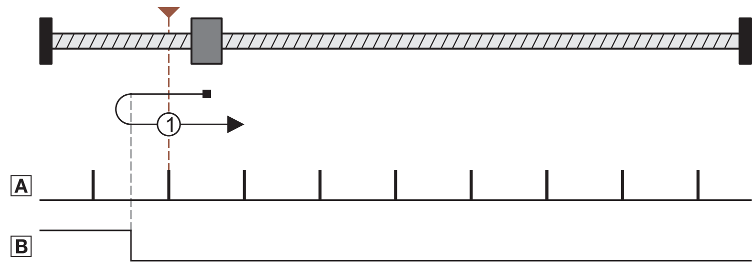
A Touch probe
B Positive travel range limit switch

Mode 10	>	Lp	Set reference	TP	Offset path	Sequence profile (optional)
	Starting velocity		Search speed			Profile speed
Lp	inactive	inactive	inactive	inactive	inactive	Active
Ln	inactive	inactive	Active	Active	Active	Active

Procedure:

1. Movement in positive direction with start profile data set.
2. Reversing when the edge of the positive travel range limit switch is positive and, at the same time, activation of search profile data set for continued reference searching.
3. Negative edge of the travel range limit switch enables home position detection.
4. Following edge of the touch probe sensor sets home position.

Mode 11: <_Ln_>_TP



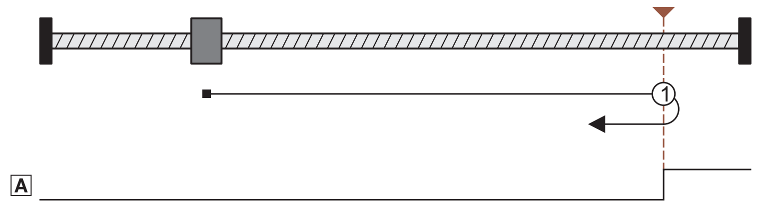
A Touch probe
B Negative travel range limit switch

Mode 11	Set reference	Ln	>	TP	Offset path	Sequence profile (optional)
	Starting velocity		Search speed			Profile speed
Lp	inactive	inactive	Active	Active	Active	Active
Ln	inactive	inactive	inactive	inactive	inactive	Active

Procedure:

1. Movement in negative direction with start profile data set.
2. Reversing when the edge of the negative travel range limit switch is positive and, at the same time, activation of search profile data set for continued reference searching.
3. Negative edge of the travel range limit switch enables home position detection.
4. Following edge of the touch probe sensor sets home position.

Mode 12: >_Lp



A Positive travel range limit switch

Mode 12	>	Lp	Offset path	Sequence profile (optional)
	Starting velocity			Profile speed
Lp	inactive	inactive	inactive	Active
Ln	inactive	inactive	Active	Active

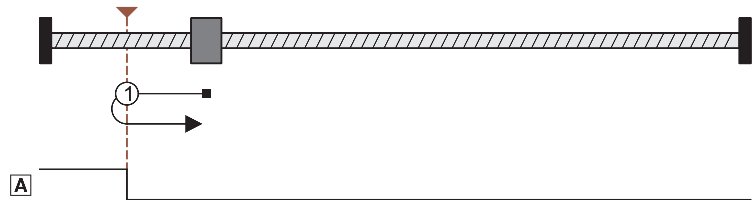
Procedure:

1. Movement in positive direction with start profile data set.
2. Positive edge of the travel range limit switch sets reference.

Note!

The load machine can also leave the travel range limit switch. There follows a return to the home position that was set with the positive edge of the travel range limit switch.
- It is possible that, as a result, the machine will remain on an operated limit switch.
- It is therefore recommended that a home value offset be set in C01227/1 in order to release the operated limit switch.

Mode 13: <_Ln



A Negative travel range limit switch

Mode 13	Set reference	Ln	Offset path	Sequence profile (optional)
	Starting velocity			Profile speed
Lp	inactive	inactive	Active	Active
Ln	inactive	inactive	inactive	Active

Procedure:

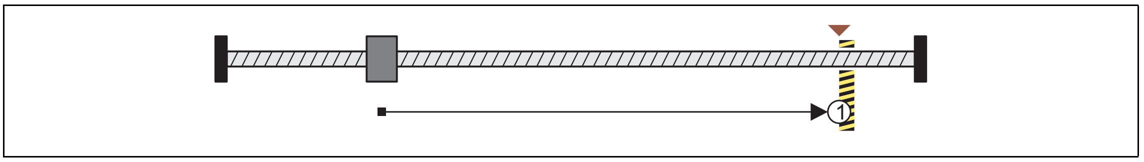
1. Movement in negative direction with start profile data set.
2. Positive edge of the travel range limit switch sets reference.

Note!

The load machine can also leave the travel range limit switch. There follows a return to the home position that was set with the positive edge of the travel range limit switch.

- It is possible that, as a result, the machine will remain on an operated limit switch.
- It is therefore recommended that a home value offset be set in C01227/1 in order to release the operated limit switch.

Mode 14: >_Mlim



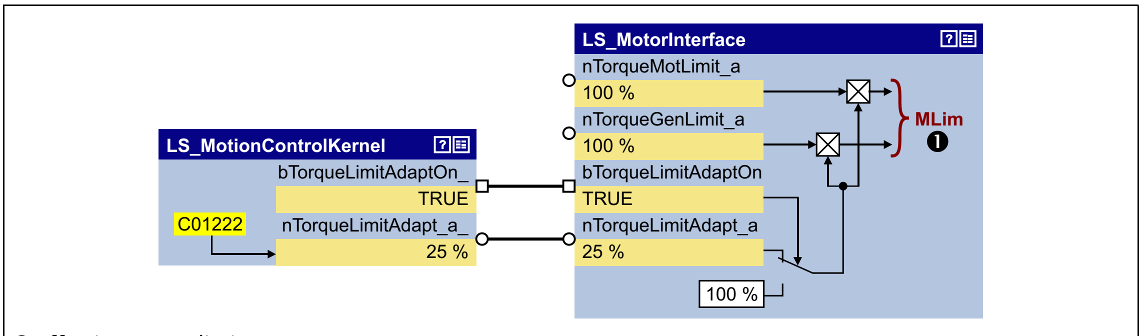
Mode 14	>	Mlim	Offset path	Sequence profile (optional)
	Starting velocity			Profile speed
Lp	Active	Active	Active	Active
Ln	inactive	inactive	Active	Active

Procedure:

1. Movement in a positive direction with reduced torque and start profile data set.
2. The reference is set if it is detected that the torque limit set in C01222 is exceeded beyond the time set in C01223 ("Homing to positive stop").
   - If a home value offset has not been set, this position is the home position.
   - If a home value offset has been set, correctly signed traversing by this offset takes place and the home position is set at the end of this travel distance.

Tip!

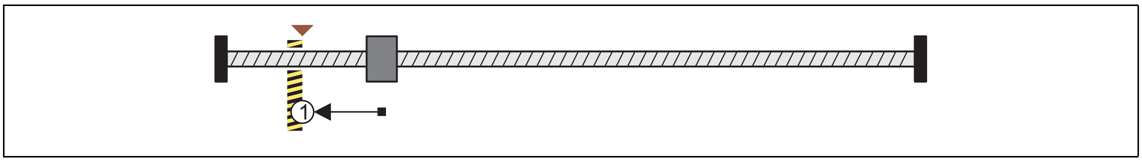
Set a home value offset in C01227/1 to avoid stopping at the positive stop.



❶ Effective torque limit

[9-26] Wiring for torque limit when homing to "Positive stop" (homing modes 14/15)

Mode 15: <_Mlim



Mode 15	Set reference	Mlim	Offset path	Sequence profile (optional)
	Starting velocity			Profile speed
Lp	inactive	inactive	Active	Active
Ln	Active	Active	Active	Active

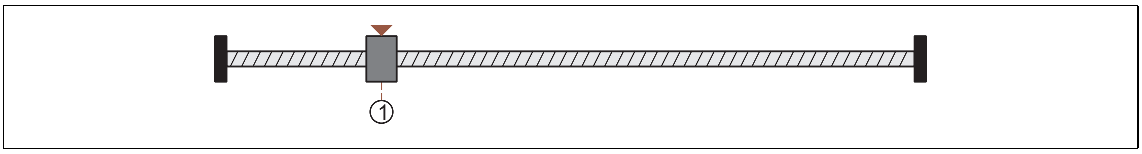
Procedure:

1. Movement in a negative direction with reduced torque and start profile data set.
2. The reference is set if it is detected that the torque limit set in C01222 is exceeded beyond the time set in C01223 ("Homing to positive stop").
   - If a home value offset has not been set, this position is the home position.
   - If a home value offset has been set, correctly signed traversing by this offset takes place and the home position is set at the end of this travel distance.

Tip!

Set a home value offset in C01227/1 to avoid stopping at the positive stop.

Mode 100: SetRef



Mode 100	SetRef	Offset path	Sequence profile (optional)
	-	Starting velocity	Profile speed
Lp	Active	Active	Active
Ln	Active	Active	Active

When the drive is at standstill, the measuring system is set via the bit 8 ("HomStartStop") in MCK control word. The current actual position now corresponds to the home position set in C01227/2 in the machine measuring system.

9.6.1.3 Home position & home value offset

If the home position is set in the course of the reference search, this determined position in the machine measuring system now corresponds to the home position set in C01227/2.

If a home value offset has been set in C01227/1, the drive continues correctly signed traversing by this offset at search speed after the home position has been found. The home position is not set until after this movement.

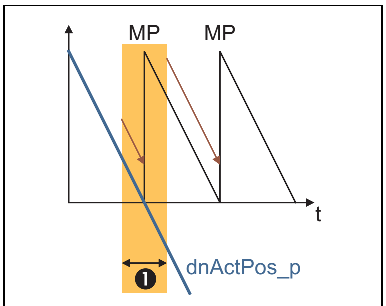
9.6.1.4 Traversing a sequence profile after completion of homing

If necessary, a sequence (following or subsequent) profile can be entered in C01228 in order to arrange for a positioning movement to be carried out immediately after completion of homing.

Note!

The positioning movement for the sequence profile set in C01228 is also carried out in the "referencing" mode. If a second sequence (following or subsequent) profile has been defined in the sequence profile, it is not automatically started!

9.6.1.5 Measurement of the distance between pre-stop mark/limit switch and TP/MP



❶ Jitter of the pre-switch off signal

Depending on how close the pre-stop mark is located to the encoder zero pulse and even jittering around it, it might occur that the drive performs one more or one less revolution after the home position detection has been enabled.

From version 14.00.00, the display parameter C01210/9 is available for this purpose. This parameter shows the distance between the signal which starts the home position detection and the detected touch probe signal or encoder zero pulse (MP) in [units].

The measurement is executed for the homing modes 4, 5, 6, 7, 10 and 11.

Referencing mode	Display in C01210/9 in [units]
4 / 5	$dnActPos\_p_{(TP/MP)}$ - $dnActPos\_p_{(bHomingMark\_TRUE\_to\_FALSE)}$
6 / 7	$dnActPos\_p_{(TP/MP)}$ - $dnActPos\_p_{(bHomingMark\_TRUE\_to\_FALSE)}$
10	$dnActPos\_p_{(TP/MP)}$ - $dnActPos\_p_{(bLimitSwitchPos\_TRUE\_to\_FALSE)}$
11	$dnActPos\_p_{(TP/MP)}$ - $dnActPos\_p_{(bLimitSwitchNeg\_TRUE\_to\_FALSE)}$

Note!

In C00926/3, the zero pulse can be shifted by ±179°. The setting only influences the generation of the zero pulse arithmetically. the pole position remains unchanged.
Possible application:

- When referencing to the zero pulse, you can shift the zero pulse in C00926/3 if the distance between pre-stop mark and encoder zero pulse is so small that the home position jitters due to the tolerance of the pre-stop mark. Thus, there is no mechanical shifting of the pre-stop mark and turning of the encoder on the shaft.

9.6.1.6 Option "Actual MCTRL position received at mains switch-off"

The encoder position can be saved with mains failure protection in the inverter. For this purpose, the bit 0 has to be set in C02652.

- From version 14.00.00, the "Actual MCTRL position received at mains switch-off" option can also be accessed in the »Engineer« via the parameterisation dialog for the basic function **Homing** (depending on the set position encoder system).
- If bit 0 is set in C02652, the actual position of the motor control is saved in the inverter (not in the memory module) and thus remains known to the drive control after mains switching.

The following overview shows the actual position after mains connection depending on the setting C02652/bit 0, C00490, C00422, C00925 (if resolver is selected) and the existence of the reference information:

C02652 bit 0	Reference available in case of mains switch-off (Status bit "HomePos Available")	Actual position and status reference in case of mains switch-on with setting		
		C00490 = 4 C00925 = 1	C00490 = 0, 3, 4, 10 C00422 = 0, 1 C00925 > 1 (with resolver)	C00490 = 3 C00422 = 2, 3, 4
		• Resolver with number of pole pairs = 1	• Incremental encoder (TTL) • Sin/cos encoder (Singleturn) • Resolver with number of pole pairs > 1 • No encoder	• Absolute value encoder (Multiturn)
0	0	0 incr.	0 incr.	$dnActPos_{\text{(mains switch-off)}} \pm\vartheta$
		Reference is unknown	Reference is unknown	Reference is unknown
	1	0 incr.	0 incr.	$dnActPos_{\text{(mains switch-off)}} \pm\vartheta$
		Reference is unknown	Reference is unknown	Reference available
1	0	$dnActPos_{\text{(mains switch-off)}} \pm\vartheta$	$dnActPos_{\text{(mains switch-off)}}$	$dnActPos_{\text{(mains switch-off)}} \pm\vartheta$
		Reference is unknown	Reference is unknown	Reference is unknown
	1	$dnActPos_{\text{(mains switch-off)}} \pm\vartheta$	$dnActPos_{\text{(mains switch-off)}}$	$dnActPos_{\text{(mains switch-off)}} \pm\vartheta$
		Reference available	Reference available	Reference available
$\pm\vartheta \equiv$ A rotation within the max. possible angle of rotation is possible at mains switch-off.				

9.6.2 Requesting the operating mode

Request for the "referencing" operating mode by means of the MCK control word:

MCK control word						
Bit 31	**...**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
			OpMode_Bit3	OpMode_Bit2	OpMode_Bit1	OpMode_Bit0
X	...	X	0	0	0	1
X = Status not significant						

If the **MCKInterface** is connected upstream to the **Motion Control Kernel** and if the operating mode is requested at the L_MckCtrlInterface FB, the *wOperationMode* and *bOperationMode_1...8* process inputs are available.

9.6.3 Execute homing

Control is exercised by means of bits 8 ... 10 in the MCK control word:

homing	MCK control word		
	Bit 10	Bit 9	Bit 8
	HomResetPos	HomSetPos	HomStartStop
Stop homing	0	0	0
Start homing	0	0	1
Set home position	0	1	X
Delete home position	1	0	X
X = Status not significant			

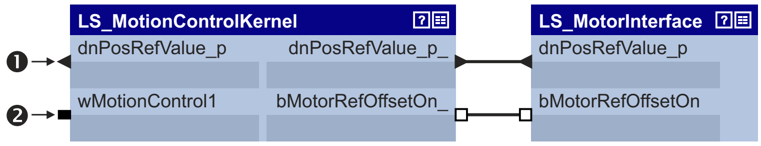
If the **Motion Control Kernel** is downstream from the **MCKInterface** in the circuit, the L_MckCtrlInterface FB provides the following process inputs for controlling the operating mode:

Designator Data type		Information/possible settings
bHomingStartStop BOOL		Start/stop homing • Only possible in the "referencing" operating mode. • Current status of the referencing process, see C01248/1.
	TRUE	If one of homing modes "4" ... "15" in C01221 is selected: **Start reference search** • The current status of the reference search is indicated via the status outputs *bHomingDone* and *bHomePosAvailable*. • If the 100_ Set_Ref_directly" referencing mode has been selected in C01221, the home position can be set manually via the *bHomeStartStop* input with the drive at a standstill. The current actual position is set as the home position (C01227/2)
		If homing mode "100: SetRef" is selected: **Setting the home position manually** • The home position is set manually with the drive at a standstill. The current actual position now corresponds to the reference position set in C01227/2 in the machine measuring system.
	TRUE↘FALSE	Stop homing. • If the *bHomingStartStop* input is reset to FALSE during active referencing, homing is cancelled and the drive is brought to a standstill.
bHomingSetPos BOOL		Set home position (Homing on the fly) • With referencing "on the fly", the home position of a machine can be set during ongoing movement. Jerking and compensating movements do not occur.
	FALSE↗TRUE	The position applied to the *dnPosRefValue_p* input of the LS_MotionControlKernel SB at the time of activation is set as home position if C01246/2 = "0: no TP" is set.
bHomingResetPos BOOL		Delete home position **Note:** With this function, positions are not deleted but only the status signals *bHomePosAvailable* and *bHomingDone* are reset. Setpoints and actual positions remain untouched until a renewed reference setting or homing.
	FALSE↗TRUE	The internal status "reference known" is reset. • The inverter is no longer referenced. • The process outputs *bHomePosAvailable* and *bHomingDone* are reset to FALSE.

9.6.3.1 Homing on the fly

With referencing "on the fly", the home position of a machine can be set during ongoing movement. Jerking and compensating movements do not occur.

The following illustration shows the relevant interfaces for selecting the home position at the SB LS_MotionControlKernel and SB LS_MotorInterface:



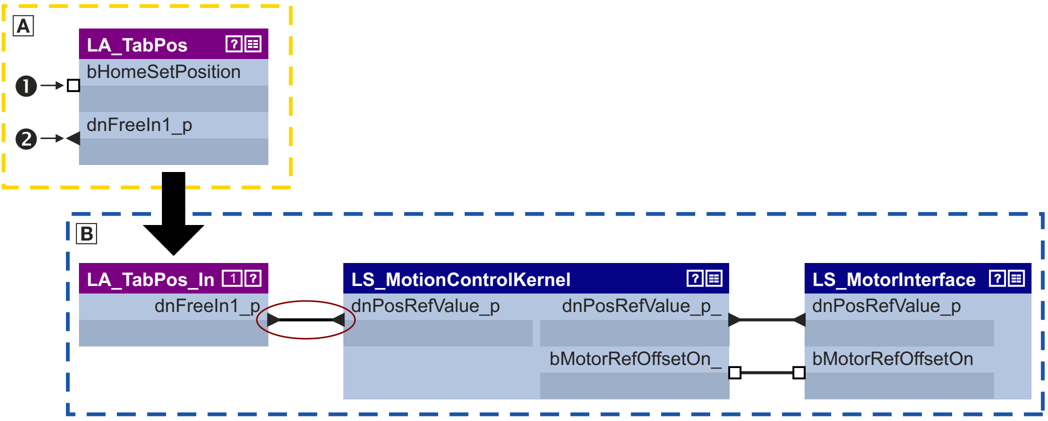
❶ Selection of the home position
❷ Setting of the home position with bit 9 ("HomeSetPos") = 1

[9-27] Interface for selecting the home position

In order that position selections can be accepted while homing on the fly, an additional connection is required in the application level.

- The following illustration shows the required modification using the example of the "table positioning" technology application.
- For transmitting the home position from the I/O level to the application level, the "free" input *dnFreeIn1_p* is used here:

A I/O level
B Application level



❶ Setting of the home position
❷ Selection of the home position

[9-28] Modified interconnection for selecting the home position

- Instead of using bit 9 in the MCK control word, the home position can also be set via a touch probe signal. The setting is made in C01246/2.

9.7 Manual jog

In this operating mode, the drive can be traversed manually in clockwise or anticlockwise direction ("jogging mode").

- As an option, it is possible to change over to a second speed during traversing.
- "Retraction" of operated (travel range) limit switches is also supported. Only traversing in the corresponding retracting direction is then possible.

Danger!

During manual jog, specially assigned profile parameters are effective. If these parameters are not set correctly, the drive may execute unexpected movements!

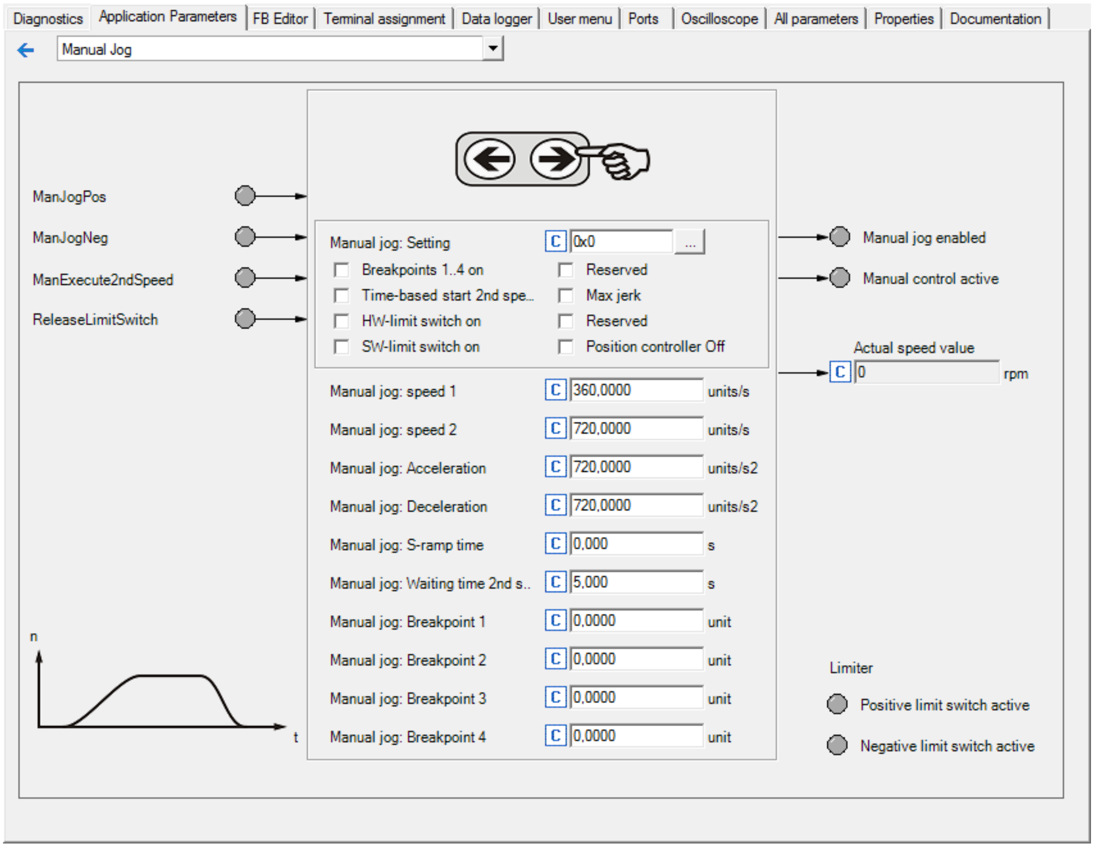
Stop!

In the Lenze setting, travel range monitoring is switched-off in C01230 by means of hardware limit switches and software limit positions for the "Manual jog" operating mode!

If travel range monitoring has been deactivated, the drive can travel into a mechanical limit during manual jogging and machine parts can be destroyed or damaged!

9.7.1 Parameter setting

Parameterisation dialog in the »Engineer«



Short overview of parameters for "manual jogging" mode"

Parameters	Info	Lenze setting Value	Unit
C01230	MCK: Manual jog setting	Bit coded	
C01231/1	Manual jog: speed 1	360.0000	units/s
C01231/2	Manual jog: Speed 2	720.0000	units/s
C01232/1	Manual jog: Acceleration	720.0000	units/s2
C01232/2	Manual jog: Deceleration	720.0000	units/s2
C01233/1	Manual jog: S-ramp time	0.000	s
C01235/1	Waiting time - 2nd speed	5.000	s
C01234/1	Manual jog: Breakpoint 1	0.0000	unit
C01234/2	Manual jog: Breakpoint 2	0.0000	unit
C01234/3	Manual jog: Breakpoint 3	0.0000	unit
C01234/4	Manual jog: Breakpoint 4	0.0000	unit

Note!

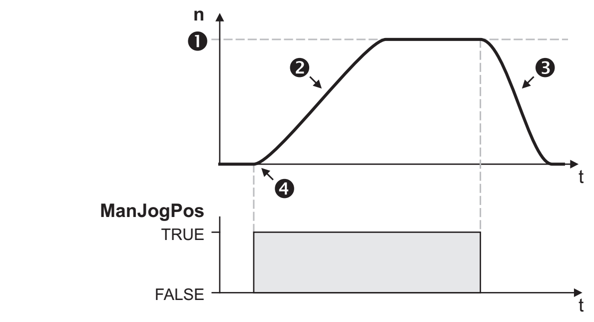
For trouble-free operation, the Machine parameters (at least gearbox factor and feed constant) must also be set correctly!

9.7.1.1 Functional settings

In C01230, various functional settings for manual jogging can be carried out in bit-coded form.

Function		Lenze setting
Bit 0	**Manual jog with breakpoint** During manual jog, a stop is automatically made at the breakpoint positions set in C01234/1...4. • The home position must be known for this function. • Travelling is continued when the pushbutton is "released" and pressed again. This button has been assigned the *ManJogPos* and/or *ManJogNeg* control function.	Off
Bit 1	**Time-based start of second speed** In manual jog mode, an automatic changeover to manual speed 2 takes place after a parameterisable waiting time. ▶ Time-based start of second speed (📖 662)	Off
Bit 2	**HW limit switch on** In the "manual jog" operating mode, a travel range monitoring mode via hardware limit switch is active. ▶ Limit position monitoring (📖 619)	Off
Bit 3	**SW limit switch on** In the "manual jog" operating mode, a travel range monitoring mode via parameterised software limit positions. ▶ Limit position monitoring (📖 619)	Off
Bit 4	Reserved	Off
Bit 5	**Maximum jerk** ▶ Setting or activation of maximum jerk for traversing profiles (📖 634)	Off
Bit 6	Reserved	Off
Bit 7	**Position controller off** In the "manual jog" operating mode, the position controller is deactivated. Thus, the compensation of the following error is switched off.	Off

9.7.1.2 Smooth start and quick stop of the drive



❶ C01231/1: Manual jog: Speed 1
❷ C01232/1: Manual jog: acceleration
❸ C01232/2: Manual jog: deceleration
❹ C01233/1: Manual jog: S-ramp time

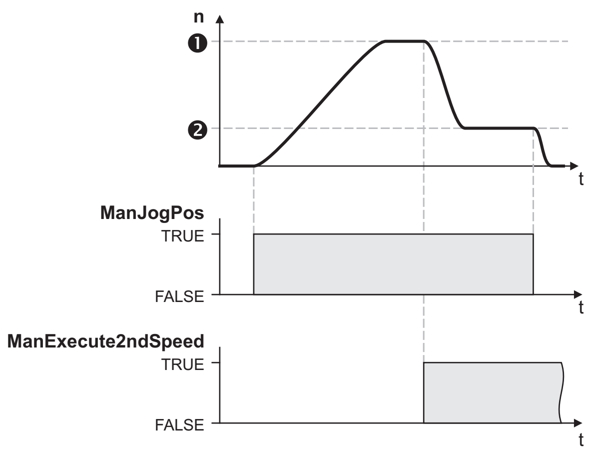
[9-29] Example: Smooth start and quick stop

- For accelerating and decelerating, different values can be set in C01232/1...2 so that smooth starting and quick stopping of the drive can be implemented.
- In order to reduce jerking, the two ramps can be set in such a way that they are s-shaped. This is done by entering a relative S-ramp time in C01233/1.

Tip!

Rapid deceleration (C01232/2) reduces the time from releasing of the "jog button" to actual stopping of the drive, with the result that it is easier to position the drive "by sight" and the desired stop position is not passed.

9.7.1.3 Second speed



❶ C01231/1: Manual jog: Speed 1
❷ C01231/2: Manual jog: Speed 2

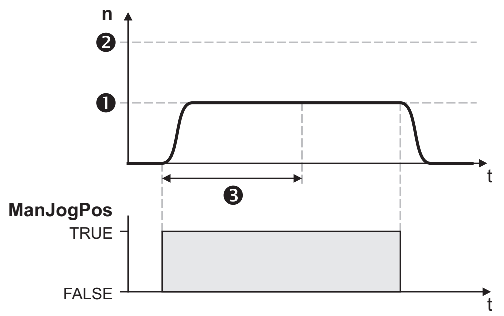
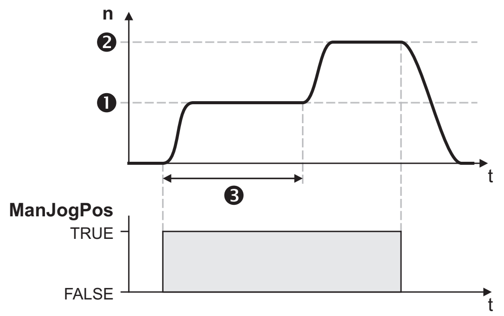
[9-30] Example: Change-over to second speed

- By setting control bit 6 (*ManExecute2ndSpeed*) in theMCK control word, you can changeover to a second speed (C01231/2) during traversing.

9.7.1.4 Time-based start of second speed

If this function has been activated by means of bit 1 in C01230 and a waiting time > "0 s" has been set in C01235/1, an automatic change-over to manual jog speed 2 takes place when the manual jog function is activated and after the waiting time has expired.

- When the waiting time is set = 0 s, the automatic change-over is deactivated.

Time-based start deactivated (C01230 - bit 1 = 0; Lenze setting)	**Time-based start activated** (C01230 - bit 1 = 1)
	
❶ C01231/1: Manual jog: Speed 1 ❷ C01231/2: Manual jog: Speed 2 ❸ C01235/1: Waiting time 2nd speed (Lenze setting: 5 s)	

[9-31] Time-based start of second speed

Tip!

By leaving the corresponding button for manual jog pressed down for longer than the waiting time and parameterising manual jog speed 2 so that it is higher than manual jog speed 1, you can enable longer distances to be travelled by means of manual jogging with this function.

9.7.2 Requesting the operating mode

Request for "manual jog" mode via the MCK control word:

MCK control word						
Bit 31	**...**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
			OpMode_Bit3	OpMode_Bit2	OpMode_Bit1	OpMode_Bit0
X	...	X	0	0	1	0
X = Status not significant						

If the **MCKInterface** is connected upstream to the **Motion Control Kernel** and if the operating mode is requested at the L_MckCtrlInterface FB, the *wOperationMode* and *bOperationMode_1...8* process inputs are available.

9.7.3 Executing manual jogging

Control is exercised by means of bits 4 ... 7 in the MCK control word:

Manual jog	**MCK control word**			
	Bit 7	**Bit 6**	**Bit 5**	**Bit 4**
	Release LimitSwitch	ManExecute 2ndSpeed	ManJogNeg	ManJogPos
Stop manual jog	0	X	0	0
Manual jog CW • With speed 1 (C01231/1)		0	0	1
Manual jog CW • With speed 2 (C01231/2)		1		
Manual jog in counter-clockwise direction • With speed 1 (C01231/1)		0	1	0
Manual jog in counter-clockwise direction • With speed 2 (C01231/2)		1		
No change in relation to previous state		X	1	1
Retract operated limit switch	1	0	0	0
X = Status not significant				

If the **Motion Control Kernel** is downstream from the **MCKInterface** in the circuit, the L_MckCtrlInterface FB provides the following process inputs for controlling the operating mode:

Designator Data type	**Information/possible settings**	
bManJogPos bManJogNeg BOOL	bManJogPos = TRUE: Manual jog right bManJogNeg = TRUE: Manual jog left Both inputs = TRUE: No change compared to previous state Both inputs = FALSE: Stop manual jog	
bManJogExecute2ndVel BOOL	Change over to speed 2 for manual jog	
	FALSE	Speed 1 (C01231/1) active.
	TRUE	Speed 2 (C01231/2) active.
bReleaseLimitSwitch BOOL	Retract operated limit switch	
	TRUE	Retract operated limit switch (in opposite direction)

Case 1: Reference known

If the reference is known and the software limit positions have been set, i.e. at least one software limit position is > 0, manual jogging is carried out until a position at the corresponding software limit position is reached unless manual jogging is aborted first. Overtravelling the set software limit positions is not possible.

Case 2: Reference not known

If the reference is not known, the travel range limits are only monitored via the limit switches (if connected).

If you end manual jogging manually by resetting *bManJogNeg* or *bManJogPos*, the drive is brought to a standstill at the rate of deceleration set for manual jogging.

9.7.3.1 Manual jog to limit position

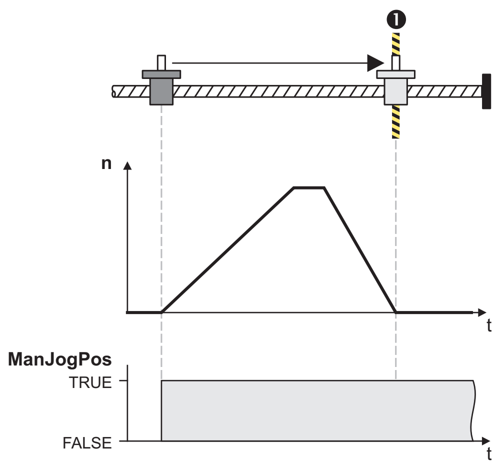
Stop!

In the Lenze setting, travel range monitoring is switched-off in C01230 by means of hardware limit switches and software limit positions for the "Manual jog" operating mode!

If travel range monitoring has been deactivated, the drive can travel into a mechanical limit during manual jogging and machine parts can be destroyed or damaged!

▸ Limit position monitoring (📖 619)

Manual jog to software limit position

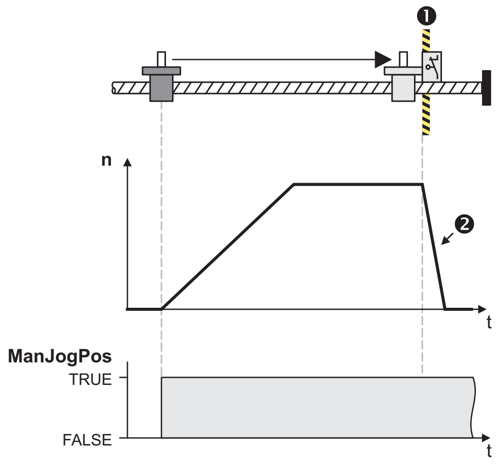


❶ Positive software limit position

[9-32] Example: Manual jog to positive software limit position

- If the reference is known and the software limit positions have been set and are active, positioning to the corresponding software limit position is carried out unless you manually stop manual jogging first by resetting control bit 4/5 (*ManJogPos*/*ManJogNeg*) in theMCK control word.
- The drive decelerates at the set rate of deceleration (C01232/2) to the position of the corresponding software limit position.

Manual jog to hardware limit position (limit switch)



❶ Positive limit switch
❷ Deceleration ramp of the quick stop function

[9-33] Example: Manual jog to positive limit switch

- When monitoring of the limit switches is switched on and a limit switch is approached during manual jog, the drive is braked within the deceleration time set for the quick stop function if the "TroubleQuickStop" error response is set in C00595/1 or C00595/2.

9.7.3.2 Retracting of an operated limit switch

If control bit 7 (*ReleaseLimitSwitch*) in the MCK control word is set, retracting from an operated limit switch is possible. Traversing is carried out in the corresponding retracting direction until the limit switch is not longer operated.

- If a direction preselection is made for retracting by means of control bit 4/5 (*ManJogPos*/*ManJogNeg*) in the retracting direction, travelling is continued, even after the limit switch has been left, until the corresponding control bit is reset.
- If, instead, a direction preselection is made against the retracting direction, the drive remains at a standstill.

Note!

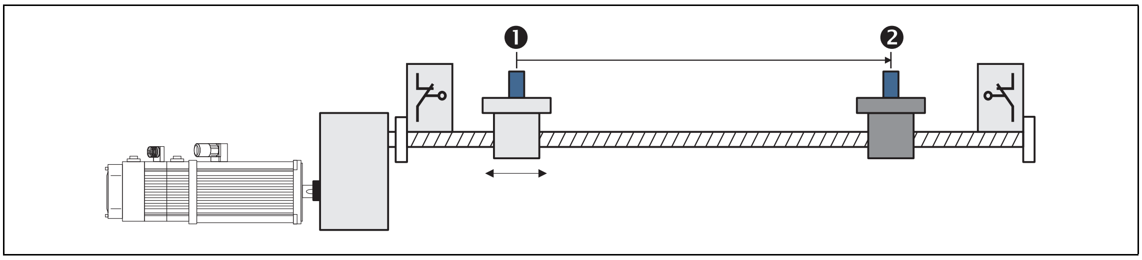
Retracting from a limit switch is only possible if this switch is still in the operated state, i.e. the corresponding limit switch input is still active. You must therefore make sure that, when travelling to a limit switch, its tripping mechanism is not "driven over" due to e.g. an excessively high mass or too much momentum so that the limit switch is no longer in the operated state as a result.

Tip!

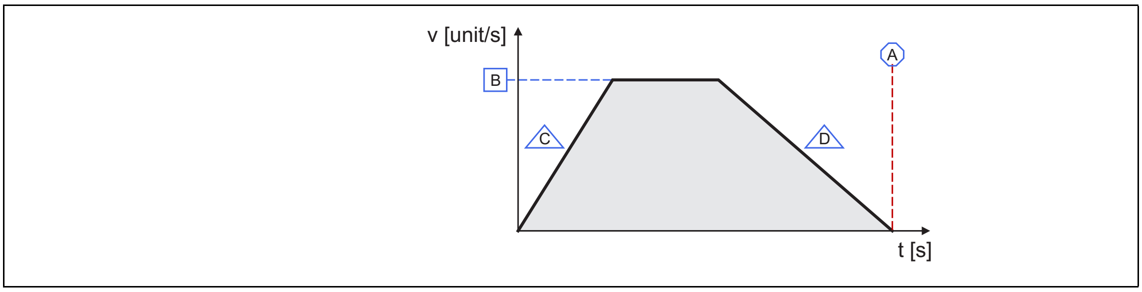
A limit switch that is in the operated state can also be moved away from again through manual jogging in the retracting direction by means of control bits 4/5 (*ManJogPos*/*ManJogNeg*).

9.8 Positioning

Positioning means that a workpiece/tool or material is moved from a starting position ❶ to a defined destination ❷:



To carry out positioning, a travel profile has to be stored in the inverter for at least the following profile parameter:

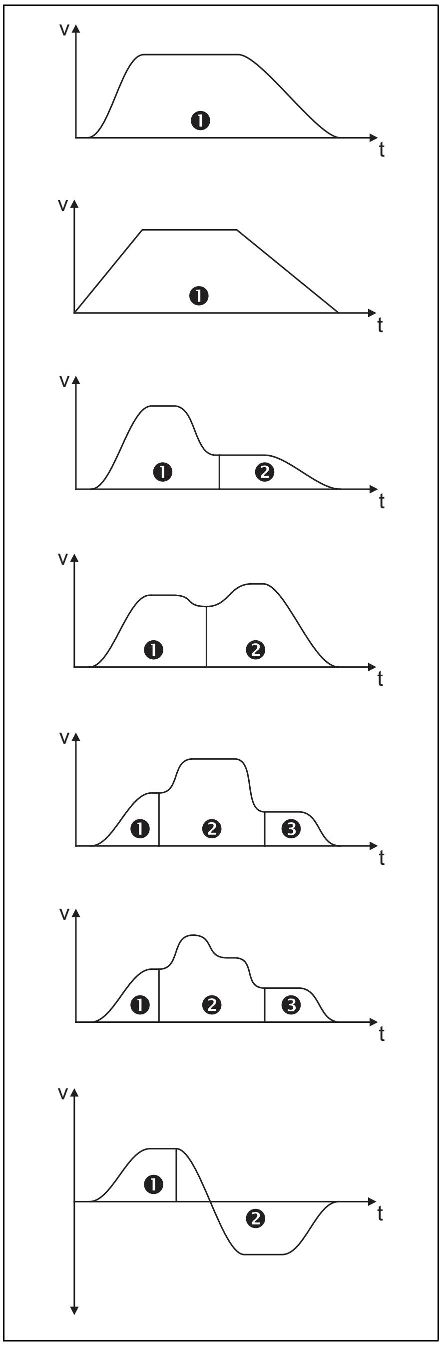


Symbol	Profile parameters
A	**Position** Target position or distance to be traversed.
B	**Velocity** Maximum velocity during the positioning process.
C	**Acceleration** Maximum acceleration during the positioning process.
D	**Deceleration** Maximum deceleration during the positioning process.

- A profile describes a motion task that can be converted to a rotary motion of the motor shaft by the **Motion Control Kernel** in the "Positioning" operating mode.
- A positioning process can be composed of a large number of profiles that are executed in a fixed manner.
- 15 different profiles can be parameterised for the 8400 TopLine inverter.
- You can find a detailed explanation of all profile parameters in the subchapter entitled "Profile entry". (📖 671)

9.8.1 Possible motion profiles

The following illustration shows different movement profiles that can be travelled in the "positioning" mode:



Asymmetrical trapezoidal profile
- with S-shaped ramps

Asymmetrical trapezoidal profile
- with linear ramps

Velocity changeover for profile with S-shaped ramps
- here: $v_{\text{Profile 1}} > v_{\text{Profile 2}}$ or $v_{\text{End1}} = v_{\text{Profile 2}}$

Velocity changeover for profile with S-shaped ramps
- here: $v_{\text{Profile 1}} < v_{\text{Profile 2}}$

Profile linkage

Profile linkage
- with speed override in profile 2

Profile linkage
- here: forwards/backwards profile

[9-34] Examples of possible motion profiles

9.8.2 Parameter setting

Short overview of parameters for "positioning" mode:

Parameters	Info	Lenze setting Value	Unit
C01216	MCK: Positioning setting	Bit coded	
C01300/1...15	Profile 1 ... 15: Mode	1: absolute (beeline)	
C01301/1...15	Profiles 1 ... 15: Position	360.0000	unit
C01302/1...15	Profile 1 ... 15: Speed	360.0000	unit/s
C01303/1...15	Profile 1 ... 15: Accel.	720.0000	unit/s2
C01304/1...15	Profile 1 ... 15: Decel.	720.0000	unit/s2
C01305/1...15	Profile 1 ... 15: Final speed	0.0000	unit/s
C01306/1...15	Profiles 1 ... 15: S-ramp time	0.000	s
C01307/1...15	Profiles 1 ... 15: Sequence profile	0	
C01308/1...15	Profile 1 ... 15: TP profile	0	
C01309/1...15	Profile 1 ... 15: TP signal source	3: TP-DigIn3	
C01310/1...15	Profiles 1 ... 15: PI position	0: Parameter value (C1301/n)	
C01311/1...15	Profiles 1 ... 15: PI speed	0: Parameter value (C1302/n)	
C01312/1...15	Profiles 1 ... 15: PI accel.	0: Parameter value (C1303/n)	
C01313/1...15	Profiles 1 ... 15: PI decel.	0: Parameter value (C1304/n)	
C01314/1...15	Profiles 1 ... 15: PI final speed	0: Parameter value (C1305/n)	
C01315/1...15	Profiles 1 ... 15: PI S-ramp time	0: Parameter value (C1306/n)	
C00595/9	MCK: Resp. to invalid PosModus	4: WarningLocked	
C00595/10	MCK: Resp. to invalid profile data	4: WarningLocked	
C00595/12	MCK: Resp. to invalid profile no.	4: WarningLocked	
C00595/14	MCK: React target out of travel range	4: WarningLocked	
C01210/1	MCK: Curr. feed	-	units
C01210/2	MCK: Curr. set position	-	units
C01210/3	MCK: Curr. actual position	-	units
C01210/4	MCK: Curr. following error	-	units
C01210/5	MCK: Positioning accuracy	-	units
C01210/6	MCK: Target position	-	units
C01211/1	Max. traversing speed 100%_C11	-	units/s
C01213/1	MCK: Max. traversing distance	-	units
C01242	MCK: Current pos profile number	-	
Greyed out = display parameter			

Note!

For trouble-free operation, the Machine parameters (at least gearbox factor and feed constant) must also be set correctly!

9.8.2.1 Functional settings

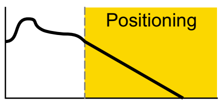
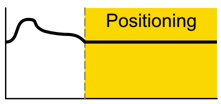
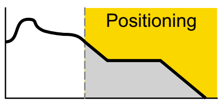
In C01216 , various functional settings regarding behaviour in the case of a changeover to the "positioning" mode can be made in bit-coded form.

- When this is being done, any mode change must be taken into account.
- Positioning with controller inhibit/enable is also possible if the corresponding setting is made.

Function		Lenze setting
Bit 0	**PosAbort at PosInit** When a changeover to "positioning" mode is made, ramping down at the rate of deceleration set in C01251 for normal stopping is carried out.	On
Bit 1	**PosExecute active at PosInit** When a changeover is made to the "positioning" mode, the specified profile is immediately executed if the "MCK PosExecute" control bit has also been set. If the MCK "PosExecute" control bit has not been set, the setpoint is continued.	Off
Bit 2	Reserved	Off
Bit 3	Reserved	Off
Bit 4	**ProfilStart at PosInit** **When it is changed to the "Positioning" operating mode, the defined profile is executed immediately without the need to set the MCK control bit "PosExecute".** **Note:** If the FB L_MckCtrlInterface writes to the MCK control word due to the setting in C01297 , a profile start may be prevented. Remedy: Set bit 0 in C01297 to "0". ▸ Alternative functions for control bit "PosExecute" (📖 606)	Off
Bit 5	**Maximum jerk** ▸ Setting or activation of maximum jerk for traversing profiles (📖 634)	Off
Bit 6	Reserved	Off
Bit 7	Reserved	Off

Note!

In the case of multiple selection, the "PosAbort at PosInit" function, which can be activated by means of bit 0, has priority over the other functions (see the following truth table).

Bit 4 ProfilStart	**Bit 1** PosExecute active	**Bit 0** PosAbort	**MCK control bit "PosExecute"**	**Behaviour when it is changed to "Positioning" operating mode**	
x	x	1	x		Ramping down the setpoint
0	**0**	**0**	x		Continuing the setpoint
0	1	0	0		
0	1	0	1		Positioning from the setpoint
1	x	0	x*		
printed in bold = Lenze setting; x = any state; * see note on bit 4 in table above					

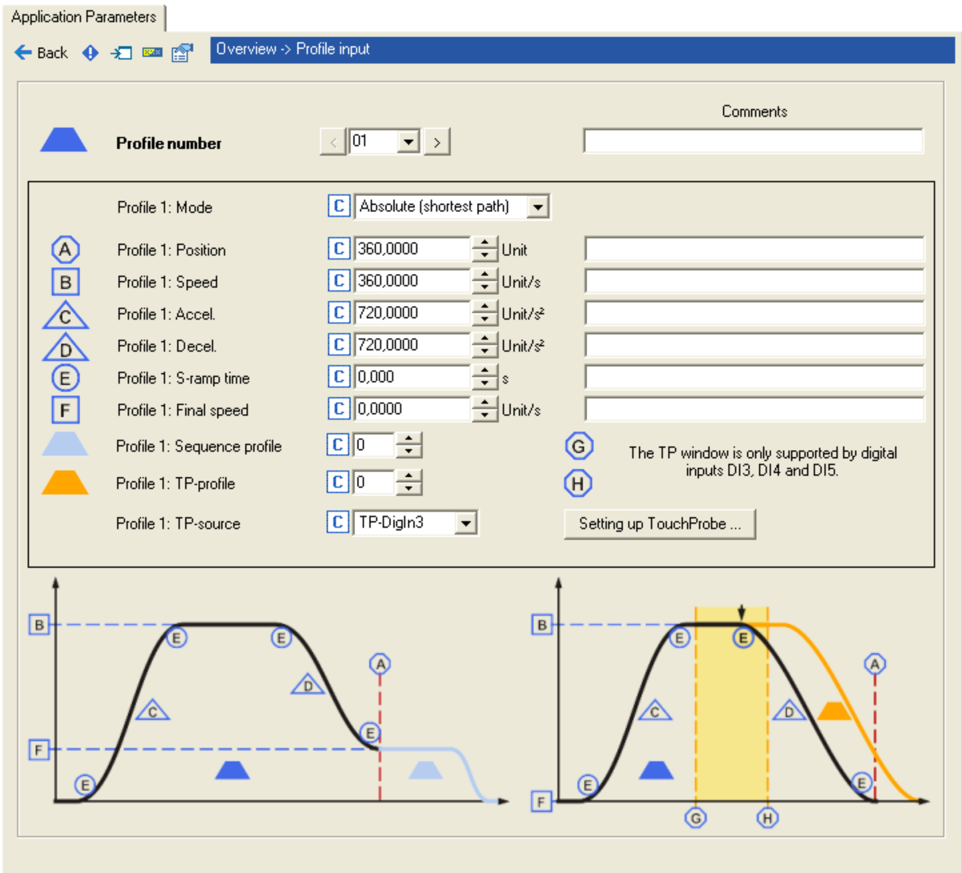
Related topics:

- Stipulation of the profile to be executed (📖 681)
- Starting/cancelling a traversing task (📖 682)

9.8.2.2 Profile entry

For setting the profile parameters, the »Engineer« provides a parameterisation dialog with easy profile entry via parameters and an extended profile entry (see marking).

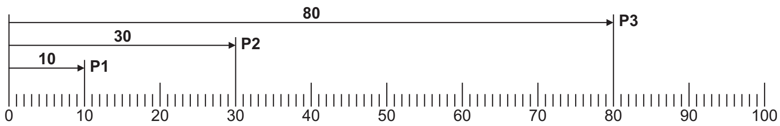
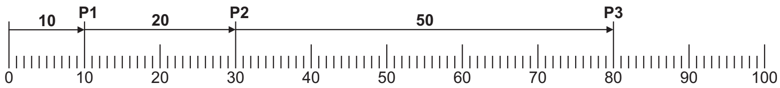
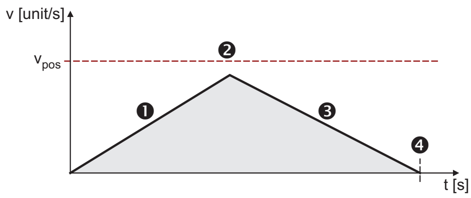
▸ Extended profile entry (📖 674)

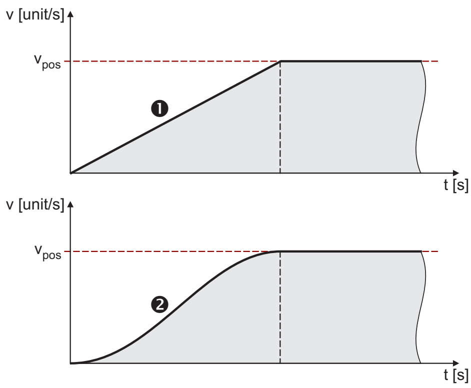


Profile entry via parameters

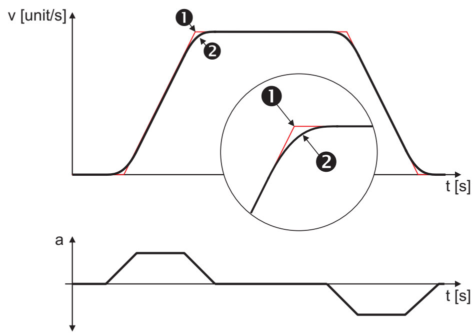
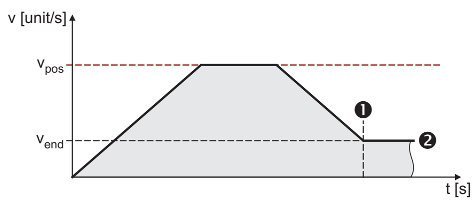
A profile is described by the following profile parameters:

Symbol	Profile parameters
	(Standard) profile Profile data set (profile numbers 1 ... 15), in which the profile data are stored.
	Mode (C01300/1...15) Selection of the way in which positioning is to be carried out. ▸ Positioning modes (📖 677)

Symbol	Profile parameters
A	**Position** (C01301/1...15) Target position or distance to be traversed. When the position is indicated, a distinction is made between absolute position and relative position: • An absolute position always indicates the distance to the defined zero position: Absolute position = Target position  • A relative position indicates the distance to the starting position (current position): Relative position = Target position - Starting position 
B	**Speed** (C01302/1...15) Maximum velocity during the positioning process. • Depending on the profile parameters of position, acceleration and deceleration , it is possible that the drive will not even reach the maximum speed. In this case, the graphic representation will be a trapezium instead of a triangle:  ❶ Acceleration ❷ Travelling speed (is not reached in this case) ❸ Deceleration ❹ Target position (or traversing distance)

C	**Acceleration** (C01303/1...15)
	Maximum acceleration during the positioning process.
	• Two types of acceleration are distinguished:
	  • Constant acceleration: the velocity increases linearly.
	  • Linearly increasing acceleration: Speed increases in S-shape.
	  A linearly increasing acceleration (S-profile) results from the setting of an S-ramp time (see more below).
	
	
	❶ Constant acceleration (L-profile)
	❷ Linearly increasing acceleration (S-profile)

D	**Deceleration** (C01304/1...15) Maximum deceleration during the positioning process.

<table>
<tr><th>Symbol</th><th colspan="2">Profile parameters</th></tr>
<tr><td rowspan="2">E</td><td colspan="2">S-ramp time (C01305/1...15)
Due to stipulation of an S-ramp time for a profile, the profile is executed with S-shaped ramps, i.e. acceleration and braking processes are initiated smoothly in order to reduce jerk and thus the stress on the drive components.
• The acceleration/deceleration stipulated in the profile is not achieved until after the specified S-ramp time.
• This kind of acceleration/deceleration is needed for sensitive machine parts with a certain amount of play.
• The unavoidable consequence of the slower increase in acceleration in the case of the S profile is that the positioning time is longer compared to the L profile, which is more efficient in terms of time.
▸ S-ramp time for jerk limitation (📖 679)</td></tr>
<tr><td>
</td><td>❶ Without jerk limitation (L profile)
❷ With jerk limitation (S profile)</td></tr>
<tr><td rowspan="2">F</td><td colspan="2">Final speed (C01305/1...15)
This specifies the velocity at which the drive is to start the next profile after reaching the target position. With a final speed not equal to "0", "velocity changeover" or "overchange" is possible, i.e. when the target position is reached, a second positioning process is started immediately without the drive coming to a standstill at the first target position.</td></tr>
<tr><td>
</td><td>❶ Target position
❷ Final speed (in this case, not equal to "0")</td></tr>
<tr><td></td><td colspan="2">Sequence profile (C01307/1...15)
for profile linkage/following block control
A special feature is automatic advancing to subsequent profiles with and without velocity changeover. For this purpose, the profile number of the desired subsequent profile (1 to 15) is simply set in the parameter "Sequence profile" (C01307/1...15) of a profile.
After execution of the profile (target position reached), the set following (subsequent) profile is started automatically. In this way, profile chains can be stipulated without additional control processes.
• If the profile parameter "Final speed" (C01305/1...15) is set to <> "0", there is a velocity changeover leading into the following (subsequent) profile at the set final speed.
• If "0" is set for the following (i.e. subsequent) profile, profile linkage does not take place.
• This function can be performed in all positioning modes.</td></tr>
<tr><td></td><td colspan="2">TP profile (C01308/1...15)
Profile number of the profile (1 ... 15) that is to be executed after a touch probe has been detected.
• If "0" is set, there will be no profile stepping through touch probe.
• Only relevant for positioning modes with touch-probe.
▸ Touch probe positioning (📖 678)</td></tr>
<tr><td></td><td colspan="2">TP signal source (C01309/1...15)
Selection of the signal source for touch probe detection.
• Only relevant for positioning modes with touch-probe.
▸ Touch probe positioning (📖 678)</td></tr>
</table>

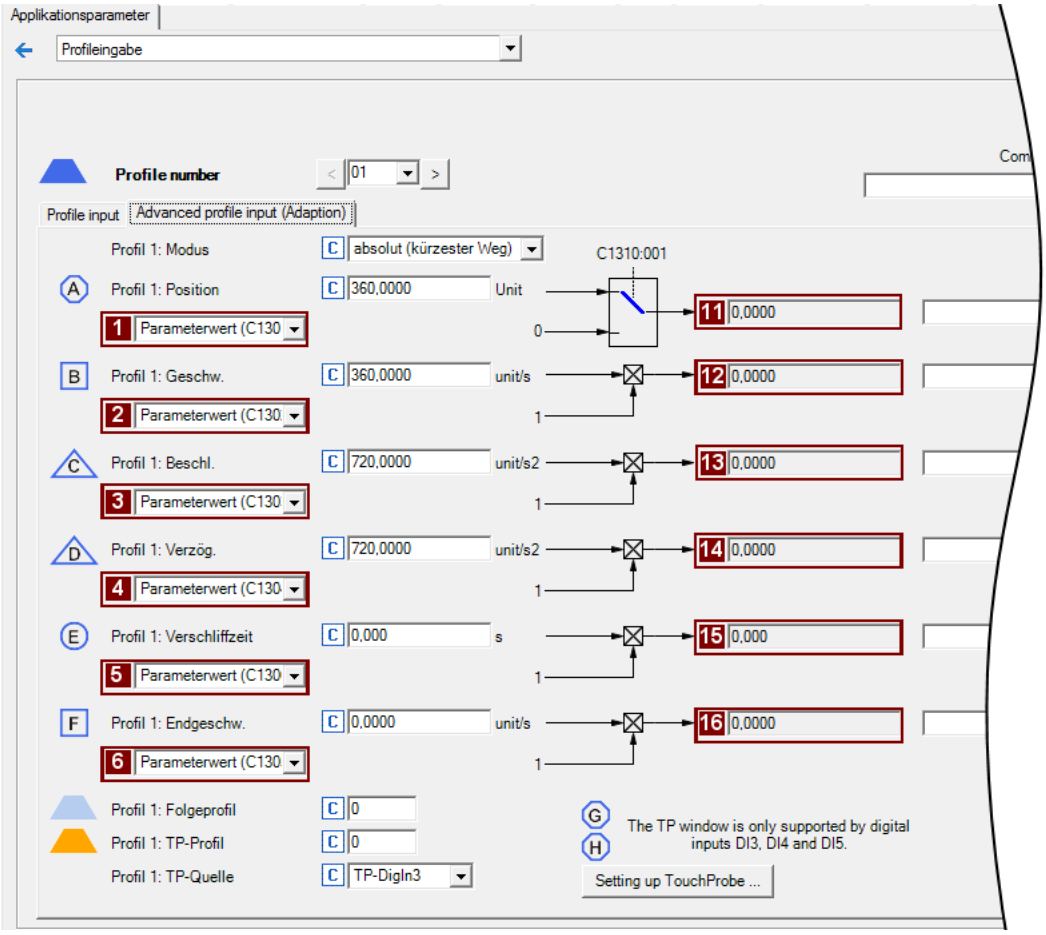
Extended profile entry

This function extension is available from version 16.00.00!

This function serves to extend the profile entry (1 ... 6) via a process data interface.

The profile data is processed for the profile start in different ways:

- In case of the profile position, the process input value is directly used for the profile start.
- The other profile data (profile speed, acceleration etc.) are evaluated with a percentage factor by means of the selected process input. The resulting value is used for the profile start.

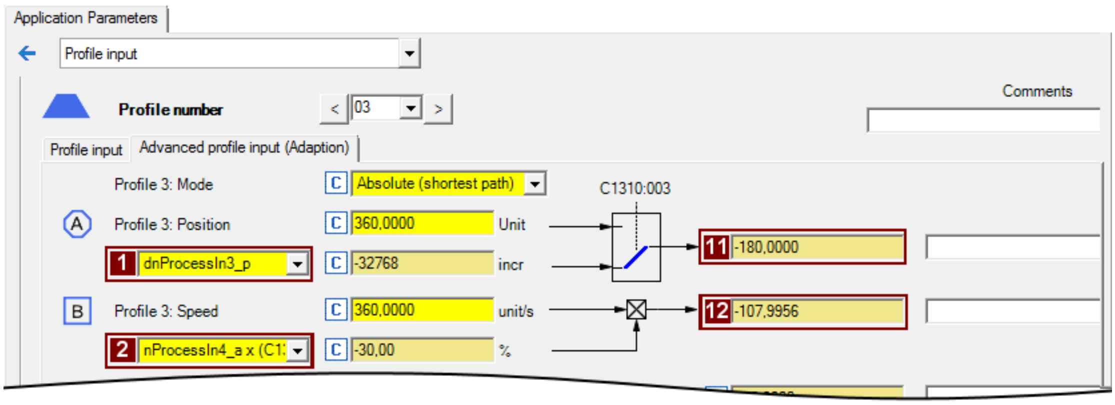


Parameters x = 1 ... 15		Lenze setting Value	Unit	Info
1	**Profile x: Position** (C01310/x)	0		Selection of the profile position value • 0: C01301/x • 1 ... 4: *dnProcessIn1..4_p* [incr]
2	**Profile x: Speed** (C01311/x)	0		Selection of the profile speed value • 0: C01302/x • 1 ... 8: *nProcessIn1..8_a* [%] x C01302/x Note: If the adaptation value for a speed value is higher than the speed limitation value C00909/x in the motor control, a following error takes place.
3	**Profile x: Acceleration** (C01312/x)	0		Selection of the profile acceleration value • 0: C01303/x • 1 ... 8: *nProcessIn1..8_a* [%] x C01303/x
4	**Profile x: Deceleration** (C01313/x)	0		Selection of the profile deceleration value • 0: C01304/x • 1 ... 8: *nProcessIn1..8_a* [%] x C01304/x
5	**Profile x: S-ramp time** (C01315/x)	0		Selection of profile S-ramp time value • 0: C01306/x • 1 ... 8: *nProcessIn1..8_a* [%] x C01306/x
6	**Profile x: Final speed** (C01314/x)	0		Selection of the profile final speed value • 0: C01305/x • 1 ... 8: *nProcessIn1..8_a* [%] x C01305/x Note: If the adaptation value for a speed value is higher than the speed limitation value C00909/x in the motor control, a following error takes place.
11	**Position** (C01320/x)			Display of the effective profile position
12	**Velocity** (C01321/x)			Display of the effective profile speed under consideration of the limitation in C01302/x by the maximum value in C01211/1 .
13	**Acceleration** (C01322/x)			Display of the effective profile acceleration
14	**Deceleration** (C01323/x)			Display of the effective profile deceleration
15	**S-ramp time** (C01325/x)			Display of the effective profile S-ramp time
16	**Final speed** (C01324/x)			Display of the effective final profile speed under consideration of the limitation C01305/x by the maximum value in C01211/1

Example

For the profile data set 3, the profile position and the profile speed are specified. For this purpose, the following values are used for the setting:

	Parameters	setting		Information
		Value	Unit	
1	**Profile 3: Position** (C01310/3)	-32768	incr	Selection of the profile position at the process input *LS_MotionControlKernel.dnProcessIn3_p* (32 bits)
2	**Profile 3: Speed** (C01311/3)	-30	%	Adaptation of the profile speed at the process input *LS_MotionControlKernel.dnProcessIn4_a* (16 bits)
11	**Position** (C01320/3)	-180	unit	Display of the resulting profile position **Profile 3**
12	**Velocity** (C01321/3)	-108	unit/s	Display of the resulting profile speed **Profile 3**



Note!

In order to avoid a following error, the resulting speed ≤ speed limitation value (C00909/x) should be set.

9.8.2.3 Positioning modes

For positioning, it is possible to select from different positioning modes in relation to the type of axis/application. These modes are described in the following table.

Stop!

In the positioning modes "continuous" and "relative", the "TroubleQuickStop" error response occurs in the Lenze setting when the maximum travel distance is exceeded (C01213/1). ▶ Monitoring of the maximum travel distance (📖 627)

Note!

For absolute positioning, the home (reference) position must be known!

- If absolute positioning (positioning modes 1, 4, 5, 8, 11, 12) is started although the home position is not known:
  - The error message set in C00595/8 is displayed (Lenze setting: "WarningLocked").
  - The error message "Ck08: Home position unknown" is entered into the logbook.

Positioning mode		Info
1	Absolute (shortest path)	Travelling along the axis takes place until an absolute position is reached via the shortest path. • Reference for the absolute position is the zero position.
8	Absolute (shortest path) to TP	
2	Continuous	In these two modes, no particular position is approached but travelling takes place at the traversing speed stipulated by means of the profile. • The values of the profile are used for accelerating and decelerating. • The direction of travel is determined by the sign of the traversing speed.
9	Continuous to TP	
3	Relative	The axis is traversed by a distance. • Reference for the distance is the target position of the previously executed profile.
10	Relative to TP	
Positioning modes for modulo measuring system (rotary table application) ▶ Activation of the modulo measuring system (📖 615)		
4	absolute (Cw)	Travelling along the axis takes place clockwise until an absolute position is reached. • Reference for the absolute position is the zero position. • In this direction, the zero position of the axis can be exceeded.
11	Absolute (Cw) on TP	
5	absolute (Ccw)	Travelling along the axis takes place counter-clockwise until an absolute position is reached. • Reference for the absolute position is the zero position. • In this direction, the zero position of the axis can be exceeded.
12	Absolute (Ccw) on TP	

9.8.2.4 Touch probe positioning

In case of a touch probe positioning, first the profile is executed according to the set profile parameters. If a touch probe is detected during the process, it is automatically changed to the profile defined in the "TP profile" profile parameter. This profile mode is also defined via the mode selection like in the following profile start processes.

If no valid TP profile is set, the travel request is aborted ("PosStop").

Preconditions for touch probe positioning

- The mode of the current profile data set contains the setting "to TP".
- For absolute positioning, the home (reference) position must be known.
- In the corresponding profile data set, the following profile parameters must be set in addition:
  - TP profile (C1308/x)
  - TP signal source (C1309/x)
- The touch probe interface is configured such that the selected TP signal source is enabled.
  - See chapter entitled "Touch probe detection". (📖 435)

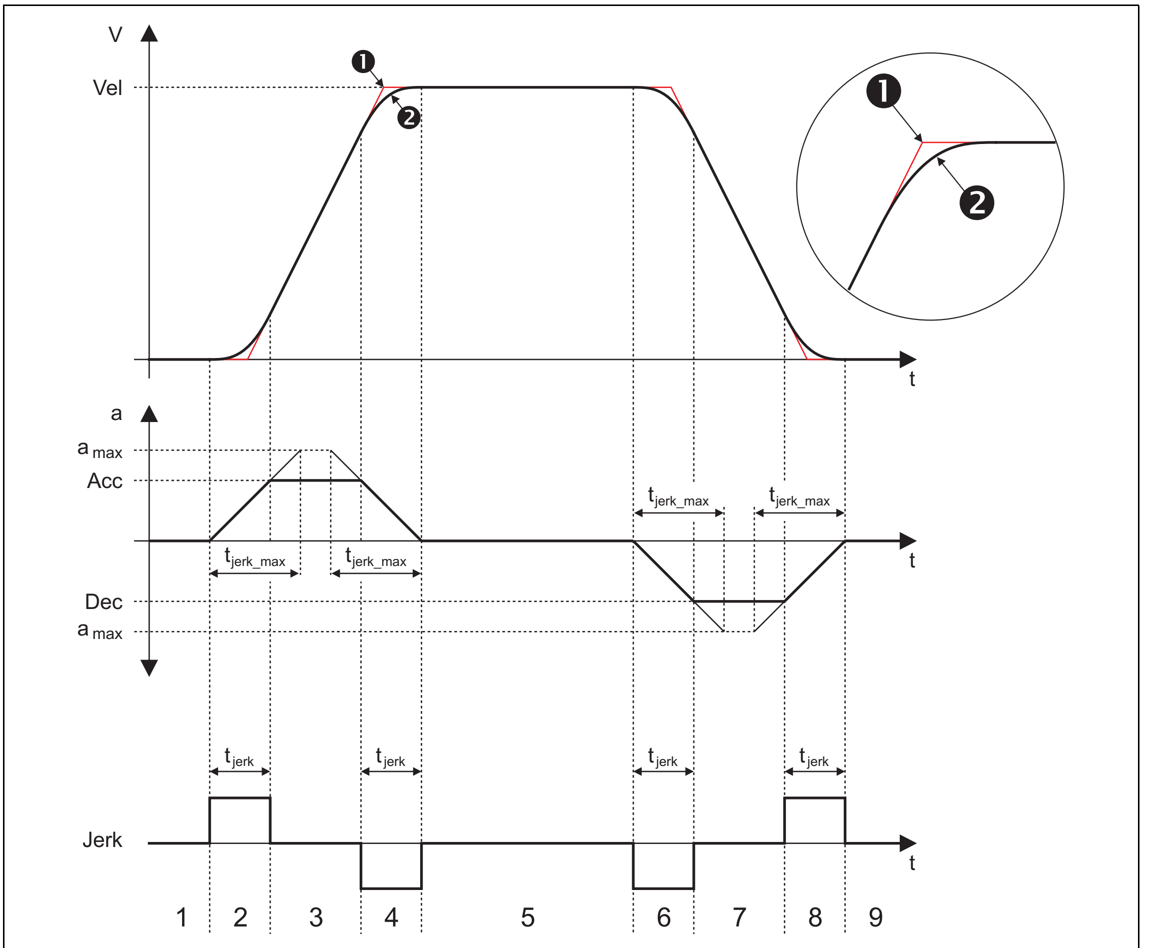
Note!

Go to C02810/x and select the edge, the digital input used for the connection of the touch-probe sensor is to respond. In the Lenze setting of C02810/x, no touch probe is detected!

9.8.2.5 S-ramp time for jerk limitation

The maximum jerk is defined by the selection of an S-ramp time t_{jerk_max}, after which the max. acceleration (a_{max}) only is reached.

- The actual jerk time t_{jerk} is reduced according to the actual acceleration Acc:



❶ Without jerk limitation
❷ With jerk limitation

1	Standstill		
2.	Acceleration with set jerk limitation	6.	Deceleration with set jerk limitation
3.	Acceleration according to acceleration profile (Acc)	6.	Deceleration according to deceleration profile (Dec)
4.	Reduce acceleration (jerk limitation)	8	Reduce deceleration (jerk limitation)
5th	Traversing with Vel according to speed profile	9	Standstill (position target reached)

Note!

If disproportionately high S-ramp times are set for low acceleration times, this may result in an incorrect profile generation.

Example: v = 100 mm/s, a = 1000 mm/s^2

→ t_{acc} = 0.1 s
→ t_{jerk_max} = 1 s

For this reason, you must only set plausible S-ramp times (this should not exceed half the value of t_{acc}).

9.8.3 Requesting the operating mode

Request for "positioning" mode by means of the MCK control word:

MCK control word							
Bit 31	...		Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				OpMode_Bit3	OpMode_Bit2	OpMode_Bit1	OpMode_Bit0
X	...	X	X	0	0	1	1
X = Status not significant							

If the **MCKInterface** is connected upstream to the **Motion Control Kernel** and if the operating mode is requested at the L_MckCtrlInterface FB, the *wOperationMode* and *bOperationMode_1...8* process inputs are available.

9.8.4 Carrying out positioning

9.8.4.1 Stipulation of the profile to be executed

The profile to be executed is stipulated by means of the MCK control word in bit-coded form:

Bit	Designation	Description							
24 ... 31	ProfileNo_Bit0 ... ProfileNo_Bit7	**Profile**	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Profile 0	0	0	0	0	0	0	0	0
		Profile 1	0	0	0	0	0	0	0	1
		Profile 2	0	0	0	0	0	0	1	0
		...	...							
		Profile 15	0	0	0	0	1	1	1	1
		All other possible settings are reserved for future extensions!								

If the **Motion Control Kernel** is downstream from the **MCKInterface**, the L_MckCtrlInterface FB provides the following processing inputs for stipulating the profile to be executed:

Designator Data type	Information/possible settings

wProfileNo
WORD
	Selection of the profile number
	• Optionally as a data word or binary coded.
	• In the Lenze setting, an operating mode change is carried out by the L_MckCtrlInterface FB at the same time as the profile is stipulated:
	  • If profile 0 is selected: Activation of "Speed follower" operating mode
	  • If profile 1 is selected: Activation of "Homing" operating mode
	  • If profile 2 is selected: Activation of "Manual jog" operating mode
	  • If profile 3 ... 15 is selected: Activation of "Positioning" operating mode
	• The operating mode change with profile number can be set in C01298/1...4.

bProfileNo_1 ... bProfileNo_8 BOOL	

Note!

The profile 0 is no valid profile for the "positioning" operating mode.

If a driving request with an invalid profile number is started, the response set in C00595/12 occurs (Lenze setting: "WarningLocked").

9.8.4.2 Starting/cancelling a traversing task

Control is exercised by means of bits 16 ... 19 in the MCK control word:

homing	MCK control word			
	Bit 19	Bit 18	Bit 17	Bit 16
	PosStop	Pos DisableFollowProfile	PosFinishTarget	PosExecute
Start travelling	0	0	0	0↗1
Complete interrupted profile	0	0	0↗1	0
Do not travel sequence profile	0	1	X	X
Cancel travelling*	1	X	X	X
*From version 02.00.00, more travel requests will be inhibited ("PosExecute" will be blocked).				
X = Status not significant				

If the **Motion Control Kernel** is downstream from the **MCKInterface** in the circuit, the L_MckCtrlInterface FB provides the following process inputs for controlling the operating mode:

Designator Data type	Information/possible settings	
bPosExecute BOOL	Start travelling	
	FALSE↗TRUE	Execute selected profile
bPosExecuteFinish BOOL	Complete interrupted profile	
	FALSE↗TRUE	A positioning process previously cancelled, e.g. by *bPosStop* or due to a device error, is resumed by travelling to the original target. From version 14.00.00, *bPosExecuteFinish* serves to permanently retravel to the original target even if the profile has been completed once or several times.
bPosDisableFollowProfile BOOL	Do not execute following profile (switch off profile linkage)	
	TRUE	Evaluation of the sequence profile number parameterised in C01307/1...15 for the selected profile is suppressed.
bPosStop BOOL	Cancel travelling	
	TRUE	Stop positioning From version 02.00.00, more travel requests will be inhibited ("PosExecute" will be blocked).

Tip!

Travel requests/profiles can also be started while the drive is running. The drive does not need to be at standstill.

Related topics:

- Target position monitoring (status "drive in target") (📖 625)
- Monitoring of the maximum travel distance (📖 627)
- Following error monitoring system (📖 628)

9.8.4.3 Override of the parameterised positioning mode

The setting of the positioning mode in MCK control word superimposes the mode selection via the "Mode" profile parameter (C1300/1...15). This means that for selected profiles the mode selection and the request of a touch probe positioning via process data are possible. The value set in C01300/1...15 is not overwritten. The following applies to the override:

A. Valid positioning mode in the MCK control word:
   The positioning mode set in the MCK control word is used.
B. Positioning mode in the MCK control word = 0:
   The positioning mode set in "mode" profile parameter (C1300/1...15) is used.
C. Invalid positioning mode in the MCK control word:
   Ck09Error message "Ck09: Invalid positioning mode". The response set in C00595/9 occurs (Lenze setting: "WarningLocked").

Bit	Designation	Description				
20 ... 23	PosModeBit0 ... PosModeBit3	**Positioning mode**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
		Positioning mode = setting in C01300/1...15	0	0	0	0
		Absolute (shortest path)	0	0	0	1
		Continuous	0	0	1	0
		Relative	0	0	1	1
		absolute (Cw)	0	1	0	0
		absolute (Ccw)	0	1	0	1
		Absolute (shortest path) to TP	1	0	0	0
		Continuous to TP	1	0	0	1
		Relative to TP	1	0	1	0
		Absolute (Cw) on TP	1	0	1	1
		Absolute (Ccw) on TP	1	1	0	0
		All other possible settings are reserved for future extensions!				

If the **Motion Control Kernel** is downstream from the **MCKInterface** in the circuit, the L_MckCtrlInterface FB provides the following process input for stipulating an override of the positioning mode:

Designator Data type	Information/possible settings	
wPosMode WORD	Override of the positioning mode set in the profile data • Only bit 0 ... bit 3 of *wPosMode* are evaluated.	
	0	Positioning mode = setting in C01300/1...15
	1	Absolute (shortest path)
	2	Continuous
	3	Relative
	4	absolute (Cw)
	5	absolute (Ccw)
	8	Absolute (shortest path) to TP
	9	Continuous to TP
	10	Relative to TP
	11	Absolute (Cw) on TP
	12	Absolute (Ccw) on TP
	All other possible settings are reserved for future extensions!	

9.8.4.4 Position teaching

The MCK control word can be used to initiate "teaching" or "latching" of the MCK setpoint position or the current position into the currently selected profile.

Bit	Designation	Description
14	PosTeachSetPos	"1" ≡ Teach MCK set position into the selected profile
15	PosTeachActPos	"1" ≡ Teach current position into the selected profile

If the **Motion Control Kernel** is downstream from the **MCKInterface**, the L_MckCtrlInterface FB provides the following process inputs for teaching:

Designator Data type	Information/possible settings	
bPosTeachSetPos BOOL	Teach MCK setpoint position	
	TRUE	Teach MCK setpoint position into the selected profile.
bPosTeachActPos BOOL	Teach current position	
	TRUE	Teach current position into the selected profile.

Note!

If request is made at the same time, the current position is taught.

9.9 Stop

If the "normal stop" operating mode is active, the drive is brought to a standstill with a parameterisable deceleration ramp.

9.9.1 Parameter setting

Short overview of parameters for "Normal stop" operating mode:

Parameters	Info	Lenze setting	
		Value	Unit
C01251/1	Normal stop: Deceleration	720.0000	units/s2
C01252/1	Stop: S-ramp time	0.000	s

9.9.2 Requesting the operating mode

Request for "Normal stop" operating mode by means of the MCK control word:

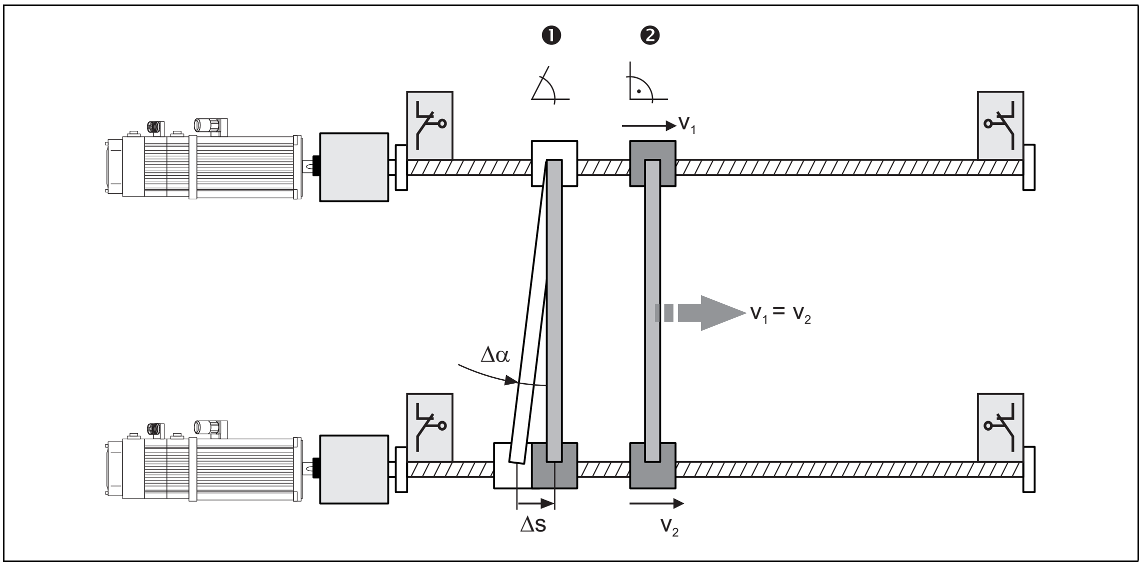
MCK control word						
Bit 31	**...**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
			OpMode_Bit3	OpMode_Bit2	OpMode_Bit1	OpMode_Bit0
X	...	X	0	1	0	0
X = Status not significant						

If the **MCKInterface** is connected upstream to the **Motion Control Kernel** and if the operating mode is requested at the L_MckCtrlInterface FB, the *wOperationMode* and *bOperationMode_1...8* process inputs are available.

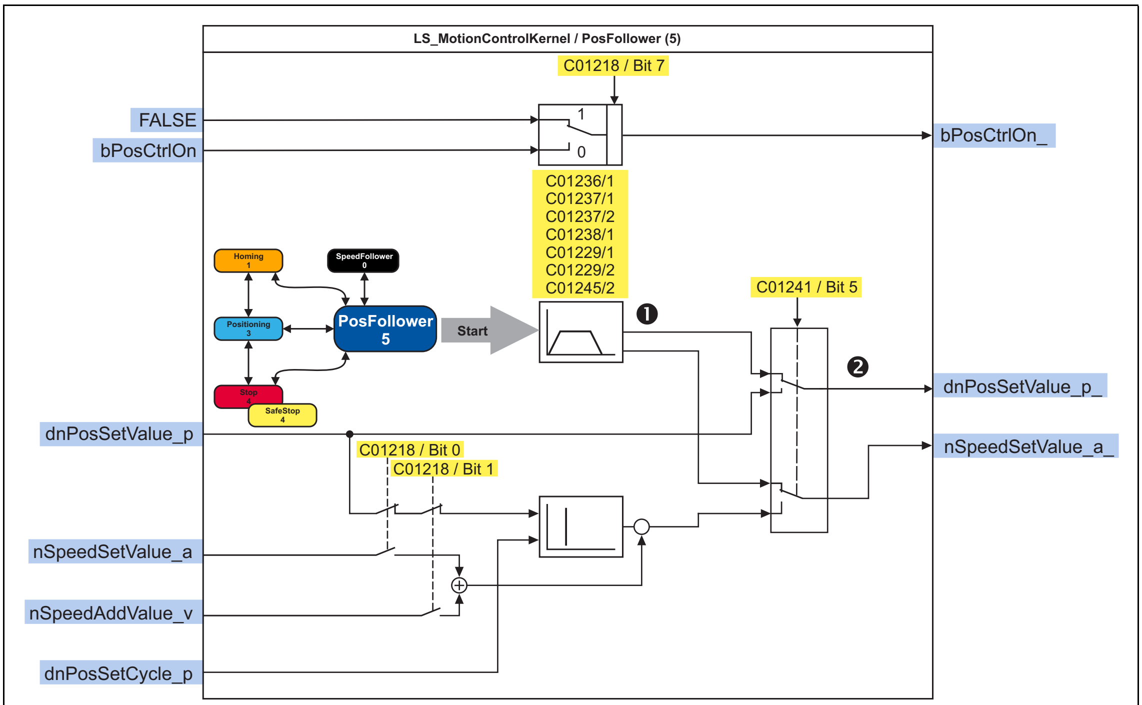
9.10 Position follower

In the "Position follower" operating mode, first one of the two drives covers a distance Δs (or an angle $\Delta\alpha$) to the synchronous position ❶.

At the time ❷, both drives follow a position setpoint with synchronous speed:



Signal flow



❶ Phase 1: assume synchronous position. In this phase, *dnPosSetValue_p* = constant.
❷ Phase 2: position follower operation

[9-35] Signal flow - position follower

9.10.1 Parameter setting

Short overview of parameters for "position follower" operating mode:

Parameters	Info	Lenze setting Value	Unit
C01218	MCK: PosFollower setting	Bit coded	
C01236/1	PosFollower: Sync. speed	360.0000	units/s
C01237/1	Pos follower: Sync. accel.	720.0000	units/s2
C01237/2	Pos follower: Sync. decel.	720.0000	units/s2
C01238/1	Pos follower: Sync. S-ramp time	0.000	s
C01292/1	PosFolger: Sync. Modus	1: absolute (beeline)	

9.10.1.1 Functional settings

In C01218, various functional settings for the position follower can be made in bit-coded form.

Function		Lenze setting
Bit 0	Speed FF control.: nSpeedSetValue_a	Off
Bit 1	Speed FF control: nSpeedAddValue_v	Off
Bit 2	HW limit switch on	On
Bit 3	SW limit switch on	On
Bit 4	Reserved	Off
Bit 5	Reserved	Off
Bit 6	Reserved	Off
Bit 7	Position controller off	Off

Speed feedforward control by means of nSpeedSetValue_a

If this function has been activated by means of bit 0 in C01218, the main setpoint *nSpeedSetValue_a* is used as the speed feedforward control value.

Speed feedforward control by means of nSpeedAddValue_v

If this function has been activated by means of bit 1 in C01218, the additive speed value *nSpeedAddValue_v* is used as the speed feedforward control value.

HW limit switch on

If this function has been activated by means of bit 2 in C01218, travel range monitoring by means of hardware limit switches is active in this mode.
▶ Limit position monitoring (📖 619)

SW limit switch on

If this function has been activated by means of bit 3 in C01218, travel range monitoring by means of parameterised software limit positions is active in this mode.
▶ Limit position monitoring (📖 619)

Position controller off

If this function is activated via bit 7 in C01218, the position controller is deactivated in this operating mode. Hence, following error control is switched-off.

9.10.2 Requesting the operating mode

Requesting the "position follower" operating mode by means of the MCK control word:

MCK control word						
Bit 31	**...**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
			OpMode_Bit3	OpMode_Bit2	OpMode_Bit1	OpMode_Bit0
X	...	X	0	1	0	1
X = Status not significant						

If the **MCKInterface** is connected upstream to the **Motion Control Kernel** and if the operating mode is requested at the L_MckCtrlInterface FB, the *wOperationMode* and *bOperationMode_1...8* process inputs are available.

9.10.3 Setpoint selection

The absolute position setpoint is stipulated via the *dnPosSetValue_p* process input.

Note!

- In order to prevent a jump between two phases, the position setpoint at the *dnPosSetValue_p* process input (see Signal flow - position follower (📖 686)) must not change while taking the synchronous position.
- The *bPosCtrlOn* process input must be set to TRUE so that position/angle control is active within motor control.

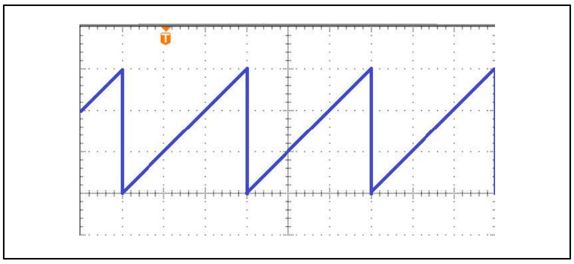
Compensating following errors

If the *bDeltaPosOn* process input is set to TRUE, a position difference (following error) is compensated for which is not internally generated from setpoint/actual positions but must be defined via the *dnDeltaPos_p* process input instead.

9.11 Modulo position follower

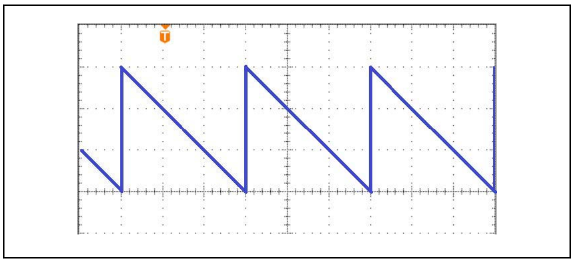
This function extension is available from version 17.00.00 onwards!

The **Position follower** operating mode can be used both for limited measuring systems and for modulo measuring systems. When using modulo systems, we should first look at the master value, as it is fundamentally different from that used in limited measuring systems. In limited measuring systems, the current position is always counted upwards and downwards. In Modulo measuring systems, there is an overflow after a defined path:



[9-36] Position in case of positive direction of rotation

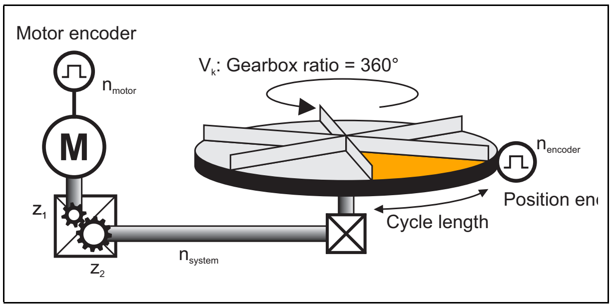
If the drive has a positive direction of rotation, the position value increases again after the overflow until the next overflow takes place.



[9-37] Position in case of negative direction of rotation

If the drive has a negative direction of rotation, the position value decreases towards zero, then jumps to the overflow value, and then decrases towards zero again.

This makes this measuring system ideal for rotary applications that start cyclically again from the beginning after a certain distance.



In the example, the overflow value comprises

- One segment or
- One complete revolution of the rotary table.

The position follower is aligned with the master value by first moving the position follower from the current position to the position of the master value. C01292/1 (PosFolger: Sync. Modus) is used to define how this alignment is to take place. Four choices are available:

- Setting 0:
  The slave does not perform any coupling movement.
  Any existing offset is retained.
- Setting 1:
  The slave performs positioning movements within its measuring system along the shortest path to the master position (K x *dnPosSetCycle_p*).
  The offset is "0"
- Setting 4:

The slave moves to the master position in a clockwise direction (K x *dnPosSetCycle_p*). The blocking zone in C01245/2 is taken into account.
The offset is "0"

- Setting 5:
  The slavemoves to the master position in an anti-clockwise direction (K x *dnPosSetCycle_p*).
  The blocking zone in C01245/2 is taken into account.
  The offset is "0"

9.11.1 Parameter setting

Brief overview of parameters for **Modulo position follower** operating mode:

Parameters	Info	Lenze setting Value	Unit
C01218	MCK: PosFollower setting	Bit coded	
C01236/1	PosFollower: Sync. speed	360.0000	units/s
C01237/1	Pos follower: Sync. accel.	720.0000	units/s2
C01237/2	Pos follower: Sync. decel.	720.0000	units/s2
C01238/1	Pos follower: Sync. S-ramp time	0.000	s
C01292/1	PosFolger: Sync. Modus	1: absolute (beeline)	
C01245/2	Blocking zone	1.0000	units

9.11.1.1 Functional settings

The following bit-coded functional settings for the position follower can be made in C01218:

Function		Lenze setting
Bit 0	Speed FF control.: nSpeedSetValue_a	Off
Bit 1	Speed FF control: nSpeedAddValue_v	Off
Bit 2	HW limit switch on	On
Bit 3	SW limit switch on	On
Bit 4	Reserved	Off
Bit 5	Reserved	Off
Bit 6	Reserved	Off
Bit 7	Position controller off	Off

Speed feedforward control by means of nSpeedSetValue_a

If this function has been activated by means of bit 0 in C01218, the main setpoint *nSpeedSetValue_a* is used as the speed feedforward control value.

Speed feedforward control by means of nSpeedAddValue_v

If this function has been activated by means of bit 1 in C01218, the additive speed value *nSpeedAddValue_v* is used as the speed feedforward control value.

HW limit switch on

If this function has been activated by means of bit 2 in C01218, travel range monitoring by means of hardware limit switches is active in this mode.
▶ Limit position monitoring (📖 619)

SW limit switch on

If this function has been activated by means of bit 3 in C01218, travel range monitoring by means of parameterised software limit positions is active in this mode.

▶ Limit position monitoring (📖 619)

Position controller off

If this function is activated via bit 7 in C01218, the position controller is deactivated in this operating mode. Hence, following error control is switched-off.

9.11.2 Requesting the operating mode

Requesting the "position follower" operating mode by means of the MCK control word:

MCK control word						
Bit 31	**...**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
			OpMode_Bit3	OpMode_Bit2	OpMode_Bit1	OpMode_Bit0
X	...	X	0	1	0	1
X = Status not significant						

If the **MCKInterface** is connected upstream to the **Motion Control Kernel** and if the operating mode is requested at the L_MckCtrlInterface FB, the *wOperationMode* and *bOperationMode_1...8* process inputs are available.

9.11.3 Setpoint selection

The absolute position setpoint is stipulated via the *dnPosSetValue_p* process input.

> **Note!**
>
> In order to prevent jumps between two phases, the position setpoint at the *dnPosSetValue_p* process input must not change while assuming the synchronous position.
>
> ▶ Signal flow - position follower (📖 686)
>
> The *bPosCtrlOn* process input must be set to TRUE so that position/angle control is active within motor control.

The signal *bReadyToOperate* of the FB **L_MckStateInterface** can be queried to start the position setpoint without a time offset. The drive is only ready to process setpoint values once this signal has switched to the HIGH level.

> **Note!**
>
> When evaluating *bReadyToOperate*, it must be noted that this signal also switches to HIGH when a different operating mode (e.g. **speed follower** operating mode) is selected and ready.

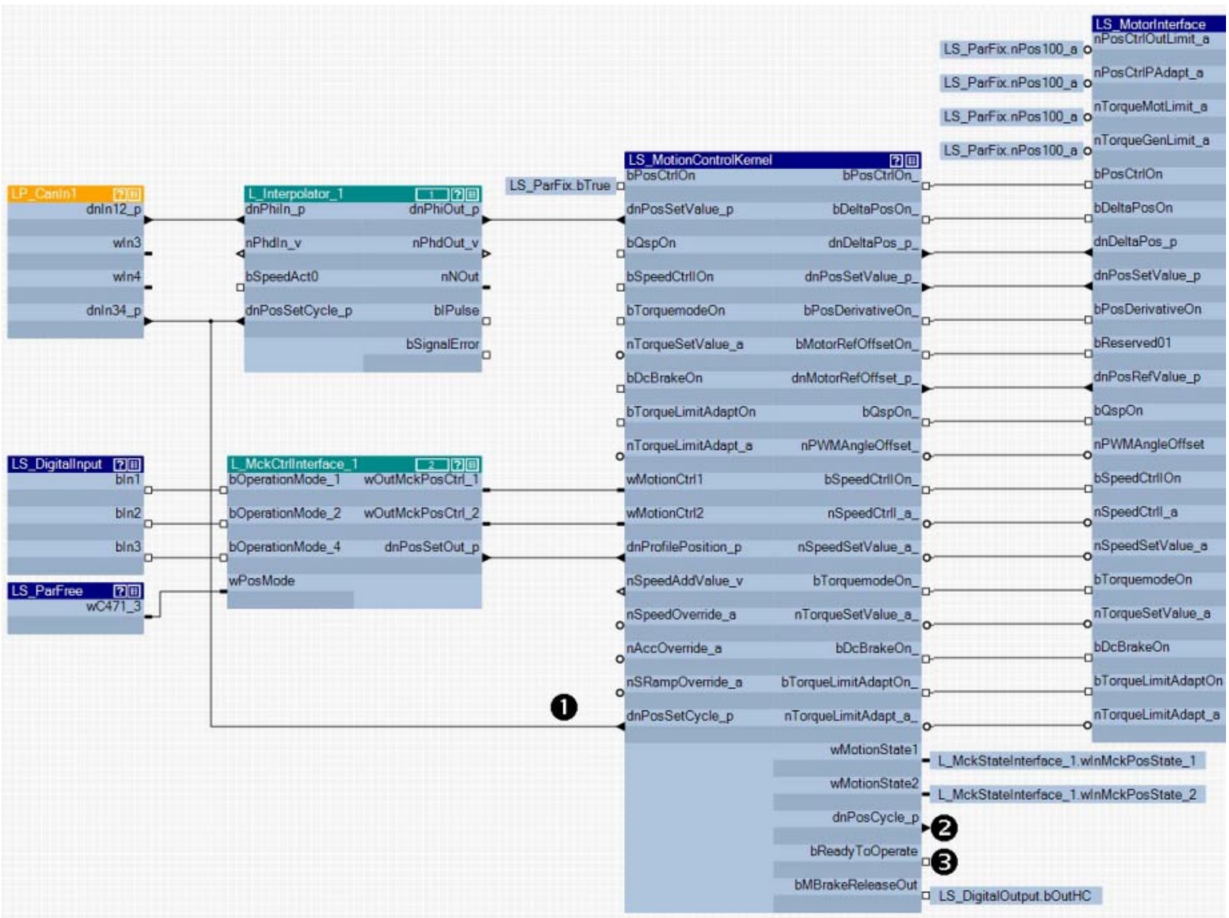
Compensating following errors

If input *bDeltaPosOn* is set to TRUE, the system compensates for any position differences (following errors) that are not internally generated from setpoint/actual positions but instead must be defined via the *dnDeltaPos_p* process input.

9.11.4 Important FB connections

To activate the **Modulo position follower** function, the master cycle length must be assigned to the *dnPosSetCycle_p* connection at the **LS_MotionControlKernel**. This can be performed using a **LS_ParFree_p** module.

Provided that the master value is transmitted from another inverter via a bus system, the master cycle length can alternatively also be transmitted using the bus. This is particularly useful if a connection is to be used for various applications for which various master cycle lengths are employed.



❶ Cycle length of master drive
❷ Cycle length of drive
❸ Ready to start the master value

The actual position setpoint must be connected to dnPosSetValue_p of LS_MotionControlKernel. Activate position follower operation via L_MCKCtrlInterface_1.

▸ Control inputs | "L_MckCtrlInterface" function block (📖 601)

9.11.5 Synchronisation

For smooth operation, the position setpoint must be specified every millisecond in the controller cycle. To this end, the inverter working as the slave must be synchronised with the position setpoint source.

▶ Synchronisation of the internal time base (📖 912)

9.11.6 Position setpoint interpolation

If the position setpoint is not specified in a 1-ms cycle, but rather at greater time intervals, the position setpoint must be interpolated using the FB **L_Interpolator_1**.

▶ Signal interpolation (📖 1629)

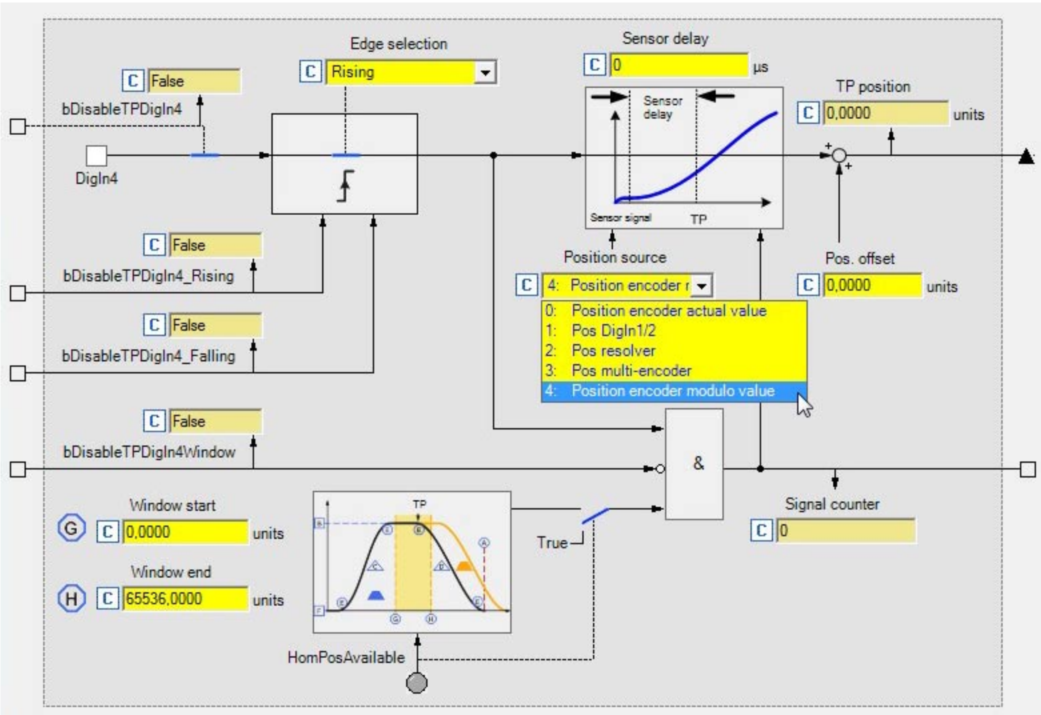
Note!

- The interpolator function is not coupled to the transfer clock of the master angle. This results in a time offset between the master angle and the output of the interpolator of one to two times the transfer clock time. This value changes after every device start.
- Generally, longer time intervals from one position setpoint to the next result in deterioration of the drive characteristics.

9.11.7 Touch probe

This function extension is available from version 20.00.00.

The touch probe function is provided via the system block **LS_TouchProbe**. The touch probe position output refers to the absolute position value. However, for a modulo application, the position value that refers to the modulo cycle is required. To achieve this, set option 4 ("Position encoder modulo actual value") in C02815/3 .... 9.



The functional dependencies of the setting C02815/3 .... 9 = 4 are defined below.

Designator Data type	Information/possible settings	
C02815/3 ... 9	Selection of the signal source for generating the TP position	
	0	Position encoder actual value
	1	Pos DigIn1/2
	2	Pos resolver
	3	Pos MultiEncoder
	4	Position encoder modulo actual value

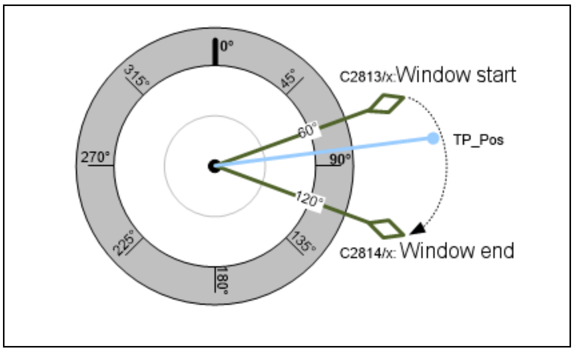
9.11.8 TP position output resulting from the setting in C02815/3 ... 9

An additional option 4 ("Position encoder modulo actual value") is added to the selection within the parameter C02815/3 .... 9. This selects the position encoder and activates the output of the triggered TP positions with a clock reference, provided that a cycle length has been set.

If no cycle length is set, the function is the same as C02815/3 .... 9 = 0. Setting the position source to position encoder modulo actual value ensures that the TP position generation is taken into account for the ongoing actual position value. This is necessary to guarantee that any TP referencing procedure or relative TP positioning is conducted with the required precision.

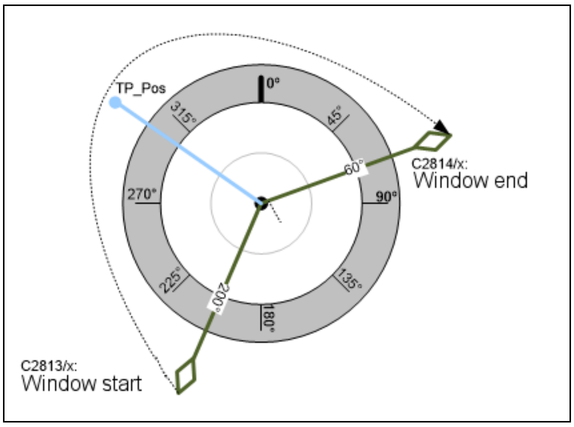
9.11.9 Handling of activity window

TP signal detection can be set within a window. If the determined touch probe position is present in this window, the position is valid and present at the process output. For the clock reference, the positions selected in C02813/1 .... 5 ("Window start") and C02814/1 .... 5 ("Window stop") must be taken into account separately.



Window start ≤ Window end

TP Pos in window:
TP_Pos ≥ Window start
and
TP_Pos ≤ Window end



Window start > Window end

TP Pos in window:
TP_Pos ≥ Window start
or
TP_Pos ≤ Window end

The window positions are not subject to any internal limitations if a measuring system with clock reference is used. The user is responsible for setting suitable window positions within the clock interval.

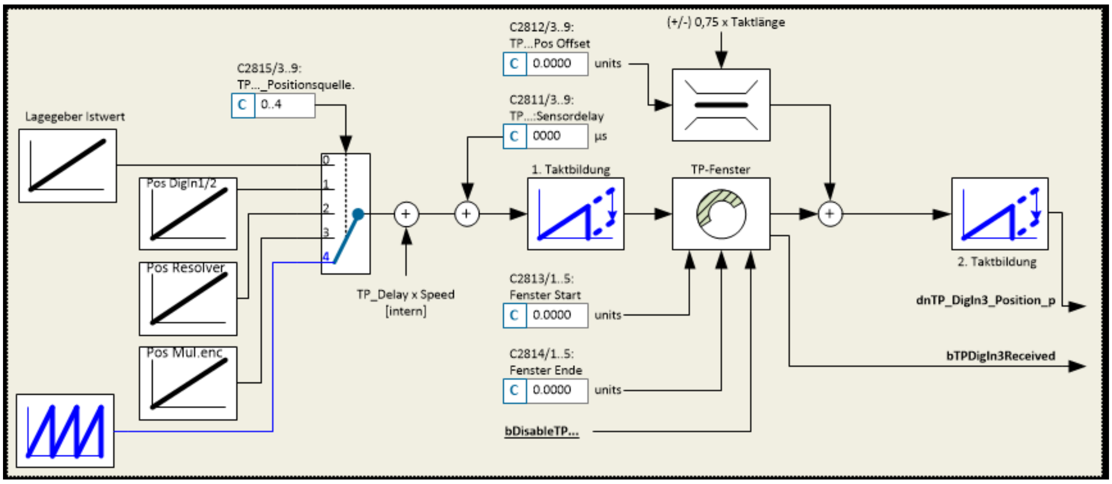
9.11.9.1 Handling of position offset

After detection of the TP signal and generation of the TP position, an offset set in C02812/3 .... 9 must be added. The set offset is subject to an internal limitation of 0.75 x cycle length before the addition to the TP position when a measuring system with clock reference is configured. Therefore, every other position in the cycle can be reached based on the TP position. No offset is added for a resulting cycle length of < 4 increments.

dnTP..._Position_p = TP_Pos + lim(C02812/x)[± 3/4 cycle length]

Correct in-cycle offsetting takes place before output to *dnTP..._Position_p*.

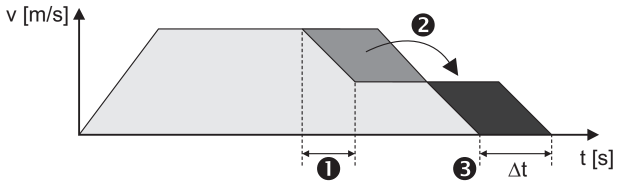
9.11.9.2 Signal flow — Touch probe



9.12 Override

An "Override" is the change of profile parameters and their acceptance during the positioning process.

- In this case the traversing profile has to be adapted accordingly, so that positioning is carried out exactly to the target position defined, even if for example a change in speed occurs during the positioning process ("Speed override"):



❶ Speed is reduced during the positioning process.
❷ In order to reach the position defined, the missing area has to be "attached" to the profile.
❸ Due to the speed reduction the positioning process takes longer (Δt).

[9-38] Override (here: speed override)

- The override for speed, acceleration, and S-ramp time affects all motion profiles that are controlled by the internal profile generator:
  - Manual jog
  - Retracting from limit switches
  - Homing
  - Point-to-point positioning

Note!

The online change of speed and acceleration is in effect from the start of the profile until the deceleration phase begins. Changing the deceleration phase by means of an override is therefore not possible!

- In the case of an override value of 0 % for the speed, the drive is brought to a standstill.
- Set override value for the speed ≤ speed limitation value (C00909/x). Otherwise a following error will be generated.
- In the case of an override value of 0 % for the acceleration, acceleration does not take place any longer.

The acceleration override also has the same effect on the deceleration ramp to the same extent specified, but only until the deceleration phase is initiated.

The override has no impact on:

- The "speed follower" operating mode
- Synchronising processes
- Setpoint controls via external setpoints
- Abort via input *bPosStop*
- Speed setpoint selection in case of error (e. g. "Fail-QSP")

9.12.1 Speed override

Activate override

If control bit 11 (*EnableSpeedOverride*) has been set to "1" in the MCK control word, a speed override is carried out in accordance with the stipulated override value.

If the **Motion Control Kernel** is downstream from the **MCKInterface** in the circuit, the L_MckCtrlInterface provides the following process input for activating the speed override:

Designator Data type	Information/possible settings	
bEnableVelOverride BOOL	Speed override	
	TRUE	Activate speed override

Stipulate override value

The override value is selected via the *nSpeedOverride_a* input at the SB LS_MotionControlKernel. The override value is a percentage of the speed of the current profile.

Designator Data type	Information/possible settings
nSpeedOverride_a INT	Value for speed override • Percentage multiplier (0 ... 199.99 %) for the currently active speed. • 16384 ≡ 100 % of the maximum traversing speed (display in C01211/1). • If the override value is 0 %, the drive is brought to a standstill.

Deactivate override

If control bit 11 (*EnableSpeedOverride*) is reset to "0", travelling takes place at the speeds that have been specified by means of the profile parameters. Acceleration/deceleration from the speed defined with override to the speed set in the profile takes places immediately in this case.

Deactivating the override within the braking/deceleration ramp does not have any effect.

Note!

If the value of a speed override is greater than the speed limitation value C0909/x in the motor control, a following error is generated. C0909/x should always be parameterised greater than or equalling the max. possible speed override value.

9.12.2 Acceleration override

Activate override

If control bit 12 (*EnableAccOverride*) in the MCK control word has been set to "1", an acceleration override takes place in accordance with the stipulated override value.

If the **Motion Control Kernel** is downstream from the **MCKInterface** in the circuit, the L_MckCtrlInterface FB provides the following process input for activating the acceleration override:

Designator Data type	Information/possible settings	
bEnableAccOverride BOOL	Acceleration override	
	TRUE	Activate acceleration override

Stipulate override value

The override value is stipulated via the *nAccOverride_a* input at the LS_MotionControlKernel SB. The override value is a percentage of the maximum acceleration that has been set for the respective operating mode (referencing, manual jog, positioning etc.) in the corresponding profile parameter.

The acceleration override results in a deceleration override. Both ramps are evaluated by multiplication by the acceleration override until the deceleration process sets in.

Designator Data type	Information/possible settings
nAccOverride_a INT	Value for acceleration override • Percentage multiplier (0 ... 199.99 %) for the currently active acceleration. • 16384 ≡ 100 % of the parameterised acceleration of the corresponding operating mode. • If the override value is 0 %, acceleration ceases.

Deactivate override

If control bit 12 (*EnableAccOverride*) is reset to "0", travelling takes place at the acceleration rates specified by means of the profile parameters. "Ramping up" from the acceleration defined with override to the acceleration set in the profile takes place immediately in this case.

9.12.3 S-ramp smoothing override

Activate override

If control bit 13 (*EnableSRampOverride*) in the MCK control word is set to "1", S-ramp smoothing override is executed according to the selected override value.

If the **Motion Control Kernel** is downstream from the **MCKInterface** in the circuit, the L_MckCtrlInterface FB provides the following process input for activating the S-ramp smoothing override:

Designator Data type	Information/possible settings	
bEnableSRampOverride BOOL	S-ramp smoothing override	
	TRUE	Activate S-ramp smoothing override

Stipulate override value

Optional stipulation of the override value is carried out via the *nSRampOverride_a* input at the LS_MotionControlKernel SB. The override value is a percentage of the S-ramp time set in the profile data.

Note!

If the *nSRampOverride_a* input remains unconnected or if an override value of "0 %" is selected, activation of the S-ramp override causes deactivation of the S-ramp time.

- Deactivation of the S-ramp time before the start of a profile with S-ramp time causes linear ramp generation.
- Deactivation of the S-ramp time during a traversing process, however, is not accepted immediately in the profile generator, but the profile generator checks automatically when an online change of the ramp form can be carried out and then initiates it automatically.

Tip!

Thus it is possible to start a travel profile with S-ramp time and then deactivate the S-ramp time, e.g. to traverse with a linear characteristic after reaching the profile speed.

Designator Data type	Information/possible settings
nSRampOverride_a INT	Value for S-ramp smoothing override • Percentage multiplier (0 ... 100 %) for the currently active acceleration. • 16384 ≡ 100 % of the parameterised S-ramp time (C01306/1...15). • Values > 16384 are ignored.

Deactivate override

If control bit 13 (*EnableSRampOverride*) is reset to "0", the S-ramp time specified by means of the profile parameters is used for profile generation.

9.13 Holding brake control

This basic function is used for low-wear control of a holding brake.

Danger!

Please note that the holding brake is an important element of the safety concept of the machine as a whole.

Thus, proceed very carefully when commissioning this system part!

Stop!

Holding brakes on Lenze motors are not intended for braking during operation. The increased wear caused by braking during operation can destroy the motor holding brake!

Note!

- **Deactivate automatic DC-injection braking when a holding brake is used!**
  - For this purpose, go to C00019 and set the Auto DCB threshold to "0".
  - Background: Controller inhibit is already activated by the holding brake control.
- If an electrically holding (self-releasing) brake is to be controlled instead of an electrically released (self-holding) brake, the trigger signal must be inverted!
  ▸ Functional settings (📖 707)
- Detailed information on mounting and electrical installation of the motor holding brake can be found in the documentation on the motor holding brake.

Intended use

Motor holding brakes are used to lock axes if the controller is inhibited or in case of "mains off" system status. This is not only important for vertical axes but also for e.g. horizontal axes which may cause various problems if the motion is not controlled.

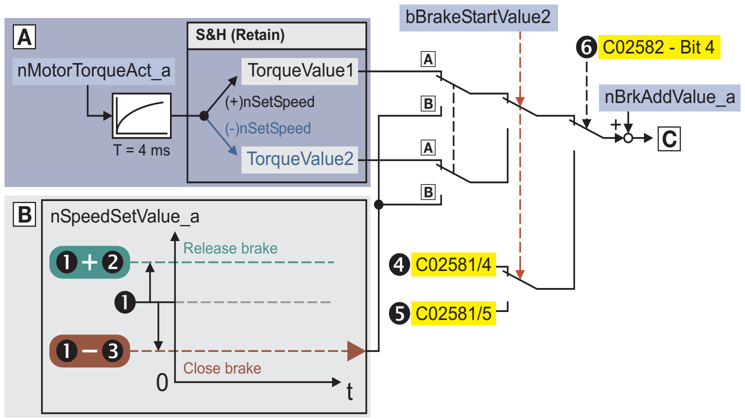
Examples:

- Loss of the reference information after mains OFF and further spinning of the drive.
- Collision with other moving machine parts.

9.13.1 Internal interfaces

In the function block editor, the LS_MotionControlKernel system block provides the following internal interfaces for the basic function "holding brake control":

inputs

Designator Data type	Information/possible settings	
bMBrakeRelease BOOL	Releasing/applying the brake in connection with the selected operating mode	
	FALSE	Apply brake. • During automatic operation, the internal brake logic controls the brake.
	TRUE	Release brake manually (forced release). • **Note!** The brake can also be released when the controller is inhibited! • During automatic operation, the internal brake logic is deactivated and the brake is released (supervisor operation). If a controller inhibit has been set by the brake control, it will be deactivated. • In semi-automatic operation, the brake is released including feedforward control.
bMBrakeStartValue2 BOOL	Selection of the torque feedforward control value ▶ Feedforward control of the motor before release	
	FALSE	Starting value 1 is active (see the following figure).
	TRUE	Starting value 2 is active (see the following figure).
	Creation of the feedforward control value for the release process of the brake: A Signal path for motor control with feedback B Signal path for motor control without feedback C Feedforward control value  ❶ C02581/1: Switching threshold ❷ C02581/2: Hysteresis for release ❸ C02581/3: Hysteresis for application ❹ C02581/4: Starting value 1 for manual feedforward control ❺ C02581/5: Starting value 2 for manual feedforward control ❻ C02582/Bit 4: Selection of the feedforward control value (automatic/manual selection)	
nMBrakeAddValue_a INT	Additive feedforward control value (speed or torque) in [%] for torque feedforward control when the respective control mode is started • For speed control: 100 % ≡ reference speed (C00011) • For torque control: 100 % ≡ maximum torque (C00057) ▶ Feedforward control of the motor before release	

Designator	Data type	Information/possible settings	
bMBrakeApplied	BOOL	Input for status detection via switching contacts at the brake • Only effective if bit 5 in C02582 is set to "1".	
		FALSE	Brake is released.
		TRUE	Brake is applied.

outputs

Designator	Data type	Value/meaning	
bMBrakeReleaseOut	BOOL	Trigger signal for switching element holding brake control via a digital output • Use bit 0 under C02582 to activate inverted switching element triggering. ▸ Functional settings	
		FALSE	Apply brake.
		TRUE	Release brake.
bMBrakeReleased	BOOL	"Brake released" status signal considering the brake release time • When the holding brake is triggered to close, *bMBrakeReleased* is immediately set to FALSE even if the brake closing time has not yet elapsed!	
		TRUE	Brake released (after the brake release time has expired).

Stop!

The digital outputs are not suitable for the "direct" control of a holding brake!

- Connect the digital output connected to the trigger signal *bMBrakeReleaseOut* with a relay or power contactor which switches the brake supply.
- When a power contactor is used, the response and release time of the earth contact is added to the response and release time of the brake. Both times must also be considered for parameterising the closing and opening time of the holding brake

Tip!

For direct triggering of a holding brake, the high current output of the 8400 TopLine inverter can be used that can switch max. 2.5 A. For this purpose, connect the *bMBrakeReleaseOut* trigger signal to the *bOutHC* input of the SB LS_DigitalOutput.

From version 15.00.00, the *bOutHC_BrakeApplied* status signal of the SB LS_DigitalInput can be used for a simple monitoring of the switching status of a holding brake connected to the high current output. ▸ Switching status of the motor holding brake at the high current output (📖 421)

9.13.2 Parameter setting

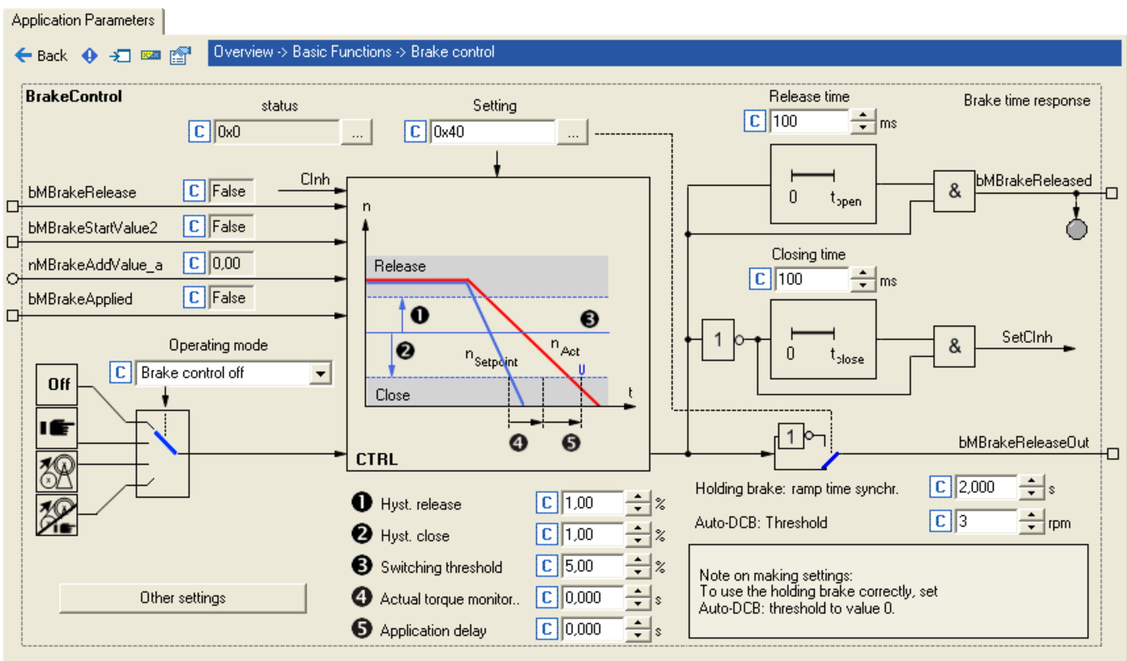
Danger!

A faultless holding brake control function requires a correct setting of the different deceleration times in the following parameters!

A wrong setting of the delay times can cause a faulty control of the brake!

How to go to the parameterisation dialog of the holding brake control:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the "**Basic functions**" button.
4. Go to the *Overview* → *Basic functions* dialog box and click the **Holding brake control** button.



Short overview of parameters for holding brake control:

Parameters	Info	Lenze setting	
		Value	Unit
C02580	Holding brake: Operating mode	0: Brake control off	
C02581/1	Holding brake: Switching threshold	5.00	%
C02581/2	Holding brake: Hyst. release	1.00	%
C02581/3	Holding brake: Hyst. close	1.00	%
C02581/4	Holding brake: FF control starting value 1	0	%
C02581/5	Holding brake: FF control starting value 2	0	%
C02582	Holding brake: Setting	0	
Greyed out = display parameter			

Parameters	Info	Lenze setting	
		Value	Unit
C02589/1	Holding brake: Closing time	100	ms
C02589/2	Holding brake: Release time	100	ms
C02589/3	Holding brake: Waiting time status	100	ms
C02589/4	Holding brake: Ramp FF control	0	ms
C02593/1	Holding brake: Actual value monitoring	0.000	ms
C02593/2	Holding brake: Application delay	0.000	ms
C02610/1	MCK: Holding brake ramp time synchr.	2.000	s
C02607	Holding brake: Status	-	
C00830/68	MCK: nMBrakeAddValue_a	-	%
C00833/80	MCK: bMBrakeRelease	-	
C00833/81	MCK: bMBrakeStartValue2	-	
C00833/82	MCK: bMBrakeApplied	-	
Greyed out = display parameter			

9.13.2.1 Operating mode

For different applications and tasks, different operating modes are available in C02580. The selected operating mode determines whether the holding brake control is used and how the holding brake will be switched.

Mode 0: Brake control off

In this mode, brake control is switched off (not active).

- The *bMBrakeReleaseOut* trigger signal for the holding brake control switching element is set to FALSE.
- The *bMBrakeReleased* status signal is set to FALSE.

Note!

In the Lenze setting, the mode 0 is preset to get into a safe state after the mains is switched on.

Mode 11: Manual control

In this mode, brake release and brake application can be directly controlled via the *bMBrakeRelease* input without special logic or automatic.

- Setting pulse inhibit or controller inhibit has no influence on the *bMBrakeReleaseOut* trigger signal for the holding brake control switching element.
- After the brake has been activated and the brake application time has expired, the controller is inhibited automatically by the basic "Holding brake control" function.

Tip!

You can use mode 11 to easily check if the brake switches correctly.

Mode 12: Automatic control

In this mode, the brake is controlled automatically.

> **Danger!**
>
> In this mode, the input *bMBrakeRelease* should be permanently set to FALSE unless manual release (supervisor operation) is required.
>
> If the *bMBrakeRelease* input is set to TRUE, the brake is released immediately, even if the controller is inhibited!

- If the requested speed setpoint reaches a parameterisable upper speed threshold that allows traversing of the drive, the brake will be released and operation enabled.
- On the other hand, if speed setpoint and actual speed fall below a parameterisable lower speed threshold, the brake will be applied under consideration of different time parameters.
- For operating modes with setpoint request via control signal (e.g. "PosExecute" in the Positioning" operating mode), the speed thresholds do not apply. Here the control logic opens and closes the holding brake through internal commands in the **Motion Control Kernel**.
- The brake will also be activated automatically if quick stop is activated in the drive, e.g. by a device command or as response to an error, and in the event of controller inhibit or pulse inhibit.
- After automatic brake activation and expiration of the brake application time, the controller is inhibited automatically by the basic "Holding brake control" function.

> **Tip!**
>
> The 2/12 mode is the usual mode to control the brake.

Mode 13: Semi-automatic control

In this mode, brake release and brake application can be directly controlled via the *bMBrakeRelease* input without special logic or automatic.

In contrast to the manual operation (mode 11)

- the feedforward control is active in this mode, preventing a sagging e.g. in case of a hoist.
- the brake in this mode also closes when the controller is inhibited in order to prevent the axis in a hoist from falling.

Related topics:

▶ TroubleQSP (📖 127)
▶ Behaviour in case of pulse inhibit (📖 716)

9.13.2.2 Functional settings

The following bit coded functional settings for the holding brake control can be made in C02582:

Bit	Option	Info
Bit 0	bMBrakeReleaseOut invert.	Activation of inverted control • "1" ≡ Inverted logic of the trigger signal for the holding brake control switching element
Bit 1	Horizontal brake protection	Brake response in case of pulse inhibit • "1" ≡ In the case of a pulse inhibit, the actual speed value is monitored which must reach the "Close" threshold value to cause the holding brake to be applied. **Note:** • This function is only active if bit 3 (horizontal/winding technology) is set as well. The function is used in order that, when the controller is inhibited, the holding brake of a drive with horizontal traverse path does not wear out during rotation. • With vertical motion (bit 3 = 0), this function is not active. Especially with hoists and activated pulse inhibit of the inverter, an immediate application of the brake is essential for safety-related reasons!
Bit 2	with hoist inv. feedfwd. control	Direction of feedforward control with vertical/hoist technology: • "0" ≡ Positive direction • "1" ≡ Negative direction **Note:** Reversal (Ccw) is then considered.
Bit 3	Horizontal application	Direction of movement of the axis • "0" ≡ The axis performs vertical movements. Gravitational acceleration causes movements. • "1" ≡ The direction of the axis is horizontal or rotary. The gravitational acceleration does not cause any movement. If C002868/1: Bit 7 is set to 1, the holding brake is released without injecting a feedforward control torque or frequency.

Bit 4	Feedforward control C2581	Selection of the feedforward control value
		• "0" ≡ Automatic selection.
		  • The torque saved at the last stop is used.
		• "1" ≡ Manual selection.
		  • *bMBrakeStartValue2* = FALSE: The feedforward control value 1 set in C02581/4 is used.
		  • *bMBrakeStartValue2* = TRUE: The feedforward control value set in C02581/5 is used.

Bit 5	Feedback monitoring	Activation of status monitoring • "1" ≡ The *bMBrakeApplied* input for status detection of the brake (via a switching contact at the brake) is monitored after the waiting time set in C02589/3 has expired.
Bit 6	Sync ramp L_NSet_1 (from version 02.00.00)	Selection of the ramp time for the synchronisation process to setpoint speed after the brake opening time has elapsed **Revised behaviour from version 02.00.00:** • "1" ≡ The ramp time of the effective acceleration of the ramp function generator (L_NSet_1) is used (Lenze setting). • "0" ≡ As before, the ramp time set in C02610/1 is used. **Note:** The changeover can be dynamically both via the ramp parameter and via bit 6.
Bit 7	Reserved	

Related topics:

- Behaviour in case of pulse inhibit (📖 716)
- Feedforward control of the motor before release (📖 717)

9.13.2.3 Switching thresholds

Stop!

Do not set the lower speed threshold for closing the brake too high to prevent excessive wear of the brake!

Note!

When comparing speeds, only the absolute value of the motor speed and not the direction of rotation is considered.

Upper speed threshold for brake release:

Switching threshold (C02581/1) + hysteresis for release (C02581/2)

Lower speed threshold for brake application:

Switching threshold (C02581/1) - hysteresis for application (C02581/3)

Tip!

The lower speed threshold for brake application should be set to approximately 5 ... 20 % of the maximum speed to minimise the wear of the brake and provide for an optimum brake reaction by a low grinding of the brake.

Related topics:

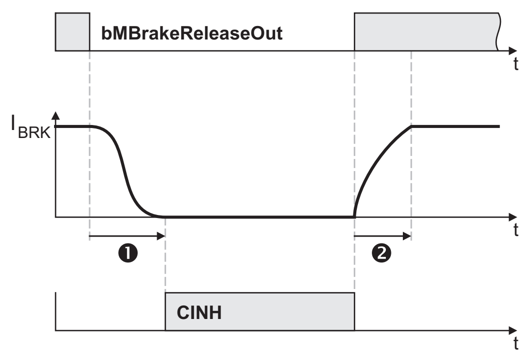
- Process when brake is released (📖 713)
- Process when brake is closed (📖 714)

9.13.2.4 Application and release time

Danger!

A wrong setting of the application and release time can cause a faulty control of the brake!

- If the application time is set too low, the controller is inhibited and the drive becomes torqueless before the brake is applied completely.



❶ C02589/1: Application time
❷ C02589/2: Release time
CINH = controller inhibit

- Every mechanical holding brake comes with a construction-conditioned application and release time which must be considered by the holding brake control and is set in C02589.
- The application and release time of the Lenze holding brake is indicated in the supplied operating instructions in the "Technical data" chapter.
- If the application and release times are too long, this is uncritical in respect of safety but leads to unnecessarily long delays during cyclical braking processes.

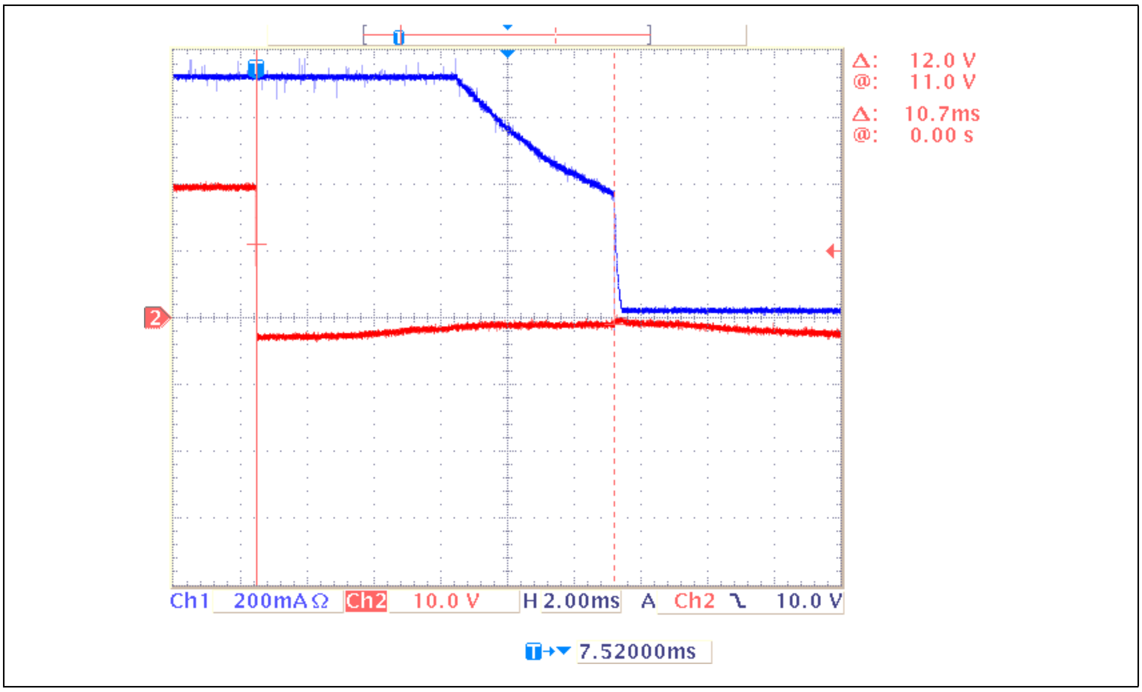
[9-39] Definition of the application and release time with the example of the PM brake

Tip!

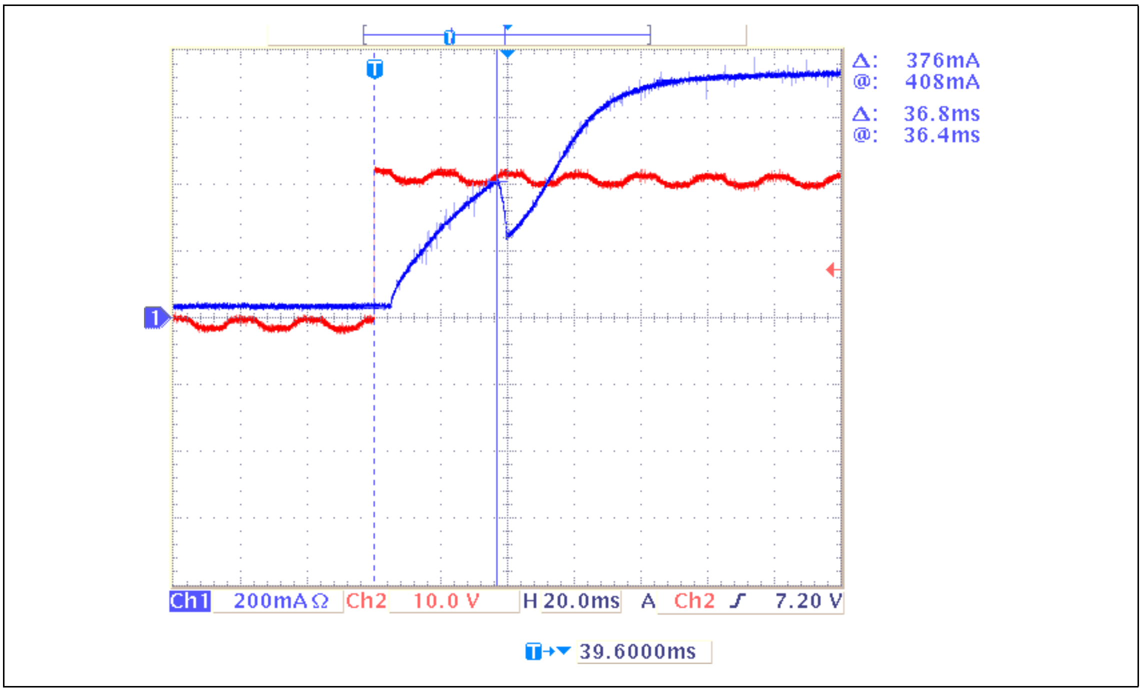
The application and release times do not only vary between the brake types but also depend on the basic conditions in the plant:

- Parameters of the hardware (cable length, temperature, level of supply voltage etc.)
- Contact elements used (brake module or contactor at the digital output)
- Type of overvoltage limitation/suppressor circuit

For optimisation purposes, detect in individual cases the response times by measurement.



[9-40] Oscillogram 1: Current characteristic for the application of a mechanical holding brake (application time: 10.7 ms)



[9-41] Oscillogram 2: Current characteristic for the release of a mechanical holding brake (release time: 36.8 ms)

Related topics:

▸ Process when brake is released (📖 713)
▸ Process when brake is closed (📖 714)

9.13.2.5 Ramp time for approaching the setpoint speed

For the "Speed follower" operating mode, a ramp time can be set in C02610/1 if the setpoint is already out of reach while the holding brake is initiating the feedforward control process.

Note!

The "Homing", "Manual jog", and "Positioning" operating modes are based on a different control/release process of the holding brake. In these operating modes, the C02610/1 setting parameter does not have any effect!

Example:

A setpoint of 90 % is selected via the ramp function generator while the brake is applied (controller is inhibited).

1. At the set ramp (in most cases C00012), the ramp function generator ramps up to 90 %.
2. The brake identifies the setpoint selection of 5 % (release switching threshold). The feedforward control of the brake provides 3 % of the setpoint and will not report the release of the brake after approx. 1 s has expired.

Conclusion: 90 % of the selected setpoint is already ramped up while the brake is only providing 3 % of the setpoint via the feedforward control.

Since at this point a step change from 3 % to 90 % may cause mechanical jerks, the setpoint is ramped up from 3 % to 90 %, using the ramp time set in C02610/1 (Lenze setting: 2 s).

Our example is based on the V/f characteristic control (VFCplus) operating mode since the servo control (SC) operating mode does not use the switching threshold for the application of the holding brake for speed feedforward control. However, ramping up to a setpoint that is out of reach is carried out for all motor control operating modes because there is always a mechanical/electric delay when controlling a holding brake.

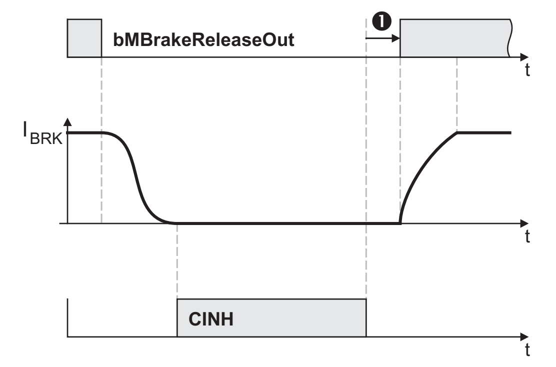
This delay is due to:

- Motor magnetisation (in the case of servo control only)
- Mechanical delay of all switching elements connected upstream to the holding brake
- Mechanical delay of the holding brake itself
- Generation of the holding torque by the motor

Related topics:

▸ Process when brake is released (📖 713)

9.13.2.6 Motor magnetising time (only with asynchronous motor)



❶ Motor magnetising time
CINH = controller inhibit

[9-42] Considering the motor magnetising time taking the PM brake as an example

- When an asynchronous motor is used, first the magnetic field required for the holding torque is created (which is already available when a synchronous motor is used) after the controller inhibit is deactivated.
- The motor is internally magnetised through internal feedforward control of the lower speed threshold. The release time set in C02589/2 is considered here.

Related topics:

▸ Process when brake is released (📖 713)

9.13.2.7 Actual value monitoring

If an actual value monitoring time > 0 s is selected in C02593/1, the actual speed time monitoring is active.

- The monitoring time starts when the speed setpoint has reached the lower switching threshold and the actual speed is still above this threshold. (see illustration [9-45] in chapter "Process when brake is closed".)
- If the actual speed is still above the threshold when the monitoring time has expired, the brake will be automatically applied in the automatic brake control mode (mode 12).

Note!

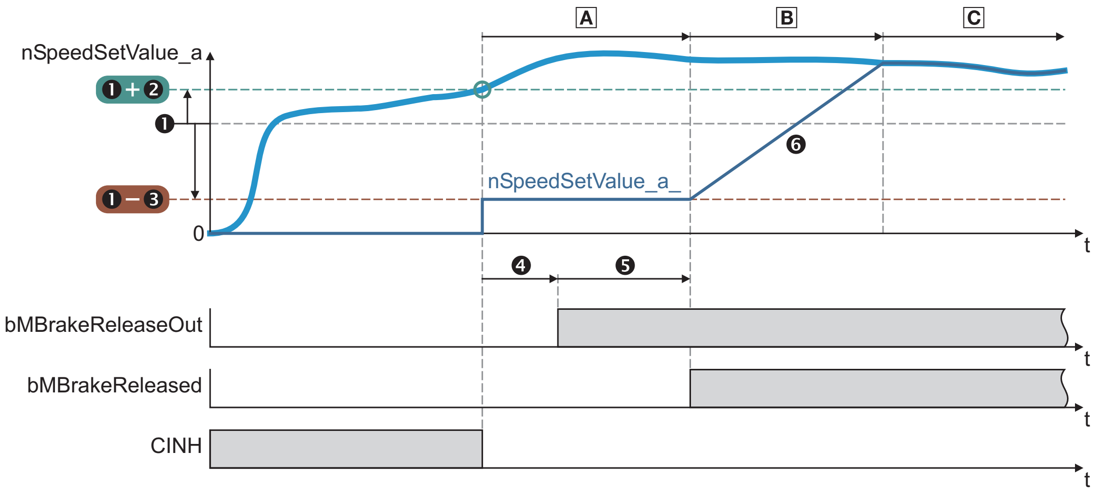
In the Lenze setting, the actual speed time monitoring is deactivated (C02593/1 = "0 s"), i.e. the brake will only be applied when the actual speed has reached the lower switching threshold.

9.13.3 Process when brake is released

1. The controller inhibit is deactivated.
2. The magnetic field required for the holding torque is created in the motor (is already available when a synchronous machine is used).
3. The *bMBrakeReleaseOut* trigger signal for holding brake switching element is set to TRUE for releasing the brake.
4. After the brake opening time has elapsed:
   - The *bMBrakeReleased* status signal ("brake released") is set to TRUE.
   - In the "Speed follower" operating mode, the drive synchronises to the already accelerated speed setpoint.
   - In the operating modes with setpoint request via control signal (operating modes "Homing", "Manual jog" and "Positioning"), the ramping process starts after the brake release at 0.
5. After the waiting time set in C02589/3 has additionally expired, the status monitoring starts again (if activated via bit 5 in C02582).

Time diagram

A Feedforward control
B Setpoint synchronisation
C Follow setpoint



❶ C02581/1: Switching threshold
❷ C02581/2: Hysteresis for release
❸ C02581/3: Hysteresis for application
❹ Time slot for feedforward control/magnetising
❺ C02589/2: Release time
❻ C02610/1: Ramp time for approaching the setpoint speed

[9-43] Release holding brake in automatic mode via speed threshold

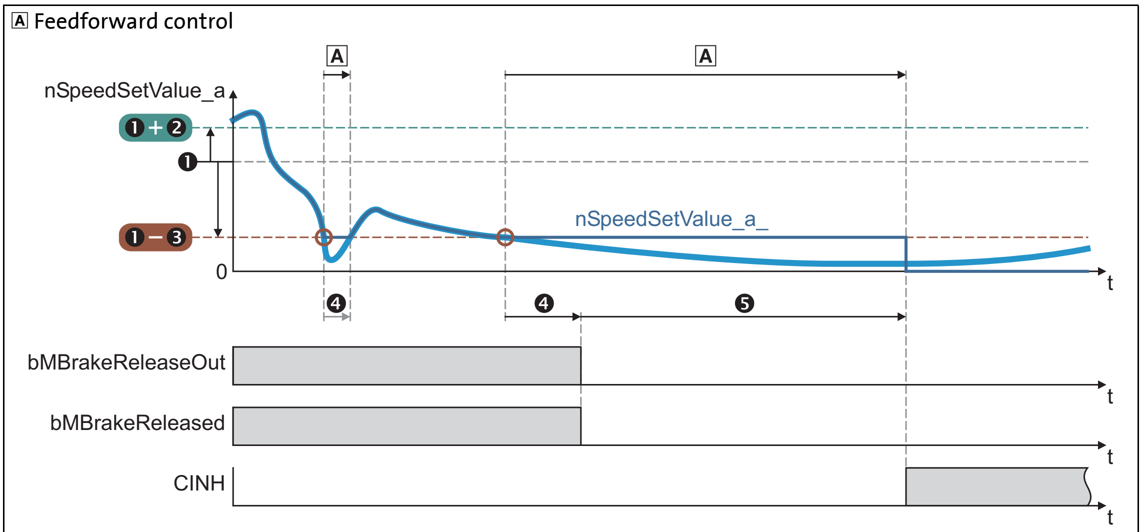
Related topics:

▶ Feedforward control of the motor before release (📖 717)

9.13.4 Process when brake is closed

1. The motor is decelerated when the setpoint is reduced by the user (e.g. turn down the potentiometer, setpoint selection via CAN).
   - The motor can also be decelerated by the "Quick stop" function or by "DC-injection braking", either directly requested by the user or as response to an error.
2. If the speed setpoint and the actual speed have fallen below the lower speed threshold or only the speed setpoint has fallen below the lower speed threshold and the actual value monitoring time has expired:
   - The *bMBrakeReleaseOut* trigger signal for the holding brake switching element is set to FALSE for closing the brake.
   - The *bMBrakeReleased* status signal is reset to FALSE.
   - In the operating modes with setpoint request via control signal (operating modes "Homing", "Manual jog" and "Positioning"), the brake closes depending on the internal state *bBusy* (setpoint generation through active profile generator) of the **Motion Control Kernel**.
   - The brake application time starts to expire.
3. After the brake application time has expired, the controller is inhibited.
4. After the waiting time set in C02589/3 has additionally expired, the status monitoring starts again (if activated via bit 5 in C02582).
5. In order to prevent the drive from further rotating/accelerating in the event of an error of the feedback contact, controller inhibit is cancelled again and the drive is held at standstill in a speed-controlled manner.

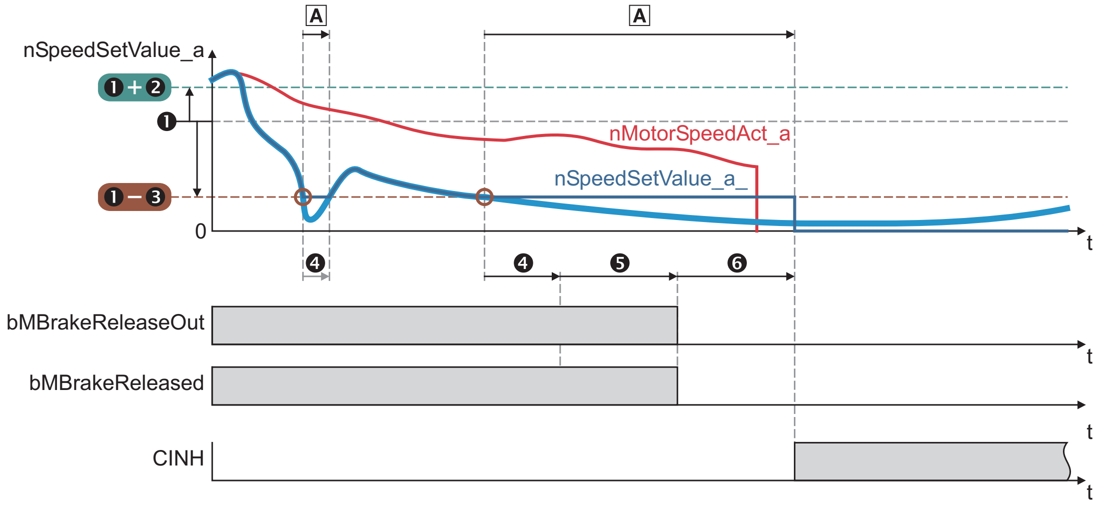
Time diagrams



❶ C02581/1: Switching threshold
❷ C02581/2: Hysteresis for release
❸ C02581/3: Hysteresis for application
❹ C02593/2: Application delay
❺ C02589/1: Application time

[9-44] Close holding brake in automatic mode via speed threshold (actual value = setpoint)

A Feedforward control



❶ C02581/1: Switching threshold
❷ C02581/2: Hysteresis for release
❸ C02581/3: Hysteresis for application
❹ C02593/2: Application delay
❺ C02593/1: Actual value monitoring time
❻ C02589/1: Application time

[9-45] Close holding brake in automatic mode with actual value monitoring time (C02593/1 > 0 s)

9.13.5 Behaviour in case of pulse inhibit

Setting the pulse inhibit causes a load-controlled coasting of the motor until the pulse is enabled again. In the enabled inverter, the pulse can be inhibited e.g. due to a DC overvoltage, DC undervoltage or the "Safe torque off" request.

The brake response to pulse inhibit can be parameterised under C02582.

Stop!

For parameterising the response to pulse inhibit in C02582, the energy conditions of the machine should be evaluated first.

The energy stored in the machine can be considerably higher than the permissible switching energy and thus lead to the destruction of the brake if applied directly!

Activate brake immediately when pulse is inhibited

If bit 1 is set to "0" in C02582 (Lenze setting), the brake will be immediately applied when the pulse is inhibited to avoid damage to the mechanical components.

Especially in the case of hoist drives, immediate engagement of the brake is absolutely necessary for safety reasons if the pulse inhibit function of the inverter has been activated!

Danger!

This behaviour is valid in (semi) automatic operation when the *bMBrakeRelease* input is set to FALSE.

When the *bMBrakeRelease* input is set to TRUE (supervisor operation) in automatic mode, the brake is not applied at pulse inhibit!

Only activate brake below threshold for brake activation

If bit 1 and bit 3 are set to "1" in C02582, the brake remains released until the lower speed threshold is reached to avoid an excessive wear of the brake.

- The braking action only takes places due to the friction in the load mechanics.
- The brake will not be applied until the motor speed has reached the threshold for brake activation. Hence, the function depends on the signal of the speed encoder.

During uncritical operation (horizontal loading condition), delayed brake application may be required to protect the brake in case of high centrifugal masses.

In case of vertical motion (bit 3 = 0), this function is not active due to safety-related reasons.

Related topics:

- Functional settings (707)
- Switching thresholds (708)

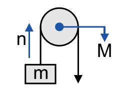
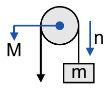
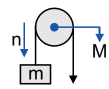
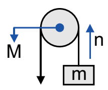
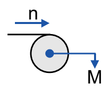
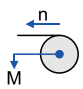
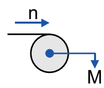
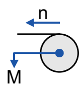
9.13.6 Feedforward control of the motor before release

The motor is precontrolled by selecting the lower speed threshold for applying the brake (control modes without feedback). When the upper speed threshold for brake release is reached, the motor is precontrolled with the for the release time set in C02589 with the lower threshold value before the brake switches to the release mode.

Here, the direction of the feedforward control depends on two conditions:

1. On the settings selected under C02582:
   - Bit 2 = feedforward control inverted (for vertical drives/hoists)
   - Bit 3 = direction of the axis
2. On the sign of the setpoint.

Truth table for the direction of the feedforward control

Setpoint	Direction	Feedforward control	Scheme	Direction	
				Feedforward control value	Start value
n ≥ 0	vertical/hoist (C02582: Bit 3 = 0)	Not inverted (C02582: Bit 2 = 0)		+	+
		Inverted (C02582: Bit 2 = 1)		−	+
n < 0		Not inverted (C02582: Bit 2 = 0)		+	−
		Inverted (C02582: Bit 2 = 1)		−	−
n ≥ 0	horizontal/winding drive (C02582: Bit 3 = 1)	Inversion via bit 2 with horizontal direction not effective		+	+
n < 0				−	−
n ≥ 0	horizontal/winding drive (C02582: Bit 3 = 1)	Release holding brake without injecting feedforward control torque or frequency C002868/1: Bit 7 = 1		0	+
n < 0				0	−

Note!

The effective direction of the torque when the drive is stopped also determines the direction of the feedforward control value. In case of emergency stop scenarios or mechanical blocking torques can be stored when the drive is stopped that act against the desired feedforward control direction. If such scenarios are not excluded in applications, you should not use the automatic storage of the holding torque.

- Alternatively use the manual selection (C02582: bit 4 = 1).

Selection of the feedforward control value

The feedforward control value can be selected via bit 4 in C02582:

- Bit 4 = 0: Automatic selection
  - The torque saved at the last stop is used.
- Bit 4 = 1: Manual selection
  - *bMBrakeStartValue2* = FALSE: The feedforward control value 1 set in C02581/4 is used.
  - *bMBrakeStartValue2* = TRUE: The feedforward control value set in C02581/5 is used.

Additive torque

If the servo control (SC) operating mode has been selected, an additive torque value in [%] can be selected via the *nBrkTorqueAdd_a* input.

Related topics:

▶ Functional settings (📖 707)
▶ Switching thresholds (📖 708)

10 Diagnostics & error management

This chapter provides information on error handling, drive diagnostics, and fault analysis.

10.1 Basics on error handling in the inverter

Many of the functions integrated in the inverter can

- detect errors and thus protect the device from damage or overload, e.g. short-circuit detection, Ixt overload detection, overtemperature detection, etc.
- detect an operating error by the user, e.g. a missing memory module, a required or missing communication module, etc.
- output a warning signal if desired, e.g. if the speed is too high or too low, etc.

Depending on the importance, the error detection in the device responds very fast (e.g. short-circuit detection < 1 ms) or in a slower cycle (e.g. temperature monitoring approx. 100 ms).

All functions provided with an error detection (e.g. the motor control) supply information to a so-called error handler. The error handler is processed every 1 ms and evaluates all information.

In this evaluation, the so-called status determining error (display in C00168) and the current error (display in C00170) are generated, and the inverter is caused to take the respective error status (e.g. TroubleQSP).

These two types of error information serve to diagnose errors systematically and contain the following information:

1. The error type (e.g. "Warning")
2. The error subject area (e.g. "CAN generally integrated")
3. The error ID within the error subject area

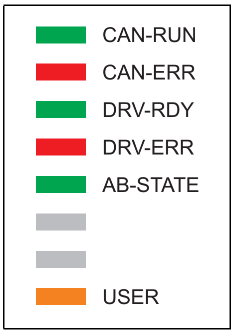
Together all types of information form the real error number which is unique in the whole device system. ▶ Structure of the 32-bit error number (bit coding) (📖 743)

In addition to the control of the device status by the error handler, a logbook function records the errors and their histories. ▶ Logbook (📖 728)

Tip!

For many device errors, the error type and hence the response of the inverter to the error can be parameterised. ▶ Setting the error response (📖 738)

10.2 LED status displays



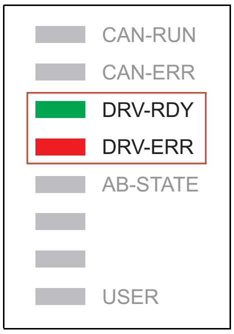
Information on some of the operating states can be quickly obtained via the LED display on the front of the inverter.

Labelling	Colour	Description	
CAN-RUN	green	CAN bus ok	▶ LED status displays for the system bus (📖 810)
CAN-ERR	red	CAN bus error	
DRV-RDY	green	Standard device ready for operation	▶ LED status displays of the device status (📖 721)
DRV-ERR	red	Warning/trouble/fault	
AB-STATE	green	Axis bus status	▶ LED status displays of the axis bus (📖 892)
-	-	These LEDs are currently not provided with a function!	
-	-		
USER	orange	Starting from version 12.00.00, this LED can be triggered with any digital process signal via the LS_DigitalOutput system block. (📖 424)	

Related topics:

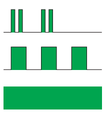
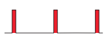
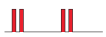
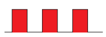
- ▶ Device control (DCTRL) (📖 105)
- ▶ Device state machine and device states (📖 119)
- ▶ System bus "CAN on board" (📖 804)

10.2.1 LED status displays of the device status

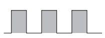


The control of the two LEDs "DRV-RDY" and "DRV-ERR" on the front of the inverter depends on the device status.

The meaning can be seen from the table below.

DRV-RDY	DRV-ERR	Description	Device status (Display in C00137)
OFF	OFF	OFF or initialisation active	Init
	OFF	Safe torque off is active	SafeTorqueOff
	OFF	Device is ready to start	ReadyToSwitchOn
	OFF	Device is switched on	SwitchedOn
	OFF	Motor data identification/operation	OperationEnabled
		The inverter is ready to switch on, switched on or the operation is enabled and a warning is indicated.	
		Fault active, quick stop	TroubleQSP
OFF		Trouble active	Trouble
OFF		Error active	Fault

Legend
Meaning of the symbols used to describe the LED states:

	LED is flashing once approx. every 3 seconds (*slow flash*)
	LED is flashing once approx. every 1.25 seconds (*flash*)
	LED is flashing twice approx. every 1.25 seconds (*double flash*)
	LED is blinking every second
	LED is permanently on

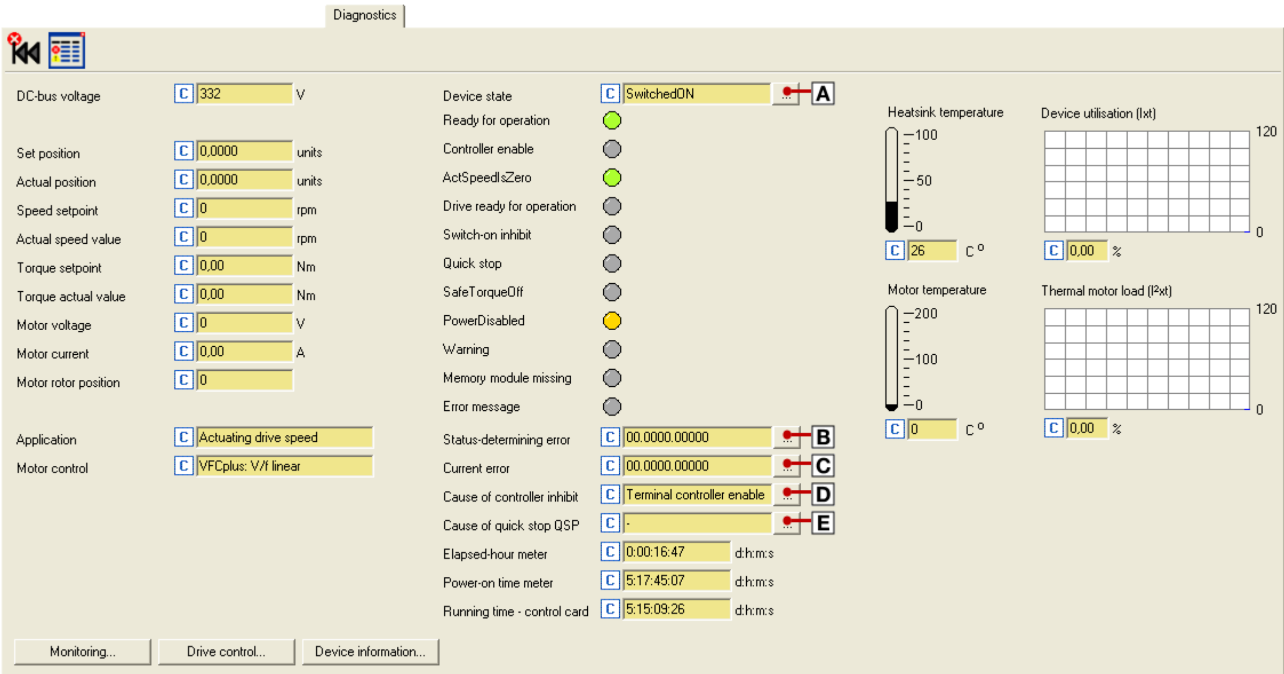
Related topics:

▶ Device state machine and device states (📖 119)

10.3 Drive diagnostics with the »Engineer«

When an online connection to the inverter has been established, the connected inverter can be diagnosed and relevant actual inverter states can be displayed in a clearly arranged visualisation using the »Engineer«.

Example: Representation of the diagnostic information in the »Engineer« V2.19



Button		Function
		Acknowledge error message (if the error cause has been eliminated).
		Display the Logbook of the inverter. (📖 728)
...	A	Display the internal state machine including the current device status.
	B	Display details of the status determining error.
	C	Display details of the current error.
	D	Display all active sources of a controller inhibit.
	E	Display all active sources of a quick stop.
Monitoring...		Configure the Monitoring. (📖 736)
Drive control...		Display the bit assignment of the following control-related words: • MCI control word (C00136/1) • CAN control word (C00136/2) • Cause of controller inhibit (C00158) • Cause of quick stop (C00159) • Status word (C00150) • Status word 2 (C00155)
Device information...		Display identification data, e.g. firmware information or serial number of individual inverter components.

How to diagnose a drive with the »Engineer«:

1. Go to the *Project view* and select the 8400 TopLine inverter.
2. Click the icon or select the **Online→Go online** command to build up an online connection with the inverter.
3. Select the **Diagnostics** tab.
   - With an online connection, the **Diagnostics** tab displays current status information about the inverter.

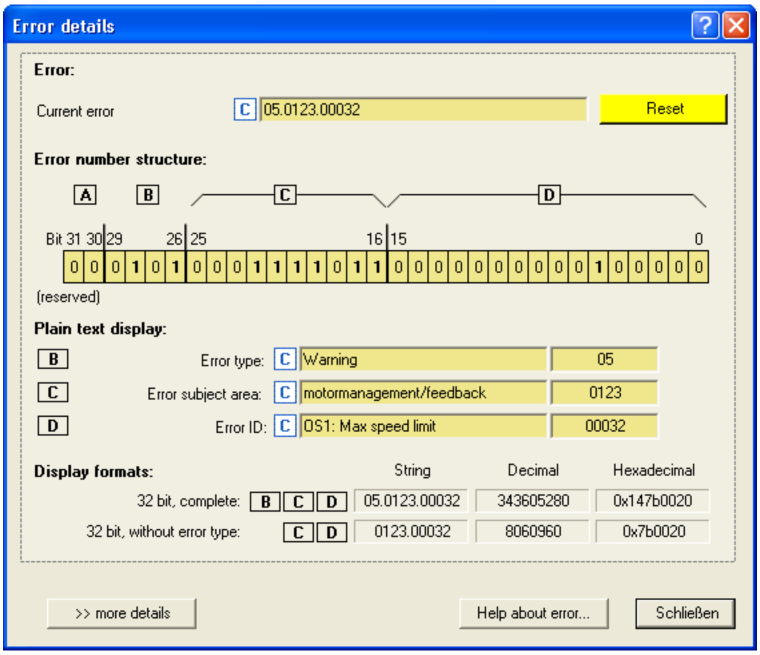
Tip!

The online connection to the inverter can be established via the following device interfaces:

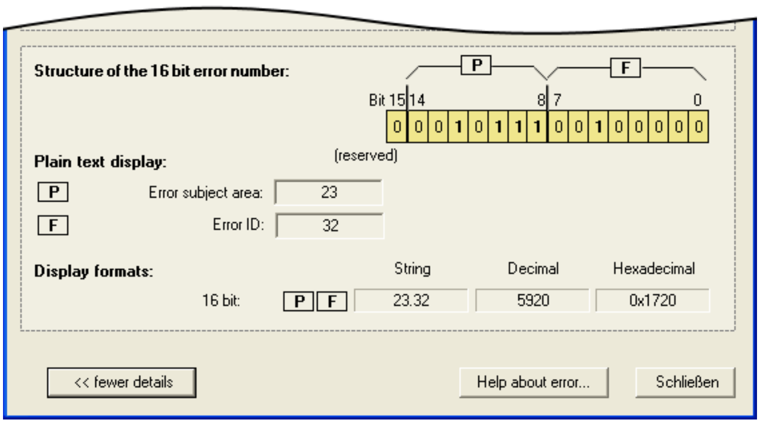
- CAN interface X1
  Diagnostics via the System bus "CAN on board" (804)
- X6 diagnostic interface
  We recommend this diagnostic connection when the X1 CAN interface is used for process communication.

10.3.1 Display details of the error

If you go to the **Diagnostics** tab and click the ... button for the status determining or current error, the *Error details* dialog box displays further information on the error:



- Click the **Help about error...** button to open the online help with information on the error cause and possible remedies.
- The **>> more details** button serves to provide more information about the structure of the 16-bit error number:

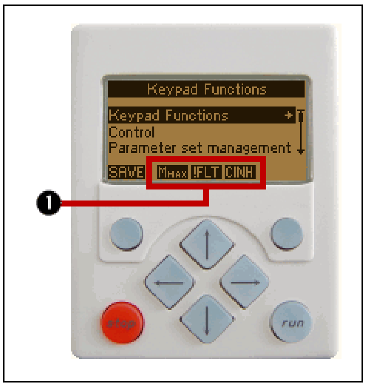


Related topics:

▸ Structure of the 32-bit error number (bit coding) (📖 743)
▸ Structure of the 16 bit error number (bit coding) (📖 746)

10.4 Drive diagnostics via keypad/bus system

Keypad display of the inverter status



- If the keypad on the front of the inverter is connected to the diagnostic interface X6, the status of the inverter is shown via different icons on the LCD display in the area ❶.

Symbol	Meaning	Notes
RDY	Inverter is switched on.	▸ SwitchedOn (🕮 125)
RUN	Inverter is enabled.	
QSP	Quick stop active	
CINH	Inverter is inhibited.	The power outputs are inhibited.
Mmax	Speed controller 1 at the limit.	The drive is torque-controlled.
Imax	Set current limit has been exceeded in motor or generator mode.	
IMP	Pulse inhibit active	The power outputs are inhibited.
!FLT	Fault	▸ Fault (🕮 129)
!TRB	Trouble	▸ Trouble (🕮 128)
!TQSP	TroubleQSP	▸ TroubleQSP (🕮 127)
WRN	Warning is active	

Display parameters

The parameters listed in the following tables serve to query current states and actual values of the inverter for diagnostic purposes, e.g. by using the keypad, a bus system or the »Engineer« (with an online connection to the inverter).

- These parameters are listed in the »Engineer« parameter list and the keypad in the **Diagnostics** category.
- A detailed description of these parameters can be found in the chapter "Parameter reference" (928).

Parameters	Display
C00051	MCTRL: Actual speed value
C00052	Motor voltage
C00053	DC-bus voltage
C00054	Motor current
C00056/1	Torque demand
C00056/2	Actual torque value
C00058	Output frequency
C00061	Heatsink temperature
C00064/1	Device utilisation (Ixt)
C00064/2	Device utilisation (Ixt) 15s
C00064/3	Device utilisation (Ixt) 3 min
C00133	Brake resistor utilisation
C00136/1	MCI control word
C00136/2	CAN control word
C00137	Device status
C00138/1	SYS control signals
C00138/2	MCK control signals
C00138/3	FWM control signals
C00150	Status word
C00158	Cause of controller inhibit
C00159	Cause of quick stop QSP
C00165/1	Status determining error (displayed as a numeric text)
C00165/2	Current error (displayed as a numeric text)
C00168	Status determining error (display of 32-bit number)
C00170	Current error
C00166/1	Error type, status determining
C00166/2	Error subject area, status determining
C00166/3	Error ID, status determining
C00166/4	Error type, current
C00166/5	Error subject area, current
C00166/6	Error ID, current
C00177/1	Switching cycles mains switching
C00177/2	Switching cycles output relay
C00177/3	Stress counter - short circuit
C00177/4	Stress counter - earth fault
C00177/5	Stress meter clamp

Parameters	Display
C00177/6	STO counter after power-on
C00177/7	DigIn CINH counter after power-on
C00177/8	IMP counter after power-on
C00178	Time the inverter was enabled (elapsed-hour meter)
C00179	Power-up time (power-on time meter)
C00180/1	Runtime - control card
C00180/2	Running time - heatsink fan
C00180/3	Running time - internal fan

Identification data

The parameters listed in the following table belong to the **Identification** category of the »Engineer« parameter list and the keypad and serve to display the identification data of the inverter:

Parameters	Display
C00099	Firmware version (as a string)
C00199/1	Device name ▶ Automatic acceptance of the device name in the »Engineer«
C00200	Firmware product type
C00201/1...6	Firmware of the control card and the power section
C00203/1...9	Product type code of the individual device components
C00204/1...9	Serial numbers of the individual device components

10.5 Logbook

The integrated logbook function of the inverter chronologically logs important events within the system and plays an important role for troubleshooting and inverter diagnostics.

Events that can be logged

The following events can be logged in the logbook:

- Error messages of the operating system (📖 743)
- Error messages generated by the application (via LS_SetError)
- Loading/saving of parameter sets, loading of the Lenze setting *(in preparation)*
- Transmitting the firmware to the inverter *(in preparation)*
- Switching on/off of the inverter

Information saved

For each event, the following information is saved in the logbook:

- Type of response to the event (e.g. fault, warning or information)
- Subject area that activated the event (e.g. CAN or USER).
- Event
- Value of power-on time meter
- Selected process values (analog % signals, binary signals)

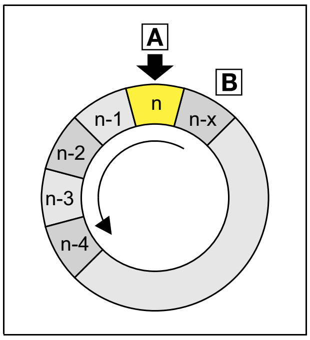
Memory depth

Maximum number of logbook entries:

- 8400 StateLine: 15 logbook entries
- 8400 HighLine/TopLine: 50 logbook entries

10.5.1 Functional description

The structure of the logbook corresponds to a ring buffer:



- As long as free logbook memory is available, the entry is placed in the next free position within the memory (A).
- If all memory units are assigned, the oldest entry (B) is deleted for a new entry.
- The newest entries will always remain available.

Note!

In the event of a supply voltage failure, the logbook is saved and reloaded automatically when the inverter is switched on. This ensures that the error history of the device does not get lost. For this reason it is very important to act with caution when deleting the logbook entries.

10.5.2 Filtering logbook entries

The logbook adds new entries to the ring buffer after they have been passed through a parameterisable filter. This filter helps you to exclude certain events from being entered into the logbook which would trigger a certain error response (fault, trouble, warning, information, etc.).

C00169 (bit 1 ... bit 6) includes a bit coded specification of the events which are to be entered into the logbook. In the Lenze setting, all events are entered into the logbook.

In the Lenze setting (C00169 = 0x067E), all events are entered into the logbook.

Note!

Events with the "No response" setting are not entered into the logbook.

Counter for multiple entries

In order to prevent the ring buffer from overflowing with identical errors with frequent occurrence e.g. during commissioning, identical errors will not lead to new line entries in the configuration of the logbook in the Lenze setting. Instead, one counter will be counted up for this error.

The time of the error is always the time of its first occurrence. A new logbook line will only be generated if an error occurs that differs from the previous error.

In C00169, bit 9 and bit 10 determine the logbook update in case of similar errors.

Meaning of bit 9 / bit 10:

- Bit 9: Error counter
- Bit 10: Update of the logbook line

In the Lenze setting (C00169 = 0x067E), bits 9 and 10 are active.

The following cases can be distinguished depending of the activation of bit 9 and bit 10:

1. Bit 9 inactive, bit 10 active/inactive
   - Every time an error occurs, a new line is created in the logbook.
   - The setting of bit 10 has no effect!
2. Bit 9 active, bit 10 inactive
   - Only one line is created for the first occurrence of the error. If the error occurs again, only the error counter in the line is increased.
3. Bit 9 active / bit 10 active (Lenze setting)
   - The time of the last occurrence of similar errors is logged.
   - The time stamp and the optional measured values are updated in the error line of the logbook.
   - When the count value = 255 is reached and an error occurs again, a new line is created.

10.5.3 Automatic recording of device-internal signals at the time the error occurs

For purposes of error analysis, two optional device-internal digital signals and an analog signal (16 bit) can be recorded at the time the error occurs.

- The digital signals to be recorded are selected in C00163/1 and C00163/2.
  - Scaling of the value recorded: 0 ≡ FALSE; 1 ≡ TRUE

- The analog signal to be recorded is selected in C00164/1.
  - Scaling of the value recorded: 16384 ≡ 100 %

10.5.4 Reading out logbook entries

With an online connection, the existing logbook entries can easily be displayed in the »Engineer«. Alternatively, the logbook entries can also be read out via the corresponding parameters (e.g. using the keypad).

Note!

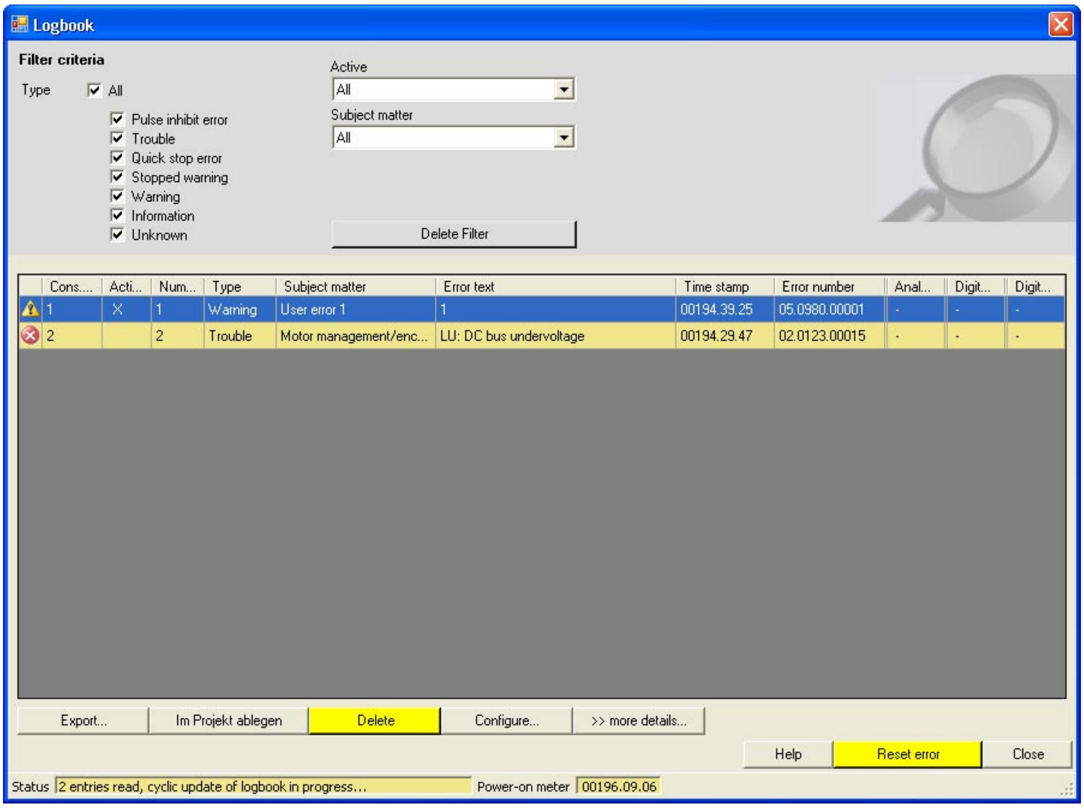
Reading out the logbook entries is only permissible by one engineering tool at a time!

- If the logbook entries are tried to be read out via two engineering tools at the same time, missing lines or a permanent refresh may be caused.
- The parameter interface for reading out the logbook by an external control/visualisation, is available as well. ▶ Reading out the logbook from an external control/visualisation (📖 734)

How to display logbook entries in the »Engineer«:

1. Go to the *Project view* and select the 8400 TopLine inverter.
2. Click the icon or select the **Online→Go online** command to build up an online connection with the inverter.
3. Select the **Diagnostics** tab from the *Workspace*.
4. Click **Logbook**.

Example: Representation of the logbook in the »Engineer« V2.13



Button	Function
Delete Filter	Reset set filter criteria to display all available logbook entries.
Export...	Export the entries available in the logbook into a *.log file. ▶ Exporting logbook entries to a file (📖 732)
Storing in the project	File the current logbook in the Engineer project to be able to access it offline, too. ▶ Storing the logbook in the project (📖 733)
Delete	Delete all entries available in the logbook.
Configuration of...	Open parameterisation dialog for configuring the logbook.
>> more details	Show more details: • Analog value 1, digital values 1 & 2 • More output types of the error numbers (32-bit, internal 32-bit and internal 16-bit). • Instead of the **>> more details** button, the **<< less details** is now displayed via which the details can be hidden again.
Help	Open online help for the logbook.
Error reset	Acknowledge existing error message if the error cause has been eliminated and thus the error is not pending anymore. • After the reset (acknowledgement) of the current error, further errors may be pending which must also be reset.
Close	Close the *Logbook* dialog box again.

10.5.5 Exporting logbook entries to a file

How to export the logbook entries to a file:

1. Click **Export...** in the *Logbook* dialog box.
   - The *Export logbook* dialog box is displayed.
2. Specify the folder, file name, and file type for the file.
3. Click the **Save** button to export the logbook entries into the given file.
   - Hidden logbook entries are not exported, i.e. the filter criteria specified are accounted for during the export.
   - The logbook entries are written to the file in the form of a semicolon separated list.

Structure of the semicolon separated list

The list includes the following information:

1. Cons. no.
2. Active
3. Counter
4. Type
5. Subject matter
6. Error text
7. Time stamp
8. Relative time
9. Error number
10. Source - analog value 1
11. Analog value 1
12. Source - digital value 1
13. Digital value 1
14. Source - digital value 2
15. Digital value 2

10.5.6 Storing the logbook in the project

If you want to display the currently available logbook entries at a later date in offline mode, i.e. without a connection to the inverter, you can store the current logbook in the project.

How to store the logbook in the project:

Go to the *Logbook* dialog box and click the **File in project** button.

- The logbook with all the entries uploaded up to now is stored in the Engineer project independent of the set filter criteria.
- A logbook of the same device already stored before will be overwritten without querying the user.
- The filter settings are not stored in the project.
- When a logbook is filed in the project, the logbook can also be opened in offline mode via the **Logbook** button on the **Diagnostics** tab.

Note!

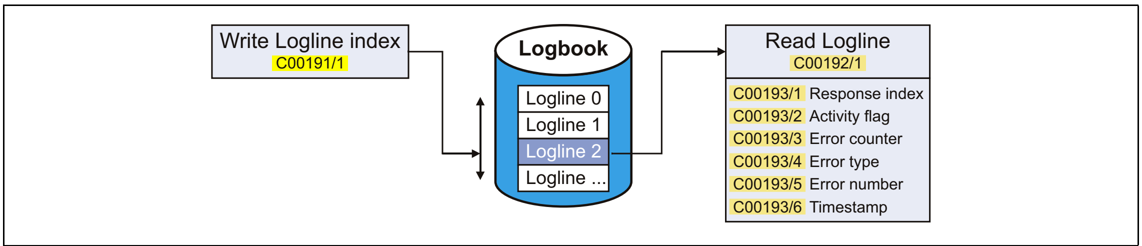
Storing the logbook changes the project.
- When the project is closed, you are asked to save the changed project.
- Only if the changed project is saved, the new logbook entries stored in the project remain stored.

10.5.7 Reading out the logbook from an external control/visualisation

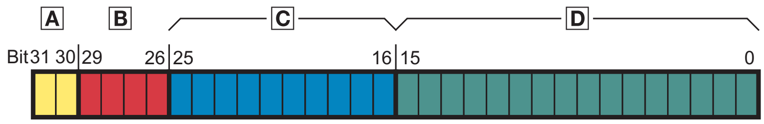
This function extension is available from version 12.00.00!

The parameters described in the following provide a second interface via which an external control or visualisation can exclusively access the logbook and read out logbook contents. Logbook contents can even be read out via this additional interface if the logbook is read out by the »Engineer« or »EASY Starter« at the same time.

Structure of the interface



- The line index of the logbook entry to be read out has to be set In C00191/1.
- Then the logbook entry addressed is stored as "OCTET STRING" in C00192/1.
  - Via this display parameter, the logbook entry can be read out by the control.
  - The "OCTET STRING" has a length of 8 double words of 4 successive bytes (= 32 bytes), starting with byte 0 in each case:

OCTET STRING	Byte 0	Byte 1	Byte 2	Byte 3
DWORD 0	Reserved	Reserved	Response index	Reserved
DWORD 1	Error activity flag	Error counter	Reserved	Reserved
DWORD 2	32-bit error number  A Reserved B Error type C Error subject area D Error ID			
DWORD 3	Time in [s] during which the power was switched on (power-on time meter).			
DWORD 4	Reserved	Reserved	Reserved	Reserved
DWORD 5	Source - analog value 1	Reserved	Analog value 1 (scaling: 16384 ≡ 100 %)	
DWORD 6	Source - digital value 1	Reserved	Digital value 1	Reserved
DWORD 7	Source - digital value 2	Reserved	Digital value 2	Reserved

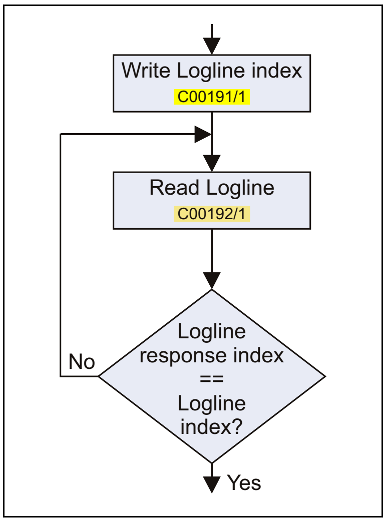
- Additionally, a read access to different individual elements of the logbook entry addressed can be executed via the subcodes of C00193. These parameters have a uniform data format (32 bits) and represent the most important part of the logbook data:

Parameters	Display
C00193/1	Response index, reference to the index requested (C00191/1)
C00193/2	Error activity flag (0 = error not active; 1 = error active)
C00193/3	Error counter (0 ... 255) • This information depends on the logbook configuration. In the Lenze setting the logbook configured so that identical errors do not produce a new line entry, but a counter is incremented for this error.
C00193/4	Error type (bits 26 ... 29 of the 32-bit error number)
C00193/5	Error subject area + error ID (bits 0 ... 25 of the 32-bit error number)
C00193/6	Time in [s] during which the power was switched on (power-on time meter).

Note!

- In the case of a simultaneous read access to the logbook entry (C00192/1) and its individual elements (C00193/x), the line index in C00191/1 must only be reset by the control after the read process is fully completed. Otherwise the data read are inconsistent.
- Depending on the device version, the maximum number of logbook entries can vary:
  - 8400 StateLine: 15 logbook entries
  - 8400 HighLine/TopLine: 50 logbook entries

Basic workflow



- The logbook can be read out by an external control or visualisation via the procedure shown on the left.
- The "Response index" query ensures that the logbook entry read really corresponds to the logbook entry requested.

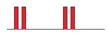
Related topics:

▶ Structure of the 32-bit error number (bit coding)

10.6 Monitoring

The inverter is provided with various monitoring functions which protect the drive against impermissible operating conditions.

- If a monitoring function responds,
  - an entry will be made into the Logbook of the inverter,
  - the response (TroubleQSP, Warning, Fault, etc.) set for this monitoring function will be triggered,
  - the status of the internal device control changes according to the selected response, controller inhibit is set, and the "DRV- ERR" LED on the front of the inverter goes on:

Response	Entry in the logbook	Display in C00168	Pulse inhibit	Disable drive function	Acknowledgement required	LED "DRV-ERR"
None						OFF
Fault	☑	☑	☑	☑	☑	
Trouble	☑	☑	☑	☑		
TroubleQSP	☑	☑			☑	
WarningLocked	☑	☑			☑	
Warning	☑	☑				
Information	☑	☑				OFF

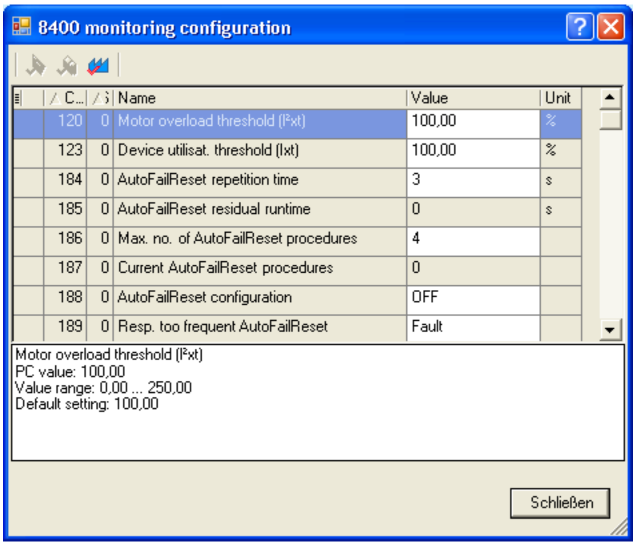
Related topics:

- LED status displays of the device status (📖 721)
- Device state machine and device states (📖 119)
- Device overload monitoring (Ixt) (📖 309)
- Motor load monitoring (I2xt) (📖 310)
- Motor temperature monitoring (PTC) (📖 313)
- Brake resistor monitoring (I2xt) (📖 314)
- Motor phase failure monitoring (📖 316)
- Mains phase failure monitoring (📖 319)
- Current monitoring for overload (📖 319)
- Maximum current monitoring (📖 320)
- Maximum torque monitoring (📖 320)
- Resolver/multi-encoder open-circuit monitoring (📖 390)
- Open-circuit monitoring - HTL encoder (📖 392)
- Motor temperature monitoring (PT1000 or KTY) (📖 395)

10.6.1 Monitoring configuration

How to configure the monitoring functions using the »Engineer«:

1. Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Diagnostics** tab from the *Workspace*.
3. Click the **Monitoring...** button.
   - The *8400 monitoring configuration* dialog box is displayed via which the desired settings can be made:



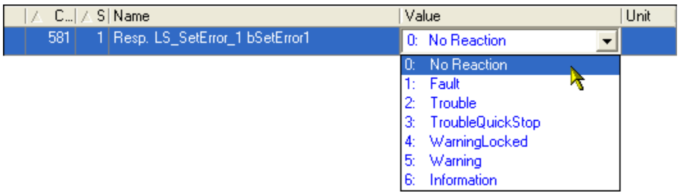
Related topics:

▶ Setting the error response (📖 738)

10.6.2 Setting the error response

When a monitoring function responds, the response set for this monitoring function (TroubleQSP, Warning, Fault, etc.) will be triggered.

- For many monitoring functions the response can be individually parameterised via parameters.



Tip!

The table in the chapter "Short overview (A-Z)" contains the error messages for which the response can be set. (📖 749)

Warning thresholds

Some of the monitoring functions are activated if a defined warning threshold (e.g. temperature) has been exceeded.

- The corresponding preset threshold values can be changed via the following parameters:

Parameters	Info
C00120	Setting of motor overload (I²xt)
C00123	Device utilisat. threshold (Ixt)
C00572	Brake resistor overload threshold
C00599	Motor phase failure threshold

10.6.3 AutoFailReset function

The AutoFailReset function serves to automatically reset the latching "Fault" and "TroubleQSP" errors as well as the latching "WarningLocked" warning.

The "latching" term means that the effect on the inverter remains active even after the error cause has been removed.

To reset latching errors and warnings, the following options are available:

- Manual reset
  - with device command C00002/19 (activated by Low-High edge)
  - by a Low-High edge at the *bResetFail* input of the LS_DriveInterface (the "FailReset" control bit in the control word must be parameterised with a value of "1").
- Automatic reset
  - using the AutoFailReset function.

Overview of the relevant parameters

Parameters	Info
C00184	Repetition time of the error reset processes
C00185	Time left until the next error reset process
C00186	Max. number of permissible unsuccessful error reset processes • When the number set is reached, the response parameterised in C00189 is executed.
C00187	Current number of unsuccessful error reset processes carried out
C00188	Configuration of the AutoFailReset function • 0: Off • 1: Fault + TroubleQSP • 2: WarningLocked • 3: All locking
C00189	Response after max. number of unsuccessful error reset processes has been reached
Greyed out = display parameter	

10.7 Maloperation of the drive

Maloperation	Cause	Remedy
Motor does not rotate	DC-bus voltage is too low • Red LED is blinking every 1 s • Display in the keypad: *LU*	Check mains voltage
	Inverter is inhibited • Green LED is blinking • Display in the keypad: CINH	Deactivate controller inhibit • Note: Controller inhibit can be set via several sources ! • C00158 displays all active sources for controller inhibit.
	Automatic start is inhibited (Bit 0 in C00142 = 1)	LOW/HIGH edge at RFR If required, correct starting condition with C00142
	DC-injection braking (DCB) is active	Deactivate DC injection brake
	Mechanical motor brake is not released	Release mechanical motor brake manually or electrically
	Quick stop (QSP) is active • Display in the keypad: IMP	Deactivate quick stop • Note: Quick stop can be set via several sources! • C00159 displays all active sources for quick stop.
	Setpoint = 0	Select setpoint
	JOG frequency = 0 at activated JOG setpoint	Set JOG setpoint in C00039/1...15
	Trouble active	Clear fault
	With C00006 = 4, "SLVC: Vector control" has been set, but no motor parameter identification has been carried out.	Execute automatic motor parameter identification with the C00002/23 device command
	Assignment of several mutually exclusive functions with a signal source in C00701	Correct configuration in C00701
Motor rotates irregularly	Motor cable is defective	Check motor cable
	Maximum motor current in motor or generator mode is set too low	Adjust settings to the application: C00022: Imax in motor mode C00023: Imax in generator mode
	Motor is underexcited or overexcited	Check parameterisation: C00006: Motor control C00015: VFC: V/f base frequency C00016: VFC: Vmin boost
	Rated motor data (stator resistance, speed, current, frequency, voltage) and cos φ and/or magnetising inductance is not adapted to the motor data	Execute automatic motor parameter identification with the C00002/23 device command - or - Adjust motor parameters manually: C00084: Motor stator resistance C00087: Rated motor speed C00088: Rated motor current C00089: Rated motor frequency C00090: Rated motor voltage C00091: Motor cosine phi C00092: Motor magnetising inductance
	Motor windings are wired incorrectly	Reverse from star connection to delta connection

Maloperation	Cause	Remedy
Motor consumes too much current	V_{min} boost has been selected too high	Correct setting with C00016
	V/f base frequency has been selected too low	Correct setting with C00015
	Rated motor data (stator resistance, speed, current, frequency, voltage) and cos φ and/or magnetising inductance is not adapted to the motor data	Execute automatic motor parameter identification with the C00002/23 device command - or - Adjust motor parameters manually: C00084: Motor stator resistance C00087: Rated motor speed C00088: Rated motor current C00089: Rated motor frequency C00090: Rated motor voltage C00091: Motor cosine phi C00092: Motor magnetising inductance
Motor parameter identification is aborted with error LP1	Motor is too small compared to the rated device power (>1 : 3)	Use device with lower rated power
	DC injection brake (DCB) is active via terminal	Deactivate DC injection brake
Drive behaviour with vector control is not satisfactory	different	Optimise or manually adapt vector control
		Execute automatic motor parameter identification with the C00002/23 device command
Torque dip in field weakening range or motor stalling when being operated in the field weakening range	Motor is overloaded	Check motor load
	Motor windings are wired incorrectly	Reverse from star connection to delta connection
	V/f reference point is set too high	Correct setting with C00015
	Override point of field weakening is set too low	Correct setting with C00080

An asynchronous motor with feedback rotates without control and with too low speed	Motor phases have been interchanged
	• Thus the rotating field of the motor is not identical anymore with the rotating field of the feedback system.
	• Therefore, the drive shows the following behaviour if V/f characteristic control (C00006 = 7) is performed:
	  • The motor rotates faster than the speed setpoint by the value set in C00074.
	  • After the controller has been enabled, the inverter will not stop if the speed setpoint = 0 or a quick stop (QSP) occurs.
	  • Among other things, the final motor current depends on the value set for the V_{min} boost and may rise up to I_{max} which can trigger the "oC5: Ixt overload" fault message.
		Check the phase position of the motor cable
		If possible: Actuate the motor with deactivated feedback (C00006 = 6) and check direction of rotation of the motor

Motor phase (LP1) monitoring does not respond if the motor phases are interrupted	Monitoring is not active (C00597 = 0)	Activate monitoring (C00597 = 1)

10.8 Operation without mains supply

Note!

Observe the following restrictions for operation without mains supply:

Safety state

Inverters of the 8400 series can be optionally equipped with the integrated "Safe torque off (STO)" safety system.

- Up to and including version 13.xx.xx, the following applies:
  If merely the external 24-V supply of the inverter is switched on, the "Safe torque off" status in C00137 (bit 10) will not be updated.
- From version 14.00.00 the following applies:
  If merely the external 24-V supply of the inverter is switched on, the "Safe torque off" status in C00137 (bit 10) will be updated.

Fan monitoring functions

The fan monitoring functions are only active if the mains supply is switched on.

The following display parameters have a value of "0" if the mains supply is switched off and the external 24 V supply of the inverter is switched on:

Parameters	Info
C00050	MCTRL: Speed setpoint
C00051	MCTRL: Actual speed value
C00052	Motor voltage
C00053	DC-bus voltage
C00054	Motor current
C00058	Output frequency
C00061	Heatsink temperature
C00064/1...3	Device utilisation (Ixt)
C00066	Thermal motor load (I^2xt)
C00177	Switching cycles
C00725	Current switching frequency

10.9 Error messages of the operating system

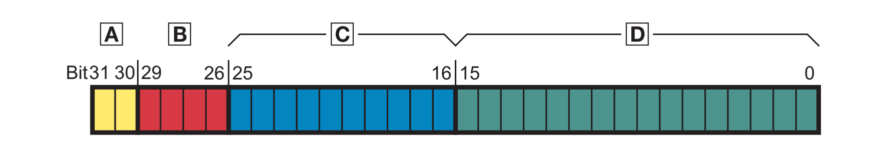
This chapter describes all error messages of the inverter operating system and possible causes & remedies.

Tip!

Each error message is also saved to the logbook in chronological order. ▶ Logbook (📖 728)

10.9.1 Structure of the 32-bit error number (bit coding)

If an error occurs in the inverter, the internal fault memory saves a 32-bit value which contains the following information:

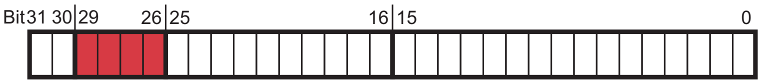


Ⓐ Reserved
Ⓑ Error type
Ⓒ Error subject area
Ⓓ Error ID

[10-1] Structure of the error number

- Display parameter: C00168
  - From version 13.00.00: C00162/1 additionally displays the 32-bit error number without error type, i.e. this error number only contains the error subject area and the error ID (bit 0 ... bit 25).
- The LS_DriveInterface system block shows the 32-bit error number at the outputs *wStateDetermFailNoLow* (Low Word) and *wStateDetermFailNoHigh* (High Word).
  - If the "Use 16BitFailNo." option is activated in C00148 (bit 15 = "1"), the LS_DriveInterface system block shows the short 16-bit error number at the output *wStateDetermFailNoLow*, and the value "0" at the output *wStateDetermFailNoHigh* (see the following chapter).
- For the sake of legibility, the error number in the logbook and in C00165 is displayed with the following syntax:
  [Error type].[Error subject area no.].[Error ID]

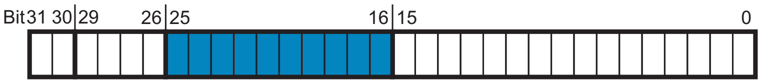
Error type



The error type gives information about the behaviour/response of the inverter to the error. The error type for some device errors can also be parameterised.

Bit 29	Bit 28	Bit 27	Bit 26	Meaning
0	0	0	0	0: No Response
0	0	0	1	1: Fault
0	0	1	0	2: Trouble
0	0	1	1	3: TroubleQSP
0	1	0	0	4: WarningLocked
0	1	0	1	5: Warning
0	1	1	0	6: Information

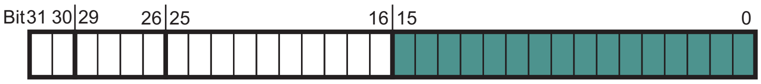
Error subject area



The error subject area indicates the internal "function unit" of the inverter in which the error has occurred:

Error subject area		Assigned errors	Remedy possible by user?
No.	**Name**		
111	Supply voltage	Errors that occur in connection with the supply voltage of the device.	Yes
119	Temperature	Errors that occur for temperature reasons.	Yes
123	Motor management / encoder	Errors that occur within the motor control or encoder evaluation.	Yes
125	Analog I/O integrated	Errors that occur in connection with the analog inputs and outputs.	Yes
126	Axis bus	Errors that occur in connection with the axis bus.	Yes
127	Extension module slot 1	Errors that are reported by the extension module, and communication errors to the plugged-in extension module.	Yes if it is a fieldbus error.
131	CAN integrated (general)	Errors related to general CAN functions.	Yes
135	CAN process data object (PDO)	Errors that are explicitly only related to the CAN-PDO (process data objects).	Yes
140	Device configuration	Errors that occur due to incompatibilities of the plugged-in individual components (fieldbus module, safety module, et al.).	Yes if the error relates to a module plugged-in by the user.
144	Parameter set	Errors that occur in connection with the parameter set or the parameter set memory (memory module).	Yes if the error relates to a missing or incompatible memory module.
145	Device firmware (internal error)	Internal error of the device firmware.	No
184	MotionControlKernel	Errors that occur within the MotionControl basic functions (e.g. profile generation, brake control, positioning).	Yes
400	Defective device hardware	Errors that occur due to defective device hardware.	No
444	Fieldbus	Errors that occur in connection with fieldbus communication.	Yes
980 ... 983	User error 1 ... User error 4	Errors generated by the user (by the application) via the LS_SetError_1 system block.	Yes
984 ... 987	User error 5 ... User error 8	Errors generated by the user (by the application) via the LS_SetError_2 system block.	Yes

Error ID

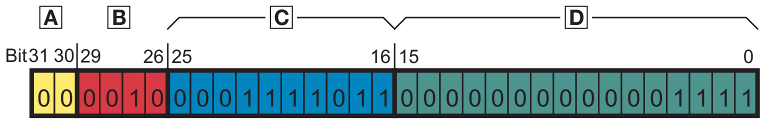


16-bit value (0 ... 65535) for error identification within the error subject area.

Example for bit coding of the error number

C00168 displays the internal error number "142278671".

- This decimal value corresponds to the following bit sequence:

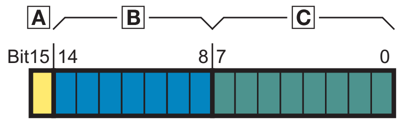


Assignment	Information	Meaning in the example
0 0	Reserved	-
0 0 1 0	Error type	2: Trouble
0 0 0 1 1 1 1 0 1 1	Error subject area	123: Motor management / encoder
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1	Error ID	15: "LU: DC bus undervoltage"

- Thus, error number "142278671" means:
  A DC bus undervoltage has been detected in the "Motor management / encoder" subject area.
  The error response is a "Fault".

10.9.2 Structure of the 16 bit error number (bit coding)

In addition to the 32-bit error number, a 16-bit error number is generated if an error occurs. It consists of the following information:

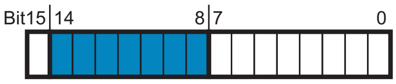


A Reserved
B Error subject area
C Error ID

[10-2] Structure of the error number

- Display parameter: C00160
- The LS_DriveInterface system block shows the 16-bit error number at the output *wStateDetermFailNoShort*.
- If the "Use 16BitFailNo." option is activated in C00148 (bit 15 = "1"), the LS_DriveInterface system block also shows the short 16-bit error number at the output *wStateDetermFailNoLow* (Low Word of the 32-bit error number).
  - The output *wStateDetermFailNoHigh* (High Word of the 32-bit error number) in this case is "0".
  - Advantage: The bus transfer of the error numbers is possible via a data word without changing the interconnection of the technology application.
- For the sake of legibility, the 16-bit error number in the logbook is displayed with the following syntax::
  [**Error subject area no.**].[**Error ID**]

Error subject area



The error subject area indicates the internal "function unit" of the inverter in which the error has occurred.

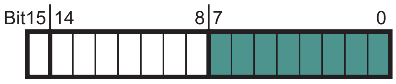
Note!

Due to the smaller value range (0 ...127), the number assignment to the error subject area differs from the 32-bit error number.

Error subject area		Assigned errors	Remedy possible by user?
No.	**Name**		
11	Supply voltage	Errors that occur in connection with the supply voltage of the device.	Yes
19	Temperature	Errors that occur for temperature reasons.	Yes
23	Motor management / encoder	Errors that occur within the motor control or encoder evaluation.	Yes
25	Analog I/O integrated	Errors that occur in connection with the analog inputs and outputs.	Yes
26	Defective device hardware	Errors that occur due to defective device hardware.	No
27	Extension module slot 1	Errors that are reported by the extension module, and communication errors to the plugged-in extension module.	Yes if it is a fieldbus error.
31	CAN integrated (general)	Errors related to general CAN functions.	Yes
35	CAN process data object (PDO)	Errors that are explicitly only related to the CAN-PDO (process data objects).	Yes

Error subject area		Assigned errors	Remedy possible by user?
No.	**Name**		
40	Device configuration	Errors that occur due to incompatibilities of the plugged-in individual components (fieldbus module, safety module, et al.).	Yes if the error relates to a module plugged-in by the user.
44	Parameter set	Errors that occur in connection with the parameter set or the parameter set memory (memory module).	Yes if the error relates to a missing or incompatible memory module.
45	Device firmware (internal error)	Internal error of the device firmware.	No
54	Fieldbus	Errors that occur in connection with fieldbus communication.	Yes
84	MotionControlKernel	Errors that occur within the MotionControl basic functions (e.g. profile generation, brake control, positioning).	Yes
100 ... 103	User error 1 ... User error 4	Errors generated by the user (by the application) via the LS_SetError_1 system block.	Yes
104 ... 107	User error 5 ... User error 8	Errors generated by the user (by the application) via the LS_SetError_2 system block.	Yes

Error ID



8-bit value (0 ... 255) for error identification within the error subject area.

Tip!

All possible 16-bit error numbers are listed in the table entitled "Short overview (A-Z)" in the second column. (📖 749)

10.9.3 Reset error message

An error message with the response "Fault", "Trouble", "TroubleQSP" or "Warning locked" must be explicitly reset (acknowledged) after the cause of the error has been eliminated.

To reset (acknowledge) a pending error message, execute device command C00002/19 = "1".

Tip!

When an online connection to the inverter has been established, use the **Diagnostics** tab of »Engineer« and click **Reset error** to reset a pending error message.

10.9.4 Export error texts

All error texts of the inverter can be exported into a text file (*.txt) for further processing.

- The error text is preceded with the corresponding 32-bit error number (no error type) and the 16-bit error number, both as decimal number.
- If there is no corresponding 16-bit error number for a 32-bit error number, the field remains empty.

Example

Output of the German and English error texts:

```
32-BitError   16-BitError   DE-de                        EN-en
0             0             No error                 No error
111           11            Versorgungsspannung          Supply voltage
119 / 19
12323 : Motor management / encoder
125           25            E/A integriert               I/O integrated
...
26214416      6672          dH10: Lüfterausfall          dH10: Fan failure
26214505      6761          dH69: Abgleichdatenfehler    dH69: Adjustment fault
```

How to export the error texts into a text file:

1. Go to the *Project view* in the *Context menu* of the 8400 TopLine inverter and execute the **Export error texts...** command.
2. Define the following options in the *Export error texts* dialog box:
   - Output file and memory location
   - Languages to be exported (German/English/French)
   - Device/module to be exported
   - Separator (tabulator or semicolon)
   - Font (UTF8, standard font or ASCII)
3. Click **OK** to start the export.
   - After the export, a message appears indicating whether the export was successful.

Tip!

From version 13.00.00, the 32-bit number of the state-determining error is displayed in C00162/1 without error type.

If, for instance, the error texts are stored in a master control or on an operator panel, the error text to be displayed can be detected by reading the C00162/1 code.

10.9.5 Short overview (A-Z)

The table below contains all error messages of the inverter operating system in alphabetical order.

Note!

For the sake of legibility, the Logbook and C00165 display the 32-bit error number with the following syntax:
[**Error type**].[**Error subject area no.**].[**Error ID**]

In this documentation, "xx", a wildcard, stands for the error type since it is configurable for many error messages.

Tip!

If you click the cross-reference in the first column, "Error number", you will reach the detailed description of the respective error message in the following chapter "Cause & possible remedies". (📖 753)

Error number 32 bits	16 $bits_{hex}$	16 $bits_{dec}$	Display in C00162/1	Error message	Response (Lenze setting)	Adjustable in	CAN emergency error code
* Only the lower 8 bits of the adjustable error ID (C161/x) can be used.							
xx.0126.00001	0x7e01	32257	8257537	Ab01: Axis bus time-out	No Reaction	C00591/1	
xx.0126.00002	0x7e02	32258	8257538	Ab02: Axis bus IO error	No Reaction	C00591/2	
xx.0126.00003	0x7e03	32259	8257539	Ab03: axis bus IO requested	Information	C00591/3	
xx.0125.00001	0x1901	6401	8192001	An01: AIN1_I < 4 mA	TroubleQuickStop	C00598/1	0xF000
xx.0125.00002	0x1902	6402	8192002	An02: AIN2_I < 4 mA	TroubleQuickStop	C00598/2	0xF000
xx.0131.00006	0x1f06	7942	8585222	CA06: CAN CRC error	No Reaction	C00592/1	0x8000
xx.0131.00007	0x1f07	7943	8585223	CA07: CAN Bus Warn	No Reaction	C00592/3	0x8000
xx.0131.00008	0x1f08	7944	8585224	CA08: CAN Bus Stopped	No Reaction	C00592/4	0x8000
xx.0131.00011	0x1f0b	7947	8585227	CA0b: CAN HeartBeatEvent	No Reaction	C00592/5	0x8130
xx.0131.00015	0x1f0f	7951	8585231	CA0F: CAN control word	Fault	C00594/1	0xF000
xx.0127.00002	0x1b02	6914	8323074	CE04: MCI communication error	No Reaction	C01501/1	0x7000
xx.0127.00015	0x1b0f	6927	8323087	CE0F: MCI control word	Fault	C00594/2	0xF000
xx.0135.00001	0x2301	8961	8847361	CE1: CAN RPDO1	No Reaction	C00593/1	0x8100
xx.0135.00002	0x2302	8962	8847362	CE2: CAN RPDO2	No Reaction	C00593/2	0x8100
xx.0135.00003	0x2303	8963	8847363	CE3: CAN RPDO3	No Reaction	C00593/3	0x8100
xx.0131.00000	0x1f00	7936	8585216	CE4: CAN Bus Off	No Reaction	C00592/2	0x8000
xx.0135.00004	0x2304	8964	8847364	CE5: CAN RPDO4	No Reaction	C00593/4	0x8100
xx.0140.00013	0x280d	10253	9175053	CI01: Module missing/incompatible	No Reaction	C01501/2	0x7000
xx.0184.00001	0x5401	21505	12058625	Ck01: Pos. HW limit switch	TroubleQuickStop	C00595/1	0x8600
xx.0184.00002	0x5402	21506	12058626	Ck02: Neg. HW limit switch	TroubleQuickStop	C00595/2	0x8600
xx.0184.00007	0x5407	21511	12058631	Ck03: Pos. SW limit position	TroubleQuickStop	C00595/3	0x8600
xx.0184.00008	0x5408	21512	12058632	Ck04: Neg. SW limit position	TroubleQuickStop	C00595/4	0x8600
xx.0184.00153	0x5499	21657	12058777	Ck05: Error following error 1	Warning	C00595/5	0x8611
xx.0184.00154	0x549a	21658	12058778	Ck06: Error following error 2	Warning	C00595/6	0x8611
xx.0184.00155	0x549b	21659	12058779	Ck07: Traversing range limit exceeded	TroubleQuickStop	C00595/7	0x8612
xx.0184.00156	0x549c	21660	12058780	Ck08: Reference position unknown	WarningLocked	C00595/8	0x8612
xx.0184.08005	0x54cd	21709	12066629	Ck09: Positioning mode invalid	WarningLocked	C00595/9	0x8600
xx.0184.08007	0x54cf	21711	12066631	Ck10: Profile data implausible	WarningLocked	C00595/10	0x8600
xx.0184.08009	0x54d1	21713	12066633	Ck11: Operating mode invalid	Warning	C00595/11	0x8600
xx.0184.08014	0x54d6	21718	12066638	Ck12: Profile number invalid	WarningLocked	C00595/12	0x8600

Error number 32 bits	16 bits$_{hex}$	16 bits$_{dec}$	Display in C00162/1	Error message	Response (Lenze setting)	Adjustable in	CAN emergency error code
xx.0184.08015	0x54d7	21719	12066639	Ck13: Error FB MCKCtrlInterface	Warning	C00595/13	0x8600
xx.0184.00015	0x540f	21519	12058639	Ck14: Target position outside SW limit position	WarningLocked	C00595/14	0x8600
xx.0184.00005	0x5405	21509	12058629	Ck15: Error message sig. brake	TroubleQuickStop	-	0x8600
xx.0184.00064	0x5440	21568	12058688	Ck16: Time overrun manual operation	Fault	-	
xx.0184.00009	0x5409	21513	12058633	Ck17: direction conflict Ccw	Information	-	
xx.0184.00010	0x540a	21514	12058634	Ck18: direction conflict Cw	Information	-	
xx.0145.00001	0x2d01	11521	9502721	dF01: FW updated	No Reaction	-	
xx.0145.00035	0x2d23	11555	9502755	dF10: AutoTrip reset	Fault	C00189	0xF000
xx.0145.00014	0x2d0e	11534	9502734	dF14: SW-HW invalid	Fault	-	
xx.0145.00015	0x2d0f	11535	9502735	dF15: DCCOM CU2 error	Fault	-	
xx.0145.00024	0x2d18	11544	9502744	dF18: BU RCOM error	Fault	-	0x6100
xx.0145.00033	0x2d21	11553	9502753	dF21: BU watchdog	Fault	-	0x6100
xx.0145.00034	0x2d22	11554	9502754	dF22: CU Watchdog	Fault	-	0x6100
xx.0145.00025	0x2d19	11545	9502745	dF25: CU RCOM error	Fault	-	0x6100
xx.0145.00026	0x2d1a	11546	9502746	dF26: Appl. watchdog	No Reaction	C00580/1	0x6200
xx.0145.00050	0x2d32	11570	9502770	dF50: Retain error	Fault	-	0x6100
xx.0145.00051	0x2d33	11571	9502771	dF51: CuCcr error	Fault	-	0x6100
xx.0145.00052	0x2d34	11572	9502772	dF52: BuCcr error	Fault	-	0x6100
xx.0400.00009	0x1a09	6665	26214409	dH09: EEPROM power unit	Fault	-	0x5500
xx.0400.00016	0x1a10	6672	26214416	dH10: Fan failure	Warning	C00566	0x5000
xx.0400.00104	0x1a68	6760	26214504	dH68: Adjustment data error CU	Fault	-	0x5500
xx.0400.00105	0x1a69	6761	26214505	dH69: Adjustment data error BU	Fault	-	0x5500
xx.0400.00106	0x1a6a	6762	26214506	dH70: ControlUnit is unequal to BaseUnit	Fault	-	0x5500
xx.0123.00094	0x175e	5982	8061022	FC01: Switching frequency reduction	No Reaction	C00590	0x2000
xx.0123.00095	0x175f	5983	8061023	FC02: Maximum speed for Fchop	No Reaction	C00588	0xF000
xx.0123.00099	0x1763	5987	8061027	FC03: Limitation field controller	No Reaction	C00570/4	0xF000
xx.0123.00057	0x1739	5945	8060985	Id1: Motor data identification error	Fault	-	0xF000
xx.0123.00056	0x1738	5944	8060984	Id2: Motor data identification error	Fault	-	0xF000
xx.0123.00058	0x173a	5946	8060986	Id3: CINH identification	WarningLocked	-	0xF000
xx.0123.00059	0x173b	5947	8060987	Id4: Resistance identification error	Warning	-	0xF000
xx.0123.00074	0x174a	5962	8061002	Id5: Pole position identification error	Fault	C00643/1	
xx.0123.00075	0x174b	5963	8061003	Id6: Resolver ident error	Fault	-	
xx.0123.00060	0x173c	5948	8060988	Id7: Motor control does not match motor data	Information	C00571/1	0xF000
xx.0123.00061	0x173d	5949	8060989	Id8: Speed sensor has not been set	Fault	C00571/2	0x7120
xx.0123.00145	0x1791	6033	8061073	LP1: Motor phase failure	No Reaction	C00597	0x3000
xx.0123.00015	0x170f	5903	8060943	LU: DC bus undervoltage	Trouble	C00600/1	0x3100
xx.0123.00016	0x1710	5904	8060944	oC1: Power section - short circuit	Fault	-	0x2000
xx.0123.00030	0x171e	5918	8060958	oC10: Maximum current reached	No Reaction	C00609	0x2000
xx.0123.00071	0x1747	5959	8060999	oC11: Clamp operation active	Fault	-	0xF000
xx.0123.00065	0x1741	5953	8060993	oC12: I2xt brake resistor overload	No Reaction	C00574	0xF000
xx.0123.00090	0x175a	5978	8061018	oC13: Maximum current for Fch exceeded	Fault	-	0xF000
xx.0123.00096	0x1760	5984	8061024	oC14: Direct-axis current controller limitation	No Reaction	C00570/1	0xF000
xx.0123.00097	0x1761	5985	8061025	oC15: Cross current controller limitation	No Reaction	C00570/2	0xF000
xx.0123.00098	0x1762	5986	8061026	oC16: Torque controller limitation	No Reaction	C00570/3	0xF000
xx.0123.00031	0x171f	5919	8060959	oC17: Clamp sets pulse inhibit	No Reaction	C00569/1	0xF000
xx.0123.00034	0x1722	5922	8060962	oC18: Current monitoring overload	No Reaction	C00584/1	0x2000
xx.0123.00066	0x1742	5954	8060994	oC19: short circuit of brake resistor	Fault	-	0xF000
xx.0123.00017	0x1711	5905	8060945	oC2: Power section - earth fault	Fault	-	0x2000
xx.0119.00050	0x1332	4914	7798834	oC5: Ixt overload	Warning	C00604	0x2000
xx.0123.00105	0x1769	5993	8061033	oC6: I2xt motor overload	Warning	C00606	0x2000
xx.0123.00007	0x1707	5895	8060935	oC7: Motor overcurrent	Fault	-	0x2000

Error number 32 bits	16 bits$_{hex}$	16 bits$_{dec}$	Display in C00162/1	Error message	Response (Lenze setting)	Adjustable in	CAN emergency error code
xx.0119.00001	0x1301	4865	7798785	oH1: Heatsink overtemperature	Fault	-	0x4000
xx.0119.00021	0x1315	4885	7798805	oH12: Motor overtemperature MultiEncoder	Fault	C00583/2	
xx.0119.00015	0x130f	4879	7798799	oH3: Motor temperature (X106) triggered	Fault	C00585	0x4000
xx.0119.00000	0x1300	4864	7798784	oH4: Heatsink temp. > shutdown temp. -5°C	No Reaction	C00582	0x4000
xx.0119.00020	0x1314	4884	7798804	oH6: Motor temperature MultiEncoder >= C121	Warning	C00583/4	
xx.0119.00002	0x1302	4866	7798786	oH7: Motor temperature resolver >= C121	Warning	C00583/3	
xx.0119.00003	0x1303	4867	7798787	oH9: Motor overtemperature resolver	Fault	C00583/1	
xx.0123.00032	0x1720	5920	8060960	oS1: Maximum speed limit reached	No Reaction	C00579	0x8400
xx.0123.00033	0x1721	5921	8060961	oS2: Max. motor speed	Fault	-	0x8400
xx.0123.00001	0x1701	5889	8060929	ot1: Max. torque reached	No Reaction	C00608	0x8300
xx.0123.00093	0x175d	5981	8061021	ot2: Speed controller output limited	No Reaction	C00567	0xF000
xx.0123.00014	0x170e	5902	8060942	OU: DC bus overvoltage	Trouble	-	0x3100
xx.0144.00001	0x2c01	11265	9437185	PS01: No memory module	Warning	-	0x6300
xx.0144.00002	0x2c02	11266	9437186	PS02: Par. set invalid	Fault	-	0x6300
xx.0144.00003	0x2c03	11267	9437187	PS03: Par. set device invalid	Fault	-	0x6300
xx.0144.00004	0x2c04	11268	9437188	PS04: Invalid MCI par. set	Fault	-	0x6300
xx.0144.00007	0x2c07	11271	9437191	PS07: Par. mem. module invalid	Fault	-	0x6300
xx.0144.00008	0x2c08	11272	9437192	PS08: Par. device invalid	Fault	-	0x6300
xx.0144.00009	0x2c09	11273	9437193	PS09: Par. format invalid	Fault	-	0x6300
xx.0144.00010	0x2c0a	11274	9437194	PS10: Memory module link invalid	Fault	-	
xx.0144.00011	0x2c0b	11275	9437195	PS11: Lenze setting loaded	No Reaction	-	
xx.0144.00012	0x2c0c	11276	9437196	PS12: Parameter sets loaded	No Reaction	-	
xx.0144.00013	0x2c0e	11277	9437197	PS13: Parameter sets saved	No Reaction	-	
xx.0123.00024	0x1718	5912	8060952	Sd2: Resolver open circuit	Fault	C00603/2	
xx.0123.00205	0x17cd	6093	8061133	Sd3: Open circuit HTL 2-fold or 4-fold	Fault	C00586	0x7300
xx.0123.00027	0x171b	5915	8060955	Sd4: MultiEncoder open circuit	Fault	C00603/1	
xx.0119.00012	0x130c	4876	7798796	Sd6: Error thermal detector resolver	Fault	C00583/5	
xx.0123.00026	0x171a	5914	8060954	Sd7: Error encoder communication	Fault	C00603/4	
xx.0123.00062	0x173e	5950	8060990	Sd8: Encoder angular drift monit.	No Reaction	C00603/3	
xx.0123.00028	0x171c	5916	8060956	Sd9: Status message Hiperface	Information	C00603/5	0x7300
xx.0123.00200	0x17c8	6088	8061128	Sd10: Speed limit for feedback system 12	Fault	C00607	0x7300
xx.0123.00201	0x17c9	6089	8061129	Sd11: Speed limit for feedback system 67	Fault	C00607	0x7300
xx.0119.00022	0x1316	4886	7798806	Sd12: Error thermal detector MultiEncoder	Fault	C00583/6	
xx.0123.00206	0x17ce	6094	8061134	Sd13: Inaccuracy SinCos	Information	-	0x7300
xx.0123.00029	0x171d	5917	8060957	Sd14: Position invalid Hiperface	Information	C00603/6	0x7300
xx.0123.00207	0x17cf	6095	8061135	Sd15: Open circuit HTL 4-fold	Fault	C00605/1	0x7300
xx.0123.00208	0x17d0	6096	8061136	Sd16: Hiperface-SinCos deviation	No Reaction	C00603/8	0x7300
xx.0123.00209	0x17d1	6097	8061137	Sd17: Encoder supply	Fault	C00603/9	0x7300
xx.0123.00210	0x17d2	6098	8061138	Sd18: V/f emergency operation	Information	-	
xx.0111.00002	0x0b02	2818	7274498	Su02: One mains phase is missing	Warning	C00565	0x3000
xx.0111.00003	0x0b03	2819	7274499	Su03: Too frequent mains switching	Fault	-	0x3000
xx.0111.00004	0x0b04	2820	7274500	Su04: CU insufficiently supplied	Warning	-	0x3000
xx.0111.00006	0x0b06	2822	7274502	Su06: Power input overload	Fault	-	0x3000
xx.0111.00007	0x0b07	2823	7274503	Su07: 24V supply off	No Reaction	-	-
Freely configurable user error messages (see LS_SetError_1 and LS_SetError_2)							
xx.0980.00001	25600_{dec} + C161/1*			User error 1	No Reaction	C00581/1	0x6200
xx.0981.00002	25856_{dec} + C161/2*			User error 2	No Reaction	C00581/2	0x6200
xx.0982.00003	26112_{dec} + C161/3*			User error 3	No Reaction	C00581/3	0x6200
xx.0983.00004	26368_{dec} + C161/4*			User error 4	No Reaction	C00581/4	0x6200
xx.0984.00001	26624_{dec} + C161/5*			User error 5	No Reaction	C00581/5	0x6200

Error number			Display in	Error message	Response (Lenze setting)	Adjustable in	CAN emergency error code
32 bits	16 bits$_{hex}$	16 bits$_{dec}$	C00162/1				
▸ xx.0985.00002	26880$_{dec}$ + C161/6*			User error 6	No Reaction	C00581/6	0x6200
▸ xx.0986.00003	27136$_{dec}$ + C161/7*			User error 7	No Reaction	C00581/7	0x6200
▸ xx.0987.00004	27392$_{dec}$ + C161/8*			User error 8	No Reaction	C00581/8	0x6200

* Only the lower 8 bits of the adjustable error ID (C161/x) can be used.

10.9.6 Cause & possible remedies

This chapter contains all error messages of the inverter operating system in numerical order of the error numbers. The list provides detailed information on the response to the error message as well as information on the cause & possible remedies.

Note!

For the sake of legibility, the Logbook and C00165 display the error number with the following syntax:
[**Error type**].[**Error subject area no.**].[**Error ID**]

In this documentation, "xx", a wildcard, stands for the error type since it is configurable for many error messages.

Tip!

A list of all error messages of the inverter operating system in alphabetical order can be found in the previous chapter "Short overview (A-Z)" (🕮 749).

User error 1 [xx.0980.00000 ... xx.0980.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 1 has been tripped via the *bSetError1* input of the LS_SetError_1 system block.	User-defined.

User error 2 [xx.0981.00000 ... xx.0981.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/2 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 2 has been tripped via the *bSetError2* input of the LS_SetError_1 system block.	User-defined.

User error 3 [xx.0982.00000 ... xx.0982.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/3 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 3 has been tripped via the *bSetError3* input of the LS_SetError_1 system block.	User-defined.

User error 4 [xx.0983.00000 ... xx.0983.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/4 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 4 has been tripped via the *bSetError4* input of the LS_SetError_1 system block.	User-defined.

User error 5 [xx.0984.00000 ... xx.0984.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/5 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 5 has been tripped via the *bSetError1* input of the LS_SetError_2 system block.	User-defined.

User error 6 [xx.0985.00000 ... xx.0985.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/6 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 6 has been tripped via the *bSetError2* input of the LS_SetError_2 system block.	User-defined.

User error 7 [xx.0986.00000 ... xx.0986.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/7 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 7 has been tripped via the *bSetError3* input of the LS_SetError_2 system block.	User-defined.

User error 8 [xx.0987.00000 ... xx.0987.65535]

Response (Lenze setting printed in bold)	**Setting:** C00581/8 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
User error 8 has been tripped via the *bSetError4* input of the LS_SetError_2 system block.	User-defined.

Su02: One mains phase is missing [xx.0111.00002]

Response (Lenze setting printed in bold)	**Setting:** C00565 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☒ **5: Warning** ☐ 6: Information	

Cause	**Remedy**
A mains phase of a three-phase supply has failed.	Check mains connection (terminal X100).

Su03: Too frequent mains switching [xx.0111.00003]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Too frequent mains switching of the power section. • The device recognises if the power section is switched on and off too frequently. • To protect internal charging connections from destruction, the device reports this error and prevents the controller inhibit. All other functions are active. Use of a power supply module in the DC-bus connection, the DC terminals of which are connected downstream to the charging connection for the voltage DC bus (e.g. 9400 from 45 kW).	The error must be acknowledged by mains switching. The charging circuit can only cool down when the mains is switched off. • After switching the mains 3 times in one minute, there must be a switching pause of 9 minutes. • Cyclic mains switching every 3 minutes is permissible. From version 12.00.00 onwards, this power supply module can be used in the DC-bus connection by enabling it via C02865 (bit 8). **Note:** For further configuration of devices in the DC-bus connection with 8400, the DC terminals of which are connected downstream to the charging connection for the voltage DC bus (e.g. 9400 from 45 kW with 8400) contact Lenze.

Su04: CU insufficiently supplied [xx.0111.00004]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☒ **5: Warning** ☐ 6: Information

Cause	Remedy
After switching on the device, the 24V supply voltage for the control electronics is too low (100 ms after switch-on U is < 19V). • The current supply voltage is displayed in C00065.	With internal supply voltage via the power electronics, the inverter must be replaced. With external supply voltage, check the correct connection and/or the stability of the supply voltage.

Su06: Mains input overload [xx.0111.00006]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
In order to protect the device from overload, the following device outputs have a hardware detection in the mains input: 7.5 kW, 11 kW, 15 kW, 30 kW, 37 kW, 45 kW. In case of the error message "Su06", this hardware detection has responded.	• Check whether all mains phases are connected (a 2-phase supply may be existent). • Provide for sufficient cooling of the device.

Su07: 24V supply off [xx.0111.00007]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy

oH4: Heatsink temp. > shutdown temp. -5°C [xx.0119.00000]

Response (Lenze setting printed in bold)	**Setting:** C00582 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
The heatsink temperature now only differs by 5 °C from the shutdown temperature of the motor.	Prevent further heating, i.e. reduce motor load or set controller inhibit so that the heatsink can cool down again.

oH1: Heatsink overtemperature [xx.0119.00001]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	**Remedy**
The heatsink temperature is higher than the fixed limit temperature (90 ° C). Maybe the ambient temperature of the controller is too high or the fan or its ventilation slots are dirty.	• Check control cabinet temperature. • Clean filter. • Clean inverter. • If required, clean or replace the fan. • Provide for sufficient cooling of the device.

oH7: Motor temperature resolver >= C121 [xx.0119.00002]

Response (Lenze setting printed in bold)	**Setting:** C00583/3 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☒ **5: Warning** ☑ 6: Information	

Cause	**Remedy**
The motor temperature detected via the resolver interface has reached the warning threshold set in C00121/1.	• Check motor temperature monitoring. • Provide for sufficient cooling of the motor. • Check terminals for open circuit or loose contact.

oH9: Motor overtemperature resolver [xx.0119.00003]

Response (Lenze setting printed in bold)	**Setting:** C00583/1 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
The motor temperature detected via the resolver interface has reached the fixed limit temperature (150 °C).	• Check motor temperature monitoring. • Provide for sufficient cooling of the motor. • Check terminals for open circuit or loose contact.

Sd6: Error thermal detector resolver [xx.0119.00012]

Response (Lenze setting printed in bold)	**Setting:** C00583/5 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
The signals of the encoder connected to the resolver interface are outside the defined operating range of the motor temperature detection.	• Check contacts of the encoder cable at the motor and controller. • Check the selected motor temperature sensor in C01190/1 as to whether it complies with the assembly in the motor. • If required, switch off thermal sensor monitoring (C00583/5="0").

oH3: Motor temperature (X106) triggered [xx.0119.00015]

Response (Lenze setting printed in bold)	**Setting:** C00585 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
The motor temperature monitoring function at the plug connector X106, terminal T1 /T2, has tripped. Possible causes: • The motor is overheated so that the thermal contact integrated into the motor has been switched. • An open circuit or a loose contact at the connections mentioned above has occurred.	• Check motor temperature monitoring. • Provide for sufficient cooling of the motor. • Check terminals for open circuit or loose contact.

oH6: Motor temperature MultiEncoder >= C121 [xx.0119.00020]

Response (Lenze setting printed in bold)	**Setting:** C00583/4 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☒ **5: Warning** ☑ 6: Information	

Cause	Remedy
The motor temperature detected via the encoder interface has reached the warning threshold set in C00121/2.	• Check motor temperature monitoring. • Provide for sufficient cooling of the motor. • Check terminals for open circuit or loose contact.

oH12: Motor overtemperature MultiEncoder [xx.0119.00021]

Response (Lenze setting printed in bold)	**Setting:** C00583/2 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
The motor temperature detected via the encoder interface has reached the fixed limit temperature (150 °C).	• Check motor temperature monitoring. • Provide for sufficient cooling of the motor. • Check terminals for open circuit or loose contact.

Sd12: Error thermal detector MultiEncoder [xx.0119.00022]

Response (Lenze setting printed in bold)	**Setting:** C00583/6 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
The signals of the encoder connected to the encoder interface are outside the defined operating range of the motor temperature detection.	• Check contacts of the encoder cable at the motor and controller. • Check the selected motor temperature sensor in C01190/2 as to whether it complies with the assembly in the motor. • If required, switch off thermal sensor monitoring (C00583/6="0").

oC5: Ixt overload [xx.0119.00050]

Response (Lenze setting printed in bold)	**Setting:** C00604 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☒ **5: Warning** ☐ 6: Information	

Cause	Remedy
The Ixt overload check has tripped. • Operating threshold = 100 % Ixt (adjustable in C00123) Possible causes: • Wrong dimensioning of the device with regard to its motor load. • Load cycles are not complied with.	• Check and, if required, correct dimensioning of the device and the motor load with regard to technical data. • Reduce motor load cycles (observe load cycles according to documentation).

ot1: Maximum torque reached [xx.0123.00001]

Response (Lenze setting printed in bold)	**Setting:** C00608 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
The device indicates that the maximally possible torque at the motor shaft has been reached. • C00057 displays the current torque.	Reduce motor load.

oC7: Motor overcurrent [xx.0123.00007]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The maximum current monitoring function has been triggered. • The instantaneous value of the motor current has exceeded the limit value set in C00939.	Check and, if required, correct dimensioning of the load with regard to the installed device power.

oU: DC bus overvoltage [xx.0123.00014]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☑ 1: Fault ☒ **2: Trouble** ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy

The device has detected an overvoltage in the DC bus. To protect the device hardware, the inverter control is switched off.
• Depending on the configuration of the auto-start lock function, set C00142 so that, when this error is tripped, the inverter only restarts after the controller inhibit has been switched.
• If this error message remains active longer than the time set in C00601, a "Fault" is tripped. Otherwise, the deactivation of the error message causes the inverter control to be enabled again
  • In case of the control types VFCplus and SLVC, the motor voltage is approached to the voltage setpoint alongside a ramp.
  • From version 15.00.00, this voltage ramp can be set in C00983/2. If the described remedies are not possible or do not have any effect, it may be required to increase this voltage ramp as otherwise an overcurrent interruption may be caused. This only happens in case of high motor power and mass inertia so that the Lenze setting of 1 s should be sufficient in the majority of cases.
	• Reduce regenerative load.
	• Use brake resistor.
	• Use a regenerative power supply unit.
	• Establish a DC-bus connection.
	• Select a braking method in C00175 which stops the ramp function generator when reaching the brake chopper threshold ("HlgStop").
	• In case of servo control (SC), set the speed controller parameters correctly.

LU: DC bus undervoltage [xx.0123.00015]

Response (Lenze setting printed in bold)	**Setting:** C00600/1 (☑ Adjustable response)
☐ 0: No Reaction ☑ 1: Fault ☒ **2: Trouble** ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information	

Cause	Remedy
The device has detected a DC bus undervoltage. The inverter control is switched off because the drive properties of the motor control cannot be provided anymore due to the DC bus undervoltage. • Depending on the configuration of the auto-start lock function, set C00142 so that, when this error is tripped, the inverter only restarts after the controller inhibit has been switched.	• Switch on mains supply or ensure sufficient supply via DC bus. • Adjust setting in C00142 if required.

oC1: Power section - short circuit [xx.0123.00016]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The device has recognised a short circuit of the motor phases. To protect the device electronics, the inverter control is switched off. • Mostly, incorrectly executed motor connections are the cause. • If the device is inappropriately dimensioned with regard to the motor load and the current limitation in the controller (Imax controller) is set incorrectly, this error message may also occur. ▸ Motor control: Defining current limits	• Check motor connections and the corresponding plug connector on the device. • Only use permissible combinations of device power and motor power. • Do not set the dynamics of the current limitation controller too high.

oC2: Power section - earth fault [xx.0123.00017]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy

The device has recognised an earth fault at one of the motor phases. To protect the device electronics, the inverter control is switched off.
• Mostly, incorrectly executed motor connections are the cause.
• If motor filter, motor cable length, and cable type (shielding capacity) are dimensioned incorrectly, this error message may occur due to leakage currents to PE.
• If motor filters with additional terminals for +UG and –UG and devices greater or equal 3 kW are used, the earth fault detection may be triggered due to leakage currents to +UG and –UG.
• A cause can also be the use of shielded motor cables longer than 50 m.
	• Check motor connections and the corresponding plug connector on the device.
	• Use motor filters, cable lengths, and cable types recommended by Lenze.
	• If motor filters with additional terminals for +UG and –UG and devices greater or equal 3 kW are used:
	  • Deactivate earth-fault detection during operation by setting the filter time (C01770) to 250 ms.
	• If motor cables longer than 50 m are used:
	  • Increase filter time for earth-fault detection during operation (C01770).

Sd2: Open circuit resolver [xx.0123.00024]

Response (Lenze setting printed in bold)	Setting: C00603/2 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
• Encoder signal interferences (EMC). • Resolver cable interrupted. • Resolver defective.	• Check resolver cable. • Check resolver. • Switch off monitoring (C00603/2 = "0: No reaction") when the resolver is not used.

Sd7: Encoder communication error [xx.0123.00026]

Response (Lenze setting printed in bold)	Setting: C00603/4 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
• Encoder cable interrupted. • Encoder defective.	• Check encoder cable • Check encoder. • Check parameter setting.

Sd4: Open circuit MultiEncoder [xx.0123.00027]

Response (Lenze setting printed in bold)	Setting: C00603/1 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
• Encoder cable interrupted. • Encoder defective. • Supply voltage for the encoder not sufficient.	• Check encoder cable. • Check encoder. • Check the set encoder supply voltage in C00421. Adapt setting to the max. permissible encoder supply voltage. If required, additional voltage drops of the encoder cable have to be taken into consideration. • Switch off monitoring (C00603/1 = "0: No reaction") when the encoder is not used.

Sd9: Status message Hiperface [xx.0123.00028]

Response (Lenze setting printed in bold)	**Setting:** C00603/5 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☒ **6: Information**	

Cause	**Remedy**
Sd9 is output when the controller has received a Hiperface encoder error code from the encoder during initialisation. • The received error codes are displayed in C00491/1...4. The error codes are explained in the data sheet of the respective encoder.	Eliminating the fault described in the error code. Afterwards, reinitialising the encoder by a renewed writing of the encoder type in C00422.

Sd14: Position invalid Hiperface [xx.0123.00029]

Response (Lenze setting printed in bold)	**Setting:** C00603/6 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☒ **6: Information**	

Cause	**Remedy**
No standstill of the Hiperface encoder during initialisation. Hence, the absolute Hiperface track and the incremental SinCos track cannot be initiated exactly and the actual position cannot be detected correctly.	Stopping encoder movement. Afterwards reinitialising the encoder by a renewed writing of the encoder type in C00422.

oC10: Maximum current reached [xx.0123.00030]

Response (Lenze setting printed in bold)	**Setting:** C00609 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
The device displays that the maximum current has been reached.	• Check and, if required, correct dimensioning of the load with regard to the installed device power. • Check the maximum current settings in C00022 (Imax in motor mode) and C00023 (Imax in generator mode).

oC17: Clamp sets pulse inhibit [xx.0123.00031]

Response (Lenze setting printed in bold)	**Setting:** C00569/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
Due to a short overcurrent, the inverter was switched off for a short time (clamp disconnection).	• Check and, if required, correct dimensioning of the load with regard to the installed device power. • Reduce the dynamics of the setpoint change or speed control.

oS1: Maximum speed limit reached [xx.0123.00032]

Response (Lenze setting printed in bold)	**Setting:** C00579 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
The device has recognised that the maximum speed has been reached.	• Limit setpoint selection to maximum values. • Adjust set speed limitation (C00909) and frequency limitation (C00910).

oS2: Max. motor speed [xx.0123.00033]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The device has recognised that the maximally permissible motor speed has been reached.	• Limit setpoint selection to the maximally permissible motor speed. • If required, adapt set maximum motor speed (C00965).

oC18: Current monitoring overload [xx.0123.00034]

Response (Lenze setting printed in bold)	**Setting:** C00584/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
The current monitoring overload has tripped because the apparent motor current has exceeded the switch-off threshold set in C00124/1 for the delay time set in C00563/1C00563/1.	• Reduce overload. • Increase switch-off threshold (C00124/1).

Id2: Motor data identification error [xx.0123.00056]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
During the identification of the motor parameters, an error has occurred that caused an abortion of the identification process. Possible causes: • Interrupted motor cable. • Switched-off power section during the identification. • Implausible start parameter settings.	• Check the motor connections and the corresponding plug connector on the device and, if necessary, the motor terminal box. • Correct start parameters for the motor parameter identification (motor nameplate data). • Stable power supply of the device.

Id1: Motor data identification error [xx.0123.00057]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
During the identification of motor parameters, an error has occurred. Possible causes: • Interrupted motor cable. • Switched-off power section during the identification. • Implausible start parameter settings.	• Check the motor connections and the corresponding plug connector on the device and, if necessary, the motor terminal box. • Correct start parameters for the motor parameter identification (motor nameplate data). • Stable power supply of the device.

Id3: CINH identification [xx.0123.00058]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☒ **4: WarningLocked** ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The device has detected controller inhibit during the motor data identification. • This cancels the identification process. The Lenze setting of the motor data is used.	• Do not set controller inhibit during the motor data identification. • Do not execute any device function which may activate controller inhibit.

Id4: Resistor identification error [xx.0123.00059]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☒ **5: Warning** ☐ 6: Information

Cause	**Remedy**
The device has recognised that an error has occurred in the calculation of the motor cable resistance. • The parameters for cable cross-section and cable length are implausible.	Enter sensible values for cable cross-section and motor cable length.

Id7: Motor control does not match motor data [xx.0123.00060]

Response (Lenze setting printed in bold)	**Setting:** C00571/1 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☒ **6: Information**	

Cause	**Remedy**
At controller enable, the device has detected that the motor control type set in C00006 cannot control the motor type set. • Example: Motor nameplate data for an asynchronous motor have been entered; however, a motor control type for a synchronous motor is set in C00006. **Note:** Since the "VFCplus" control types are able to control every motor to a certain extent, this error message will never occur here.	Enter correct motor nameplate data and set a matching motor control type in C00006: • Motor nameplate data asynchronous motor → motor control type must be ASM, SLVC or VFCplus servo control. • Motor nameplate data synchronous motor → motor control type must be PSM, SLPSM or VFCplus servo control.

Id8: Speed encoder has not been set [xx.0123.00061]

Response (Lenze setting printed in bold)	**Setting:** C00571/2 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
When being In controller enable status, the device has detected that a motor control type with feedback has been set in C00006 , but no speed sensor has been set in C00495.	Set the speed sensor in C00495. **Note:** The error can only be reset if the settings in C00006 and C00495 match.

Sd8: Encoder angular drift monit. [xx.0123.00062]

Response (Lenze setting printed in bold)	**Setting:** C00603/3 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
• Encoder signal interferences (EMC). • Encoder cable interrupted. • Encoder defective. • Faulty parameter setting of the encoder.	• Check encoder cable, use shorter encoder cable if required. • Check encoder. • Check parameter setting. • If required, switch off monitoring (C00603/3).

oC12: I2xt overload - brake resistor [xx.0123.00065]

Response (Lenze setting printed in bold)	**Setting:** C00574/1 (☑ Adjustable response)
☒ 0: No Reaction ☐ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information	

Cause	**Remedy**
Too frequent and too long braking processes.	Check drive dimensioning.

oC19: Brake resistor - short circuit [xx.0123.00066]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Monitoring from version 17.00.00 onwards: • Short circuit at the terminals of the brake resistor. • Low-resistance brake resistor.	• Check terminals of the brake resistor. • Check brake resistor.

oC11: Clamp operation active [xx.0123.00071]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The device indicates that the "CLAMP" overcurrent limitation has been activated. • A permanent clamp operation causes an overload disconnection.	Reduce setpoint generation dynamics or motor load.

Id5: Pole position identification error [xx.0123.00074]

Response (Lenze setting printed in bold)	**Setting:** C00643/1 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
With 360° pole position identification: • The rotor position detected via the encoder system does not comply with the controlled output position (plausibility check). With pole position identification with minimum movement: • The encoder system has detected a movement that is greater than the permitted movement set in C00645/1.	With 360° pole position identification: • Check parameter setting of the pole position identification. • If required, adapt error tolerance for plausibility check in C00645/2. With pole position identification with minimum movement: • If required, adjust permitted movement in C00645/1.

Id6: Resolver ident. error [xx.0123.00075]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
• A controller inhibit was set during resolver error identification. • A time-out occurred while the algorithm was being processed. This error may occur if more than 60 seconds pass by between the setting of the "Resolver error identification" device command and the enable of the controller. • The setpoint speed was too small (|nset| < 500 rpm). • The setpoint speed was not traversed for at least 1 second.	• Reduce the gain of the speed controller. • Check the shielding of the motor and encoder. • Check encoder/encoder connection. • Controller is enabled after the "Resolver error identification" device command has been set. • Ensure that the speed profile is traversed for at least 1 second at constant setpoint speed (|nset| > 500 rpm).

oC13: Maximum current for Fch exceeded [xx.0123.00090]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The device has detected a motor current which exceeds the maximum current limit at permanent switching frequency of the inverter. • If a permanent switching frequency inverter is set, a certain limit arises for the maximum current, depending on the setting. If this current limit is exceeded due to a load impulse or overload, an error message is displayed.	• Observe the maximum current setting depending on the set switching frequency of the inverter. • Reduce the required load or setting of the dynamic switching frequency if necessary.

ot2: Speed controller output limited [xx.0123.00093]

Response (Lenze setting printed in bold)	**Setting:** C00567 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
The output of the speed controller has reached the internal limit value. In this status, the speed controller is not able anymore to correct the system deviation. • Only during "Closed loop" operation or with vector control (SLVC).	• Observe load requirements. • Correct dimensioning or reduce setpoint generation dynamics if necessary. ▶ Motor control

FC01: Switching frequency reduction [xx.0123.00094]

Response (Lenze setting printed in bold)	**Setting:** C00590 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Load-dependent switching frequency reduction	• Observe load requirements. • Correct dimensioning or reduce setpoint generation dynamics if necessary. ▶ Motor control

FC02: Maximum speed for Fchop [xx.0123.00095]

Response (Lenze setting printed in bold)	**Setting:** C00588 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Maximum speed for chopper frequency has been reached. • The maximum speed has been exceeded depending on the switching frequency.	Select the correct maximum speed as a function of the switching frequency. ▶ Motor control: Determine speed limits

oC14: Direct-axis current controller limitation [xx.0123.00096]

Response (Lenze setting printed in bold)	**Setting:** C00570/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Direct-axis current controller limitation is active.	• Observe load requirements. • Correct dimensioning or reduce setpoint generation dynamics if necessary. ▶ Motor control

oC15: Cross current controller limitation [xx.0123.00097]

Response (Lenze setting printed in bold)	**Setting:** C00570/2 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Cross current controller limitation is active.	• Observe load requirements. • Correct dimensioning or reduce setpoint generation dynamics if necessary. • Check parameter setting of the current controller with regard to the motor controllers (e.g. reduce Vp). ▶ Motor control

oC16: Torque controller limitation [xx.0123.00098]

Response (Lenze setting printed in bold)	**Setting:** C00570/3 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Actuator limitation according to speed controller.	• Observe load requirements. • Correct dimensioning or reduce setpoint generation dynamics if necessary. ▶ Motor control

FC03: Field controller limitation [xx.0123.00099]

Response (Lenze setting printed in bold)	**Setting:** C00570/4 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
The output of the field controller has reached its maximum limit value. The drive is at the torque limit in the field weakening range.	• Observe load requirements. • Correct dimensioning or reduce setpoint from the field weakening range if necessary. ▶ Motor control

oC6: I2xt overload - motor [xx.0123.00105]

Response (Lenze setting printed in bold)	**Setting:** C00606 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☒ **5: Warning** ☐ 6: Information	

Cause	Remedy
Thermal overload of the motor.	Only self-ventilated motors can be monitored using the I2xt function. • Check whether is it a self-ventilated motor. If not, set C00606 to "0: No Reaction". • Observe load requirements. • Correct dimensioning if necessary. • For VFCplus control type: Check Vmin boost (C00016). ▶ Set Vmin boost

LP1: Motor phase failure [xx.0123.00145]

Response (Lenze setting printed in bold)	**Setting:** C00597 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Motor phase failure - power section • This error message is displayed if a motor phase carries less current of one half-wave than set in C00599.	• Check the motor connections and the corresponding plug connector on the device and, if necessary, the motor terminal box. • Check the trigger threshold (C00599).

Sd10: Speed limit - feedback system 12 [xx.0123.00200]

Response (Lenze setting printed in bold)	**Setting:** C00607 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
Maximally permissible speed of the feedback system connected to DI1/DI2 reached.	Reduce speed of the rotation shaft/feedback system. $n_{encoder}$ <= (f_{max} x 60) / encoder increments (for f_{max} = 200 kHz)

Sd11: Speed limit for feedback system 67 [xx.0123.00201]

Response (Lenze setting printed in bold)	**Setting:** C00607 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
Maximally permissible speed of the feedback system connected to DI6/DI7 reached.	Reduce speed of the rotation shaft/feedback system. $n_{encoder}$ <= (f_{max} x 60) / encoder increments (for f_{max} = 100 kHz)

Sd3: Open circuit HTL 2-fold or 4-fold [xx.0123.00205]

Response (Lenze setting printed in bold)	**Setting:** C00586 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
• HTL encoder cable interrupted. • HTL encoder is defective. Note: The reason can also be a very dynamic acceleration or an approach against a blocked motor shaft, e.g. with a a closed holding brake or when referencing to positive stop (mode 14/15) and a waiting time (C01223) >100 ms.	• Check HTL encoder cable. • Check HTL encoder. • Check related terminals. • Switch off monitoring (C00586 = "0: No reaction") when the HTL encoder is not used.

Sd13: Inaccuracy SinCos [xx.0123.00206]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☒ **6: Information**

Cause	**Remedy**
Only in case of "Multi-Encoder" speed encoder selection (C00495 = 3): A plausibility check for the digital and analog signals of the absolute value encoder (Hiperface) or sin/cos encoder has responded. • The reason for this are EMC interferences. • This information is given since angle errors (position error) can occur in such a case.	Eliminate EMC interference.

Sd15: Open circuit TL 4-fold [xx.0123.00207]

Response (Lenze setting printed in bold)	**Setting:** C00605/1 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
From version 15.00.00 onwards, Sd15 will be tripped if due to different signal levels (quadruple evaluation of the digital inputs) an open circuit is detected. The quadruple evaluation is activated if "5: encoder signal FreqIn1267" is selected as speed or position encoder (C00495/1 or C00490/0.	• Check HTL encoder cable. • Check HTL encoder. • Check wiring of the input terminals for open circuit or loose contact.

Sd16: Hiperface-SinCos deviation [xx.0123.00208]

Response (Lenze setting printed in bold)	**Setting:** C00603/8 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
The deviation between the position of a Hiperface encoder detected from the evaluation of the SinCos track and the position of the Hiperface track detected in parallel is monitored. From version 15.00.00 onwards, Sd16 is triggered if the electrical deviation is higher than 45° (depending on the number of motor pole pairs). Monitoring is only active as long as the motor speed is below 100 rpm for at least 128 ms. Above this speed, no evaluation is possible.	

Sd17: Encoder supply [xx.0123.00209]

Response (Lenze setting printed in bold)	**Setting:** C00603/9 (☑ Adjustable response)
☐ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information	

Cause	Remedy
The voltage cannot be built up.	• Check encoder connection. • An encoder is connected that is not supported by the device and/or cannot be supplied. • If the supply voltage is too low (e.g. voltage drop due to long encoder cable), adapt the voltage in C00421. • **Important**: A too high voltage can destroy the encoder.

Sd18: V/f emergency operation [xx.0123.00210]

Response (Lenze setting printed in bold)	
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☑ 6: Information	

Cause	Remedy
Error in encoder system	Check all available encoder error messages. Perform the troubleshooting measures for these error messages as described in this manual.

An01: AIN1_I < 4 mA [xx.0125.00001]

Response (Lenze setting printed in bold)	Setting: C00598/1 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☑ 2: Trouble ☒ **3: TroubleQuickStop** ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Open-circuit monitoring for analog input 1 has tripped. • Only if the analog input has been configured as a current loop of 4 ... 20 mA (C00034/1 = 2).	• Check wiring of the analog X3/A1I input terminal for open circuit. • Check minimum current values of the signal sources.

An02: AIN2_I < 4 mA [xx.0125.00002]

Response (Lenze setting printed in bold)	Setting: C00598/2 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☑ 2: Trouble ☒ **3: TroubleQuickStop** ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Open-circuit monitoring for analog input 2 has tripped. • Only if the analog input has been configured as a current loop of 4 ... 20 mA (C00034/2 = 2).	• Check wiring of the analog X3/A2I input terminal for open circuit. • Check minimum current values of the signal sources.

Ab01: Axis bus time-out [xx.0126.00001]

Response (Lenze setting printed in bold)	Setting: C00591/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
The axis bus data error monitoring has been triggered as the monitoring time for data errors set in C02431/3 has been exceeded. • The current number of incorrectly transmitted telegrams is displayed in C02438/1. • The current number of incorrectly received telegrams is displayed in C02438/2. • The current axis bus error status is displayed in C02436.	• Check wiring and bus terminating resistor. • Assign different IDs to nodes. • Eliminate electrical interference (e.g. EMC).

Ab02: Axis bus IO error [xx.0126.00002]

Response (Lenze setting printed in bold)	Setting: C00591/2 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
An axis bus node has put the IO axis bus into the "error" status with parameterised "master/slave" function ("release cord" principle). • In the "Error" status, all nodes start their adjustable response, e.g. a synchronised braking of the drive system or the master only brakes the network to standstill.	Remove the fault of the node which pulled the "release cord". **Note:** The "error" status can only be reset by the node defined as "master".

Ab03: axis bus IO requested [xx.0126.00003]

Response (Lenze setting printed in bold)	
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☑ 6: Information	

Cause	Remedy
Note that Ab02 was requested for this drive.	-

CE04: MCI communication error [xx.0127.00002]

Response (Lenze setting printed in bold)	**Setting:** C01501/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
Communication error with extension module in slot 1.	• Check setting of sync window (C01123) if sync signal source (C01120) is set to "4: MCI". • Eliminate EMC interference. • Switch off inverter, correctly plug in the module, switch on the inverter again. • Switch mains or restart inverter. • Replace module/inverter. • Please contact Lenze if the problem occurs again.

CE0F: MCI control word [xx.0127.00015]

Response (Lenze setting printed in bold)	**Setting:** C00594/2 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Bit 14 ("SetFail") of the wMciCtrl control word of the LS_DriveInterface system block has been set.	Trace back signal source on the bus (e.g. PROFIBUS) that sets bit 14 ("SetFail").

CE4: CAN bus off [xx.0131.00000]

Response (Lenze setting printed in bold)	**Setting:** C00592/2 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
CAN on board: "Bus off" status • Received too many faulty telegrams. • Damaged cable (e.g. loose contact). • Two nodes with the same ID.	• Check wiring and bus terminating resistor. • Set identical baud rate for each bus node. • Assign different IDs to nodes. • Eliminate electrical interference (e.g. EMC).

CA06: CAN CRC error [xx.0131.00006]

Response (Lenze setting printed in bold)	**Setting:** C00592/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
CAN on board: A faulty CAN telegram has been detected.	• Check wiring and bus terminating resistor. • Eliminate electrical interference (e.g. EMC).

CA07: CAN bus warning [xx.0131.00007]

Response (Lenze setting printed in bold)	**Setting:** C00592/3 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
CAN on board: Incorrect transmission or reception of more than 96 CAN telegrams. • The current number of incorrectly transmitted CAN telegrams is displayed in C00372/1. • The current number of incorrectly received CAN telegrams is displayed in C00372/2. • The current CAN error status is displayed in C00345.	• Check wiring and bus terminating resistor. • Set identical baud rate for each bus node. • Assign different IDs to nodes. • Eliminate electrical interference (e.g. EMC).

CA08: CAN bus stopped [xx.0131.00008]

Response (Lenze setting printed in bold)	**Setting:** C00592/4 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
CAN on board: The device has received the "Stop Remote Node" NMT telegram.	Check CAN master (NMT master).

CA0b: CAN HeartBeatEvent [xx.0131.00011]

Response (Lenze setting printed in bold)	**Setting:** C00592/5 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
CAN on board: Cyclic node monitoring • Being a Heartbeat consumer, the device has not received a Heartbeat telegram from Heartbeat producer 1 ... 15 within the defined time. • The current states of the Heartbeat producers are displayed in C00347/1...15.	• Reactivate Heartbeat producers by mains switching, restarting the inverter, or a CAN Reset Node. • Reparameterise CAN Heartbeat producer time or switch off consumer monitoring and reset error status if latched. ▸ Heartbeat protocol

CA0F: CAN control word [xx.0131.00015]

Response (Lenze setting printed in bold)	**Setting:** C00594/1 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	**Remedy**
Bit 14 ("SetFail") in the wCANControl control word of the LS_DriveInterface system block has been set.	Trace back signal source on the CAN bus that sets bit 14 ("SetFail").

CE1: CAN RPDO1 [xx.0135.00001]

Response (Lenze setting printed in bold)	**Setting:** C00593/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
CAN on board: Time monitoring for RPDO1 has been triggered. • RPDO1 has not been received within the monitoring time set in C00357/1 or was faulty.	• Set correct telegram length for CAN master (transmitter). • Eliminate electrical interference (e.g. EMC). • Adjust monitoring time C00357/1 or switch off time monitoring.

CE2: CAN RPDO2 [xx.0135.00002]

Response (Lenze setting printed in bold)	**Setting:** C00593/2 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
CAN on board: Time monitoring for RPDO2 has been triggered. • RPDO2 has not been received within the monitoring time set in C00357/2 or was faulty.	• Set correct telegram length for CAN master (transmitter). • Eliminate electrical interference (e.g. EMC). • Adjust monitoring time C00357/2 or switch off time monitoring.

CE3: CAN RPDO3 [xx.0135.00003]

Response (Lenze setting printed in bold)	**Setting:** C00593/3 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
CAN on board: Time monitoring for RPDO3 has been triggered. • RPDO3 has not been received within the monitoring time set in C00357/3 or was faulty.	• Set correct telegram length for CAN master (transmitter). • Eliminate electrical interference (e.g. EMC). • Adjust monitoring time C00357/3 or switch off time monitoring.

CE5: CAN RPDO4 [xx.0135.00004]

Response (Lenze setting printed in bold)	**Setting:** C00593/4 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
CAN on board: Time monitoring for RPDO4 has been triggered. • RPDO4 has not been received within the monitoring time set in C00357/4 or was faulty.	• Set correct telegram length for CAN master (transmitter). • Eliminate electrical interference (e.g. EMC). • Adjust monitoring time C00357/4 or switch off time monitoring.

CI01: Module missing/incompatible [xx.0140.00013]

Response (Lenze setting printed in bold)	**Setting:** C01501/2 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
The optional communication module has been removed or there is a connection problem or incompatibility with the standard device.	• Check connection between the communication module and standard device. • Check if the module is plugged in correctly. • In case of an incompatibility, either the module or the software of the standard device is out of date. In this case, please contact Lenze.

PS01: No memory module [xx.0144.00001]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☒ **5: Warning** ☐ 6: Information

Cause	**Remedy**
Memory module is either not available or not snapped into place correctly.	• If a memory module has been provided: Plug the memory module into the slot of the standard device intended for this purpose. • If a memory module has been provided: Check if the memory module has been plugged-in correctly.

PS02: Par. set invalid [xx.0144.00002]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The parameter set stored in the memory module is invalid. The reason for this can be as follows: • Incomplete storage of the parameter set due to voltage failure. • The plugged-in module stems from a device with new firmware (compare C00099) or from a different device type (e.g. 8400 BaseLine).	The error can only be removed by loading the Lenze setting with the C00002/1 = "1: On / start" device command. • In order to prevent the error, do not switch off the voltage during the saving process. • If the parameter set is to be transferred from one device with a higher version to a device with a lower version, use the "copy parameter set" function of the keypad. Make sure that you do not use functions that are not available in the older device.

PS03: Par. set device invalid [xx.0144.00003]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The parameter set saved to the memory module is incompatible to the standard device. • An incompatibility of the parameter set is caused e.g. when the memory module of an 8400 HighLine is plugged into an 8400 StateLine or the parameter set in the memory module has a higher version than expected by the standard device. If the parameter set stored in the memory module is compatible with the standard device but has a different (lower) version, this message is only output as "Information". The message can be eliminated by saving the parameter set again. **Note:** If you save the parameter set to a higher device version, you can no longer load this parameter set to a lower device version. ▸ Replacement of the inverter	When the memory modules are replaced, observe the compatibility: • OK: StateLine V2.0 to StateLine V3.0 • OK: StateLine V2.0 to HighLine V2.0 • Not OK: HighLine Vx.x to StateLine Vx.x • Not OK: StateLine V3.0 to StateLine < V3.0

PS04: Par. set Mci invalid [xx.0144.00004]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The parameter set saved to the communication module is incompatible to the standard device. • An incompatibility of the parameter set is caused e.g. when the MCI module parameters in the memory module do not match the plugged communication module.	When the memory modules are replaced, observe the compatibility: • Not OK: Profibus V1.0 to EtherCAT V1.0

PS07: Par. memory module invalid [xx.0144.00007]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The parameter set saved to the memory module is invalid. • The error occurs while loading the parameter set. • The memory module plugged in the device lacks a code or a code is incorrect.	Please contact Lenze.

PS08: Par. device invalid [xx.0144.00008]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The parameter set in the device is invalid. • The error occurs while loading the parameter set. • One code in the device is incorrect.	Please contact Lenze.

PS09: Par. format invalid [xx.0144.00009]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
The code format is invalid. • The error occurs while loading the parameter set.	Please contact Lenze.

PS10: Memory module binding invalid [xx.0144.00010]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device personalisation is active: The binding ID of the memory module does not comply with the binding ID of the inverter.	• Use memory module/inverter with matching binding IDs. • Contact machine manufacturer. **Note:** It is not possible for Lenze to modify a replacement device via special accesses in such a way that it cooperates with a personalised memory module.

PS11: Lenze setting loaded [xx.0144.00011]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy

PS12: Parameter sets loaded [xx.0144.00012]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy

PS13: Parameter sets saved [xx.0144.00013]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy

dF01: FW updated [xx.0145.00001]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy

dF14: SW-HW invalid [xx.0145.00014]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dF15: DCCOM CU2 error [xx.0145.00015]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dF18: BU RCOM error [xx.0145.00024]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dF25: CU RCOM error [xx.0145.00025]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dF26: Appl. watchdog [xx.0145.00026]

Response (Lenze setting printed in bold)	**Setting:** C00580/1 (☑ Adjustable response)
☒ **0: No Reaction** ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☑ 5: Warning ☐ 6: Information	

Cause	Remedy
Time-out of the application. The required computing time of the application exceeds the available computing time.	Reduction of the function block interconnection or the complexity of the application.

dF21: BU watchdog [xx.0145.00033]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dF22: CU watchdog [xx.0145.00034]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dF10: AutoTrip reset [xx.0145.00035]

Response (Lenze setting printed in bold)	**Setting:** C00189 (☑ Adjustable response)
☑ 0: No Reaction ☒ **1: Fault** ☑ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
Too frequent auto-trip reset.	• Check the error cause that activates the auto-trip reset. • Eliminate error cause and reset (acknowledge) error manually afterwards.

dF50: Retain error [xx.0145.00050]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
An error has occurred when accessing retain data. • Either caused by an internal hardware error or by lack of mains switching after a firmware download.	Mains switching • Please contact Lenze if the problem occurs again.

dF51: CuCcr error [xx.0145.00051]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Mains switching • Please contact Lenze if the problem occurs again.

dF52: BuCcr error [xx.0145.00052]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Mains switching • Please contact Lenze if the problem occurs again.

Ck01: Pos. HW limit switch [xx.0184.00001]

Response (Lenze setting printed in bold)	**Setting:** C00595/1 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☒ **3: TroubleQuickStop** ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
MCK: The hardware limit switch in positive traversing direction has tripped. • The *bLimitSwitchPos* input for travel range monitoring via positive hardware limit switch has been set to FALSE (fail-safe).	Reset error message and retract limit switch.

Ck02: Neg. HW limit switch [xx.0184.00002]

Response (Lenze setting printed in bold)	**Setting:** C00595/2 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☒ **3: TroubleQuickStop** ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
MCK: The hardware limit switch in negative traversing direction has tripped. • The *bLimitSwitchNeg* input for travel range monitoring via negative hardware limit switch has been set to FALSE (fail-safe).	Reset error message and retract limit switch.

Ck15: Error status sign. brake [xx.0184.00005]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☒ **3: TroubleQuickStop** ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
MCK: The status monitoring of the holding brake control has tripped.	• Check configuration of the *bMBrakeApplied* input for status detection of the brake (via a switching contact at the brake). • Check wiring/function of the switching contact. • Adapt waiting time (C02589/3). • Deactivate status monitoring (via bit 5 in C02582).

Ck03: Pos. SW limit position [xx.0184.00007]

Response (Lenze setting printed in bold)	**Setting:** C00595/3 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☒ **3: TroubleQuickStop** ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
MCK: The device has detected that the position is outside the positive software limit position (C01229/1).	• Increase permissible traversing range (change setting of the software limit positions). • Deactivate software limit position monitoring.

Ck04: Neg. SW limit position [xx.0184.00008]

Response (Lenze setting printed in bold)	**Setting:** C00595/4 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☒ **3: TroubleQuickStop** ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
MCK: The device has detected that the position is outside the negative software limit position (C01229/2).	• Increase permissible traversing range (change setting of the software limit positions). • Deactivate software limit position monitoring.

Ck17: Direction conflict Ccw [xx.0184.00009]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☒ **6: Information**

Cause	**Remedy**
Positioning profile is started in positioning mode 5 or 12 (absolute (Ccw) or absolute (Ccw) to TP); however, the Cw rotating direction is defined by the profile generation.	Start positioning with zero speed.

Ck18: Direction conflict Cw [xx.0184.00010]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☐ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☒ **6: Information**

Cause	**Remedy**
Positioning profile is started in positioning mode 4 or 11 (absolute (Ccw) or absolute (Ccw) to TP); however, the Ccw rotating direction is defined by the profile generation.	Start positioning with zero speed.

Ck14: Target position outside SW limit position [xx.0184.00015]

Response (Lenze setting printed in bold)	**Setting:** C00595/14 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☒ **4: WarningLocked** ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
MCK: It has been attempted to position a target outside the software limit positions (C01229/1 and C01229/2).	• Select a target within the software limit positions. • Increase permissible traversing range (change setting of the software limit positions). • Deactivate software limit position monitoring.

Ck16: Time overflow manual operation [xx.0184.00064]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	**Remedy**
PC manual control: The connection monitoring has tripped. • The online connection between the PC and the inverter has been interrupted for a longer period of time than the timeout set in C00464/1.	• Check communication link between PC and inverter. • Check voltage supply/function of the inverter. • Adjust the timeout (C00464/1).

Ck05: Following error 1 [xx.0184.00153]

Response (Lenze setting printed in bold)	Setting: C00595/5 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☒ **5: Warning** ☑ 6: Information	

Cause	Remedy
MCK: Following error limit 1 (C01215/1) has been exceeded.	• Optimise control mode. • Increase following error limit. • Deactivate following error monitoring.

Ck06: Following error 2 [xx.0184.00154]

Response (Lenze setting printed in bold)	Setting: C00595/6 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☒ **5: Warning** ☑ 6: Information	

Cause	Remedy
MCK: Following error limit 2 (C01215/2) has been exceeded.	• Optimise control mode. • Increase following error limit. • Deactivate following error monitoring.

Ck07: Travel range limit exceeded [xx.0184.00155]

Response (Lenze setting printed in bold)	Setting: C00595/7 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☒ **3: TroubleQuickStop** ☑ 4: WarningLocked ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
MCK: The maximum travel distance has been exceeded. • The maximum travel distance is displayed in C01213/1.	• Check profile parameters. • Deactivate travel range limit monitoring.

Ck08: Home position unknown [xx.0184.00156]

Response (Lenze setting printed in bold)	Setting: C00595/8 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☒ **4: WarningLocked** ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
MCK: Home position is unknown.	Perform homing.

Ck09: Positioning mode invalid [xx.0184.08005]

Response (Lenze setting printed in bold)	Setting: C00595/9 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☒ **4: WarningLocked** ☑ 5: Warning ☑ 6: Information	

Cause	Remedy
MCK: The positioning mode defined via the *wPosProfileMode* input is not supported.	Define a valid positioning mode.

Ck10: Implausible profile data [xx.0184.08007]

Response (Lenze setting printed in bold)	**Setting:** C00595/10 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☒ **4: WarningLocked** ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**

Cause 1: The profile data results in a braking distance that is longer than the distance to be travelled.
• a: Occurs if a small specified distance within a few increments is to be positioned with final speed and S rounding.
• b: It has been detected that the braking distance required for sizing the profile phases correctly regarding the S rounding and overchange is longer than the selected distance in the first cycle when *bExecute* = TRUE.
Cause 2: Reversing process in overchange profile linkage
• If the linkage of two profiles with final speed causes a reversing process in the second profile and this is to smooth another sequence profile with final speed too, the error is set and ramped down to standstill.
Cause 3: Final speed at overchange is higher than maximum profile speed
Cause 4: Traversing, accelerating or braking is not possible due to 0-parameters for speed, acceleration or deceleration.
	On cause 1a:
	• Set the S rounding to zero for the respective profile and go to the next profile with final speed zero.
	On cause 1b:
	• The error can be avoided with C02868/Bit02!
	On cause 2:
	• The continuous profile linkage with overchange must not provide any reversing process due to the profile selection. Here, the profile which causes a reversing must be defined with the final speed 0.
	• From version 16.00.00 onwards, the following modes with final speed are possible via C02868/Bit02:
	  • Reversing
	  • Profiles where the final speed cannot be reached
	On cause 3:
	• Profiles in an overchange profile linkage with final speeds higher than their max. profile speeds are not supported. The final speed of a profile is limited internally to the max. profile speed in the profile data set.
	• The error can be avoided with C02868/Bit02!
	On cause 4:
	• Ensure that the corresponding profile parameters for speed, acceleration and deceleration are set to non-zero when starting a traversing process.

Ck11: Invalid operating mode [xx.0184.08009]

Response (Lenze setting printed in bold)	**Setting:** C00595/11 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☒ **5: Warning** ☑ 6: Information	

Cause	**Remedy**
MCK: The operating mode defined via the *wMckOperationMode* input is not supported.	Define a valid operating mode.

Ck12: Invalid profile number [xx.0184.08014]

Response (Lenze setting printed in bold)	**Setting:** C00595/12 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☒ **4: WarningLocked** ☑ 5: Warning ☑ 6: Information	

Cause	**Remedy**
MCK: The positioning profile number in the positioning operating mode specified via the *wPosProfileNo* input is invalid.	Define a valid profile number.

Ck13: Error - MCKCtrlInterface function block [xx.0184.08015]

Response (Lenze setting printed in bold)	**Setting:** C00595/13 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☑ 3: TroubleQuickStop ☑ 4: WarningLocked ☒ **5: Warning** ☑ 6: Information	

Cause	**Remedy**
MCK: An error in the L_MckCtrlInterface_1 function block has occurred.	Check the configuration and parameterisation of the L_MckCtrlInterface_1 FB. In this regard, also observe the status messages of the FB (*wFailState* or C01299 output).

dH09: EEPROM power section [xx.0400.00009]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dH10: Fan failure [xx.0400.00016]

Response (Lenze setting printed in bold)	**Setting:** C00566 (☑ Adjustable response)
☑ 0: No Reaction ☑ 1: Fault ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☒ **5: Warning** ☐ 6: Information	

Cause	Remedy
The device fan has failed. Possible causes: • The short-circuit check of the fan connection has tripped. • The speed monitoring of the fan has tripped.	• Check the fan for short-circuit. • Clean the fan.

dH68: Adjustment data error CU [xx.0400.00104]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dH69: Adjustment data error BU [xx.0400.00105]

Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

dH70: ControlUnit is unequal to BaseUnit [xx.0400.00106]

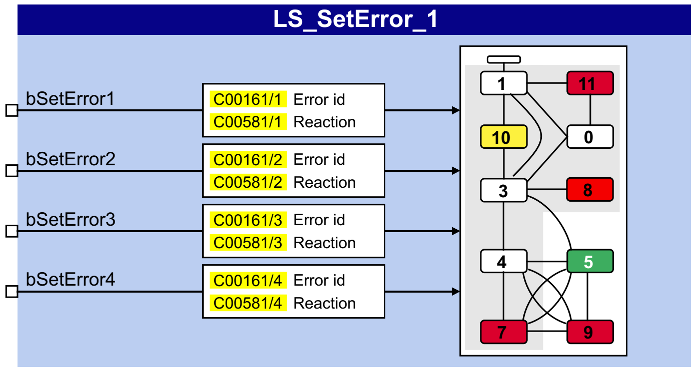
Response (Lenze setting printed in bold)
☐ 0: No Reaction ☒ **1: Fault** ☐ 2: Trouble ☐ 3: TroubleQuickStop ☐ 4: WarningLocked ☐ 5: Warning ☐ 6: Information

Cause	Remedy
Device error	Please contact Lenze.

10.10 "LS_SetError_1" system block

This system block is used to implement error handling within the application.

- The application can trip up to four different user error messages with parameterisable error IDs and error responses via the four boolean inputs.
- If several inputs are set to TRUE at the same time, the input with the lowest number will trip the error message.



inputs

Designator Data type	Information/possible settings
bSetError1 BOOL	Input for tripping "User error 1" • Error subject number: 980 • Error number: (C00581/1 x 0x0400000) + (980 x 0x10000) + (C00161/1)
bSetError2 BOOL	Input for tripping "User error 2" • Error subject number: 981 • Error number: (C00581/2 x 0x0400000) + (981 x 0x10000) + (C00161/2)
bSetError3 BOOL	Input for tripping "User error 3" • Error subject number: 982 • Error number: (C00581/3 x 0x0400000) + (982 x 0x10000) + (C00161/3)
bSetError4 BOOL	Input for tripping "User error 4" • Error subject number: 983 • Error number: (C00581/4 x 0x0400000) + (983 x 0x10000) + (C00161/4)

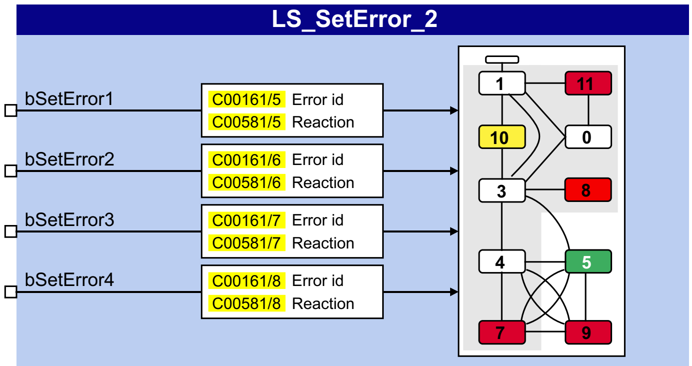
Parameters

Parameters	Possible settings			Info
C00161/1...4	0	...	65535	Error ID for user errors 1 ... 4
C00581/1...4				Response to user errors 1 ... 4
	0	No Reaction		
	1	Fault		
	2	Trouble		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	Warning		
	6	Information		

10.11 "LS_SetError_2" system block

This system block is used to implement error handling within the application.

- The application can trip up to four different user error messages with parameterisable error IDs and error responses via the four boolean inputs.
- If several inputs are set to TRUE at the same time, the input with the lowest number will trip the error message.



inputs

Designator Data type	Information/possible settings
bSetError1 BOOL	Input for tripping "User error 5" • Error subject number: 984 • Error number: (C00581/5 x 0x0400000) + (984 x 0x10000) + (C00161/5)
bSetError2 BOOL	Input for tripping "User error 6" • Error subject number: 985 • Error number: (C00581/6 x 0x0400000) + (985 x 0x10000) + (C00161/6)
bSetError3 BOOL	Input for tripping "User error 7" • Error subject number: 986 • Error number: (C00581/7 x 0x0400000) + (986 x 0x10000) + (C00161/7)
bSetError4 BOOL	Input for tripping "User error 8" • Error subject number: 987 • Error number: (C00581/8 x 0x0400000) + (987 x 0x10000) + (C00161/8)

Parameters

Parameters	Possible settings			Info
C00161/5...8	0	...	65535	Error ID for user errors 5 ... 8
C00581/5...8				Response to user errors 5 ... 8
	0	No Reaction		
	1	Fault		
	2	Trouble		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	Warning		
	6	Information		

11 Oscilloscope function

This function extension is available from version 12.00.00!

The oscilloscope function integrated in the 8400 TopLine can be used as support for commissioning, maintenance, and troubleshooting. It is operated via a user interface in the engineering tool.

Typical applications

- Graphic display of measured variable (e.g. speed setpoint, actual speed value and torque)
- Detection of process values without additional measuring instruments (e.g. oscilloscope, voltmeter and ammeter)
- Convenient documentation for fine tuning of control circuits or parameter changes of the inverter

Special features

- Recording and saving of measured values in the 8400 TopLine
- Simultaneous measuring on eight independent channels
- Measuring fast and slow signals by means of adjustable sample rate
- Triggering on a channel, an application variable or system event
- Detecting measured values before and after the trigger event
- Transferring measured values to the Engineering PC for the purpose of graphic display and evaluation in the engineering tool
- The measured values represented in the form of interpolated curves can be optionally shown and hidden, represented in any colour or overlaid with the signal characteristics of other variables recorded.
- Cursor and zoom function for the measurement analysis
- Saving & loading oscilloscope configurations on the Engineering PC
- Export of measured values via the clipboard for further processing
- Linking channel values with arithmetic operations (addition, subtraction and multiplication)
- Simple signal analysis by frequency transformation of time signals with FFT ("Fast Fourier Transformation").

Functional description

When an online connection to the 8400 TopLine has been established, use the oscilloscope user interface of the engineering tool to set the trigger condition and the sample rate and select the signal sources to be recorded. Here, "signal sources" are the internal output signals of the function, system, application and port blocks.

The changed configuration is only then taken over by the device when pressing the "start" button in the toolbar.

The values are validated after each input acknowledgement by the "Return" key. If the check shows invalid settings, the oscilloscope triggers an error.

With an online connection, the measured 8400 TopLine values are transferred to the engineering tool and graphically represented on the oscilloscope user interface as soon as the measurement has been completed.

11.1 Technical data

Oscilloscope function of 8400 TopLine	
Number of channels	1 ... 8
Depth of the measured value memory	Max. 8192 measured values, depending on the number of channels and the size of the signal sources to be recorded
Data width of a channel	Max. 32 bits, corresponding to the data type of the signal sources to be recorded
Sample rate	1 ms or a multiple thereof
Time base	5, 10, 20, 50, 100, 200, 500 ms or 1 s
Trigger level	Corresponding to the value range of the signal sources to be triggered
Trigger selection	The trigger is activated if the trigger value set for the respective channel is fallen short of or exceeded. The trigger value "must actually pass" the threshold.
Trigger delay	- 200 % ... + 200 %
Trigger source	Channel 1 ... 8

11.2 Operation

This chapter describes step-by-step how to record the signal characteristics of 8400 TopLine signal sources and represent, analyse, document and process them in the oscilloscope.

Note!

Recording can only be started when an online connection has been established to the 8400 TopLine.

11.2.1 User interface

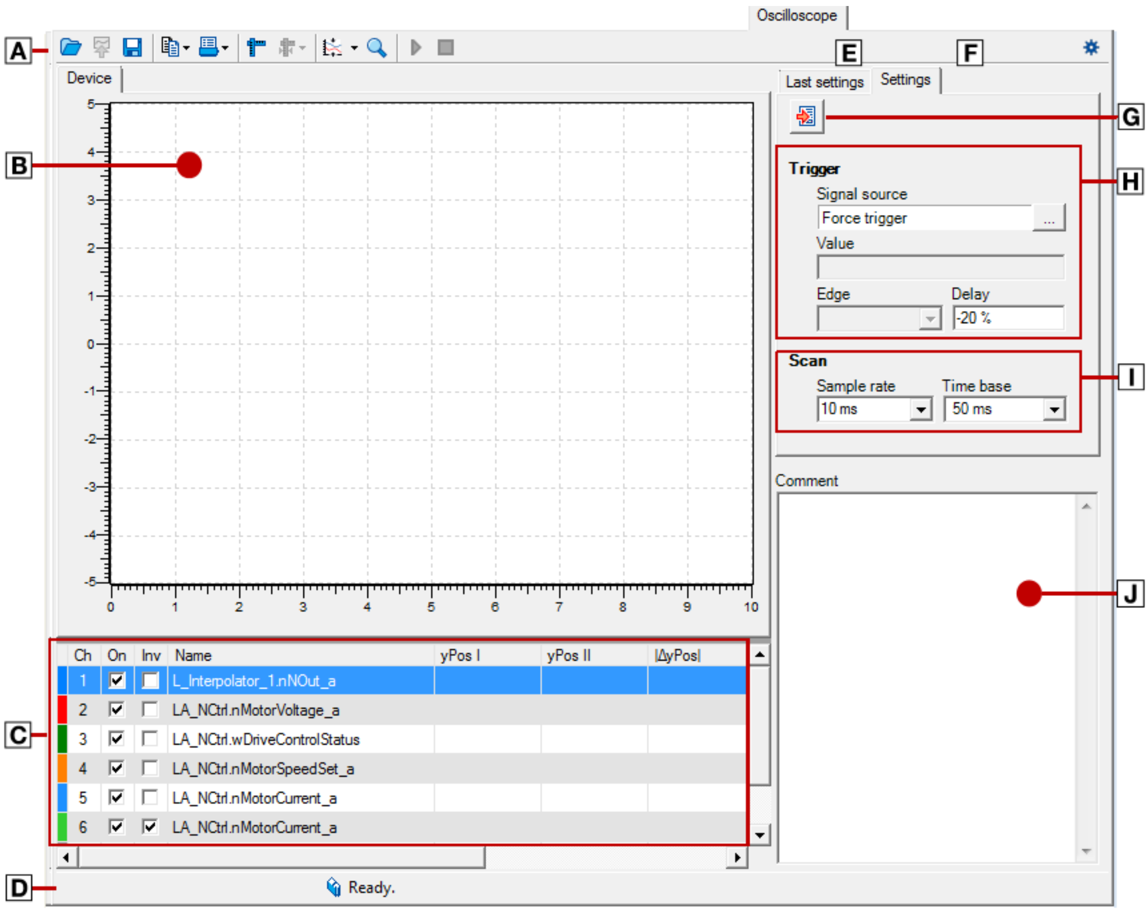
The oscilloscope user interface is available in the following Lenze engineering tools:

☑ »Engineer« from version 2.16	☐ »PLC Designer«	☑ »EASY Starter« from version 1.9

How to go to the oscilloscope user interface:

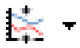
1. Go to the *Project view* and select the servo inverter.
2. Select the **Oscilloscope** tab from the *Workspace*.

The oscilloscope user interface contains the following control and function elements:



Ⓐ Oscilloscope toolbar
Ⓑ Oscillogram selection
Ⓒ Channel list
▸ Selecting the signal sources to be recorded
Ⓓ Status bar
Ⓔ Last settings
Ⓕ Settings
Ⓖ Import settings from a loaded oscillogram
Ⓗ Trigger settings
Ⓘ Input fields for sampling rate and time base
Ⓙ Input field for comments

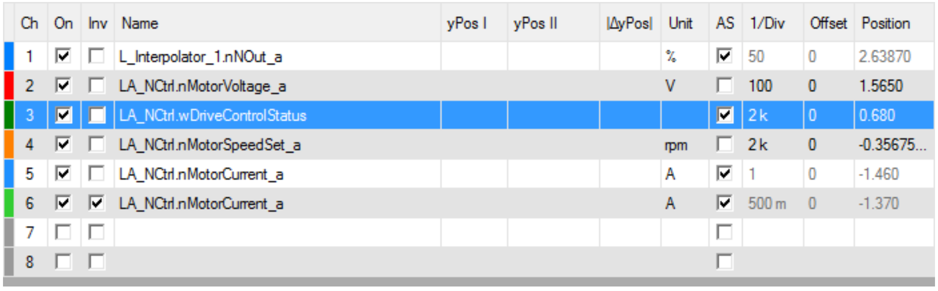
Oscilloscope toolbar

Symbol	Function
	Loading the oscillogram file (📖 801)
	Upload recorded oscillogram from device • Transmit values from the measured value memory of the 8400 TopLine to the Engineering PC. • Only possible when an online connection has been established to the 8400 TopLine.
	Saving the oscillogram in a file (📖 800)
	Copy to clipboard: Copy as text | Copy as picture | Copy as table • For documentation purposes, it is possible to copy the measured value of an oscilloscope as a table or, alternatively, the oscilloscope user interface as a picture, to the clipboard for use in other programs.
	Printer settings | Print view | Print
	Show cursor
	Automatically scale vertically • Set all Y positions to zero
	Activate zoom function ▸ Adjusting the representation (📖 793)
	Start recording • The settings are transferred to the device and the recording is started.
	Stop recording
	Oscilloscope settings • Cyclic recording of oscillograms • Always load oscillograms after recording is completed without a query

11.2.2 Selecting the signal sources to be recorded

The oscilloscope supports up to eight channels. Thus, the **channel list** can record maximally eight signal sources.

The **channel list** serves to configure the signal sources to be recorded. Four signal sources have already been selected in the default setting:



Ch	On	Inv	Name	yPos I	yPos II	|ΔyPos|	Unit	AS	1/Div	Offset	Position
1	☑	☐	L_Interpolator_1.nNOut_a				%	☑	50	0	2.63870
2	☑	☐	LA_NCtrl.nMotorVoltage_a				V	☐	100	0	1.5650
3	☑	☐	LA_NCtrl.wDriveControlStatus					☑	2 k	0	0.680
4	☑	☐	LA_NCtrl.nMotorSpeedSet_a				rpm	☐	2 k	0	-0.35675...
5	☑	☐	LA_NCtrl.nMotorCurrent_a				A	☑	1	0	-1.460
6	☑	☑	LA_NCtrl.nMotorCurrent_a				A	☑	500 m	0	-1.370
7	☐	☐						☐			
8	☐	☐						☐			

Name	Meaning
-	Curve colour for representation in the oscillogram • A double-click on the colour area of the channel serves to set a user-defined colour.
Ch	Channel number
On	Cam visible / invisible
Inv	Inversion yes / no
Name	Name of the signal source
yPos I	y position of cursor I
yPos II	y position of cursor II
|ΔyPos|	Difference of the y positions of both cursors • Difference = |yPos I - yPos II|
Unit	Unit of the signal source
AS	Select/deselect channel for automatic scaling
1/Div	Vertical scaling factor
Offset	Offset value • The offset value is subtracted from the recorded raw value before scaling is executed. This serves, for instance, to make very slight value fluctuations visible within one constantly very high recording value (e.g. harmonics with low amplitude).
Position	Position value • The position value determines the vertical position of the zero point of the y axis of a curve with regard to the vertical curve scale (- 5 ... + 5).

How to select a signal source for recording:

1. Double-click a non-assigned line in the **channel list** to open the *Select signal source* dialog box.
   - Double-clicking an already pre-assigned selection enables you to assign it with another signal source.
2. Select a new variable in the *Select signal source* dialog box.
3. Click the **OK** button.
   - The dialog box is closed and the selection is accepted.

How to delete a selection:

1. Go to the **channel list** and click the signal source to be removed.
2. Right-click the *context menu* to open it.
3. Select the **Delete signal source** command in the *context menu*.

You can add so many signal sources for recording until all eight channels are assigned.

11.2.3 Selecting the recording time/sample rate

How to define the sampling rate and time base for recording:

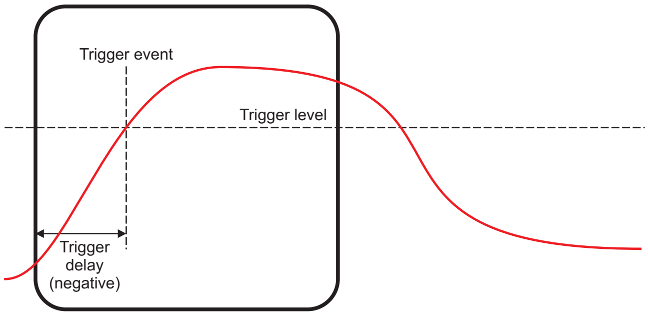
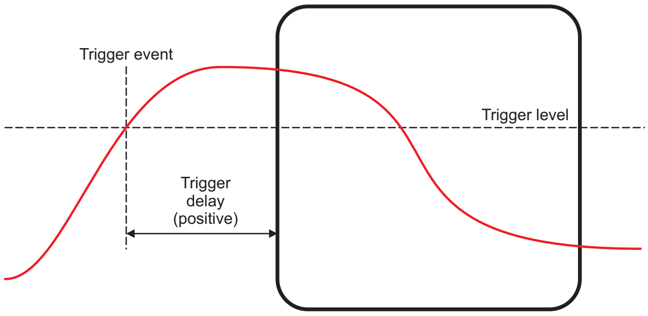
1. Select the desired time base from the **time base** list field.
   - The current time base setting multiplied by ten results in the recording time.
   - Since the measured value memory of the 8400 TopLine has a limited capacity, usually a compromise is made between sample rate and recording time.
2. Enter the desired sampling rate in [ms] in the **sampling rate** input field.

11.2.4 Defining the trigger condition

The trigger condition serves to define the starting time of recording in the 8400 TopLine. The oscilloscope provides various trigger conditions by means of which recording of the measured values can be controlled.

If the **Last settings** tab is in the foreground, click the **Settings** tab to show the input fields for configuring the trigger condition. The **Settings** tab contains the button which serves to import settings from a loaded oscillogram.

setting	Function
Signal source	Selection of the trigger source
Variable	Selection of an application variable as signal source
Channel	The oscilloscope triggers on a channel configured in the **channel list**.

setting	Function
System event	The following system events serve as trigger source: • Fault, • Warning • Error
Direct trigger	No trigger condition, recording takes place immediately after clicking the symbol ▶ in the *oscilloscope toolbar*.
Value	Value from which on triggering is activated.
Deceleration	Delay between recording and trigger event. Unit: [%]
Trigger delay	Select a negative delay time to detect signals **prior to** the trigger event.  • In the oscillogram, the trigger time is marked by a dashed line. • When triggering on occurrence of an event, it is thus possible to detect the values that have caused the event.
	Select a positive delay time to detect signals occurring a certain time **after** the trigger event. 
Edge	Three trigger types are available:
Positive edge	First, the selected trigger value must be fallen below and then exceeded in order that the trigger is activated.
Negative edge	First, the selected trigger value must be exceeded and then fallen below in order that the trigger is activated.
Change	For triggering on a Boolean signal source: • Trigger activation requires a state change. For triggering on a different signal source: • The current value must be different than the last value in order that the trigger is activated.

11.2.5 Starting recording

Note!

Recording can only be started when an online connection has been established to the 8400 TopLine.

Go to the *oscilloscope toolbar* and click the ▶ button to activate recording. Recording starts as a function of the settings for triggering the signal source.

To obtain a maximum sample rate when recording the signal source values, the data are first stored in the measured value memory of the 8400 TopLine and then transferred to the Engineering PC. The current recording status is displayed in the status bar.

11.2.6 Cyclic recording

How to record oscillogram cyclically:

1. Click the symbol in the *oscilloscope toolbar* to open the **Oscilloscope settings** dialog box.
2. In order that the recording process is restarted automatically after the upload of an oscillogram, set the checkmark accordingly.
   - Cyclic recording is only possible for time base values ≥ 500 ms.

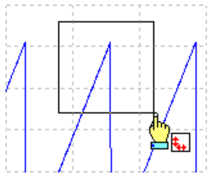
For monitoring certain situations, this serves, for instance, to obtain the increased view of the interesting part of a characteristic even after the cyclic update, as originally zoomed.

11.2.7 Adjusting the representation

After the variable values have been recorded and the oscillogram has been transferred to the PC, it is visualised. If required, the representation can now be adjusted by using the zoom or the automatic scaling function.

Zoom function

Go to the *oscilloscope toolbar* and click the icon to activate the zoom function.

Zoom function	Procedure	
Zoom selection		Hold down the left mouse button and draw the oscillogram section to be zoomed:  • While being drawn, the selection is shown with a frame. • When the left mouse button is released, the selection is zoomed in the oscillogram.
Horizontal stretching		Hold down the left mouse button and move the mouse pointer on the horizontal scale to the left to stretch the shown selection from the right edge.
		Hold down the right mouse button and move the mouse pointer on the horizontal scale to the right to stretch the shown selection from the left edge.
	Moving the mouse pointer in opposite direction continuously reduces the stretching.	
Vertical stretching		Hold down the left mouse button and move the mouse pointer on the vertical scale to the bottom to stretch the shown selection from the top.
		Hold down the right mouse button and move the mouse pointer on the vertical scale to the top to stretch the shown selection from the bottom.
	Moving the mouse pointer in opposite direction continuously reduces the stretching.	
Return to original representation		Click the right mouse button in the oscillogram to return step by step to the original representation.

Automatic scaling function

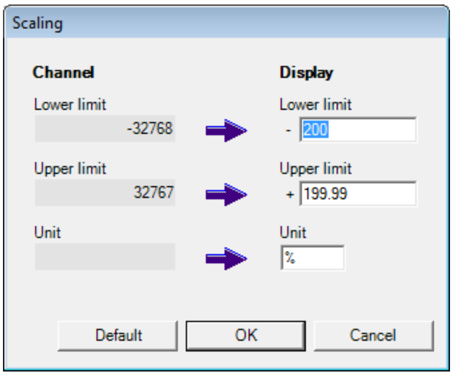
Use the automatic scaling function to automatically scale and reposition the representation of selectable signal characteristics in the oscillogram and reset the offset to "0".

How to carry out automatic scaling:

1. Activate the automatic scaling for each channel in the **channel list** by a checkmark in the "AS" column.
2. Go to the *oscilloscope toolbar* and click the icon to activate the automatic scaling function for the activated channels.
3. Click the **OK** button.
   - The dialog box is closed and the selected channels/signal sources are scaled automatically.
4. Go to the *oscilloscope toolbar* and click the arrow next to the symbol to set all the displayed curves to the Y position "0". This way, the curves are displayed "above one another".

How to carry out the scaling manually for each channel:

1. Go to the **channel list** and click the entry to be changed in the "unit" column (double-clicking an empty line has no effect.)
   - The dialog for entering the scaling opens which permits the entry of the lower and upper limit and the entry of the unit as shown in the example:



2. Enter the lower limit, the upper limit and the unit.
   - On the right, you can enter any scaling values as upper and lower limits.
   - On the left side of the dialog, the limits of the data type of the recorded value are displayed.
3. Click the **OK** button.
   - The dialog box is closed.

11.2.8 Cursor function: Reading individual measured values

In addition to the zoom and scaling function, the oscilloscope offers a "cursor function" that can be used to display individual measured values of a selectable channel or the difference between two measured values.

How to use the cursor function:

1. Go to the *oscilloscope toolbar* and click the icon to activate the cursor function.
   - Another button is displayed which enables a centering of two vertical measuring lines that are independent of each other and can be moved.
   - The status bar displays the position of both measuring lines and the difference between them.
2. Select the channel for which individual measured values are to be indicated from the **Channel** list field.

3. Hold down the left mouse button and drag the red vertical measuring line to the desired position.
    - The active measuring line is represented by a continuous line, the inactive measuring line is represented by a dashed line.
    - If you click the inactive measuring line, it automatically becomes active.
    - The value measured at the position of the active measuring line is indicated in the value group box.
    - The difference between the values measured at the two measuring lines is indicated in the Differential value group field.
    - Comparing peak values: Several values displayed in the oscillogram can be compared by means of a horizontal measuring line. This measuring line is automatically generated based on the current cursor position and thus cannot be moved separately.

11.2.9 Automatic recording directly after mains connection

This function expansion is available from version 21.00.00.

Prerequisite

- The oscilloscope is configured using the Lenze Engineering Tool (e.g. EASY Starter).
- The user must configure the individual channels and the trigger conditions.

How to start the recording immediately after connection to the mains:

1. Set C00759/0 = 1
   - The oscilloscope is automatically prepared after connection to the mains. Recording is triggered once the selected trigger condition is reached.
2. The device command C00002/10 = 1 is used to save the configuration in the memory module with mains failure protection.

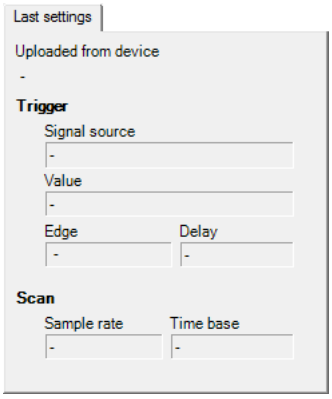
   Now the internal oscilloscope is sufficiently prepared and the mains supply can be switched off.

The next time the device is connected to the mains, the oscilloscope integrated in the device will start automatically and begins measurement upon detecting the trigger event.

The recorded oscillogram is loaded from the device and displayed on the **Oscilloscope** tab.

11.2.10 Last settings

All information included in the **Last settings** tab refer to the oscillogram loaded into the device:



The contents cannot be changed.

11.3 Managing oscillograms

If several oscillograms are loaded in the oscilloscope at the same time, the oscillogram to be displayed is selected via the corresponding tab below the toolbar. In general, the following oscillograms are to be distinguished:

Device oscillogram

The device oscillogram is the only oscillogram which can be used to establish a connection to the target system to carry out an oscilloscope measurement.

MERGE oscillogram

If two or more oscillograms are loaded in the oscilloscope, a "MERGE" tab is available.

- In the merge tab, several characteristics from the currently loaded data records can be overlaid, e.g. to compare signal characteristics from different recordings. ▶ Overlay function (📖 802)

Loaded oscillogram

An oscillogram loaded from a file.

11.3.1 Commenting the oscillogram

The **Comment** text field serves to enter a comment on the selected oscillogram.

- If you execute the Saving the oscillogram in a file command, the comment is saved together with the oscillogram in the file.

11.3.2 Saving the oscillogram in a file

After the signal sources to be recorded have been selected and the required settings have been entered, you can save the configuration and recording, if already executed, for future use in the project or export them to a file.

Note!

The reuse of a saved configuration is only reasonable for devices of the same type, as otherwise due to a scaling of the oscilloscope channels that is not adapted, incorrect values are displayed!

How to save an oscillogram in the project:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Open oscillogram file* dialog box appears.
2. Specify a file name in the **File name** input field.
3. Click the **Filing in the project** button.
   - The dialog box is closed and the current oscillogram is filed in the project.

Note!

The oscillogram is only saved if the entire project is saved!

How to save an oscillogram as external file:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Open oscillogram file* dialog box appears.
2. Press the **Save as external file ...** button.
   - A new window opens in which the directory and the file name for the oscillogram to be saved have to the specified.
3. Click the **Save** button.
   - The dialog box is closed and the current oscillogram is saved.

11.3.3 Loading the oscillogram file

Configurations/oscillograms already saved can be reloaded into the oscilloscope any time, e.g. for the overlay function.

Note!

The reuse of a saved configuration is only possible for devices of the same type, as otherwise due to a scaling of the oscilloscope channels that is not adapted, incorrect values are displayed!

How to load an oscillogram file from the project:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Load oscillogram file* dialog box appears.
2. Select the file to be loaded from the upper list field.
3. Click the **OK** button.
   - The dialog box is closed and the oscillogram file is accepted.

How to load an oscillogram file from an external file:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Load oscillogram file* dialog box appears.
2. Press the **Load from external file...** button.
   - A new window opens in which the directory and the file name for the oscillogram file to be loaded have to be selected.
3. Click **Open**.
   - The dialog box is closed and the oscillogram file is loaded.
   - The oscillogram is displayed on an additionally appearing tab.
   - If the configuration to be loaded contains signal sources that are no longer available in the device, these variables are automatically removed from the configuration.

11.3.4 Overlay function

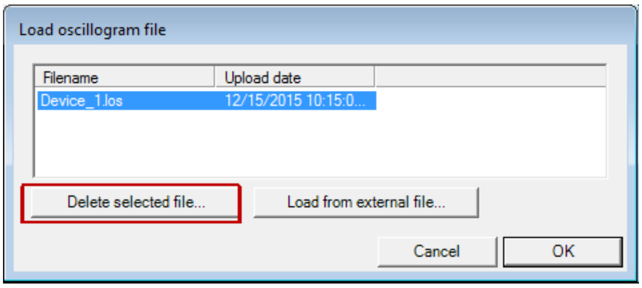
The overlay function serves to lay several characteristics from the currently loaded oscillogram files on top of each other, e.g. to compare signal characteristics from different recordings.

- If two or more oscillograms are loaded in the oscilloscope, e.g. the device oscillogram and an oscillogram previously saved in the project, a "MERGE" tab is available.
- If the MERGE tab is selected, the desired characteristics to be overload or compared can be selected from the loaded files in the **channel list**.
- If a device oscillogram is used in the merge tab, an update is carried out in the MERGE oscillogram in case of a renewed recording.
- Removing signal sources from the device oscillogram causes the characteristics in the MERGE oscillogram to be deleted.

11.3.5 Deleting an oscillogram file saved in the project

How to delete an oscillogram file saved in the project:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Load oscillogram file* dialog box appears.



2. Select one or several oscillogram files from the upper list field.
3. Press the **Delete selected file...** button.
   - The selected file(s) is/are deleted and the dialog box is closed.

12 System bus "CAN on board"

The inverter has an integrated CANopen interface ("CAN on board") which is used to exchange i.a. process data and parameter values between the nodes. Furthermore, other modules can be connected via this interface such as decentralised terminals, operator and input devices (HMIs), as well as external controls and host systems.

The interface transfers CAN objects following the CANopen communication profile (CiA DS301, version 4.02) developed by the umbrella organisation of CiA (CAN in Automation) in conformity with the CAL (CAN Application Layer).

Tip!

- In the »Engineer« parameter list and in the keypad, category **CAN**, you can find the parameters relevant for the CANopen interface classified in different subcategories.
- Information on CAN communication modules and CANopen interfaces of other Lenze devices is provided in the "CAN" communication manual in the Lenze library.

12.1 General information

For many years, the system bus (CAN) based on the CANopen communication profile has been integrated in Lenze inverters. Due to the lower number of data objects available, the functionality and compatibility of the previous system bus are lower as compared to CANopen. For parameter setting, two parameter data channels are always available to the user while CANopen provides only one active parameter channel.

The system bus (CANopen) of the Inverter Drives 8400 is a further development of the system bus (CAN) including the following properties:

- Full compatibility with CANopen DS301 V4.02.
- Support of the "Heartbeat" NMT slave function (DS301 V4.02)
- 1 or 2 parameterisable parameter data channels for transmitting SDOs (*Service Data Objects*)
- Up to four parameterisable process data channels (dependent on the device version) for transmitting PDOs (*Process Data Objects*)
  - All process data channels are functionally equivalent
  - Monitoring of the receive PDOs for data reception
- Adjustable error response to ...
  - physical CAN errors (frame, bit, ACK error)
  - bus-stop, bus working
  - absent PDOs
- Telegram counters for SDOs and PDOs
- Bus status diagnostics
- Boot-up telegram generation
- Emergency telegram generation
- Reset node telegram generation (for master configuration)
- Sync telegram generation and response to sync telegrams:
  - Data transmission/reception
  - Device-internal time base synchronisation
- Abort codes
- All CAN on board functions can be parameterised via codes
- Object directory (all mandatory functions, optional functions, indexes)

12.1.1 General data and application conditions

Range	Values
Communication profile	CANopen, DS301 V4.02
Communication medium	DIN ISO 11898
Network topology	Line terminated at both ends
Node addresses that can be set (max. number of nodes)	Depending on the number of SDO channels set in C00366: • 1 SDO: Node address 1 ... 127 (max. 127 nodes) • 2 SDO: Node address 1 ... 63 (max. 63 nodes) • adjustable via DIP switches or via code C00350.
Adjustable baud rates	20, 50, 125, 250, 500, 1000 kbps • adjustable via DIP switches or via code C00351.

Range	Values
Process data	For "BaseLine C" version: • Max. 2 transmit PDOs (TPDOs) with 1 ... 8 bytes (adjustable) • Max. 2 receive PDOs (RPDOs) with 1 ... 8 bytes (adjustable)
	For "StateLine C" version: • Max. 4 transmit PDOs (TPDOs) with 1 ... 8 bytes (adjustable) • Max. 4 receive PDOs (RPDOs) with 1 ... 8 bytes (adjustable) (Process data channel 4 is available from version 15.00.00!)
Parameter data	Max. 2 server SDO channels with 1 ... 8 bytes • Because of the 2 server SDO channels, the address range from 1 ... 63 is available
Transfer mode for TPDOs	• in case of data change (including adjustable blocking time) • Time-controlled, 1 to x ms • After the reception of 1 to 240 sync telegrams

12.1.2 Supported protocols

Reports	
Standard PDO protocols	PDO write PDO read
SDO protocols	SDO download SDO download initiate SDO download segment
	SDO upload SDO upload initiate SDO upload segment
	SDO abort transfer
	SDO block download SDO block download initiate SDO block download end
	SDO block upload SDO block upload initiate SDO block upload end
NMT protocols	Start remote node (master and slave)
	Stop remote node (slave)
	Enter pre-operational (slave)
	Reset node (slave and local device)
	Reset communication protocol (slave)
Monitoring protocols	Heartbeat (heartbeat producer and heartbeat consumer) • Up to 15 Heartbeat Producers can be monitored.
	Emergency telegram (to master)
More protocols	Transmitting and receiving a sync telegram • Synchronisation of the internal time base to the reception of the CAN sync telegram is possible. ▶ Synchronisation of the internal time base

12.1.3 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

Tip!

The communication times in the CAN network depend on:
- the processing time in the device
- the telegram runtime (baud rate/telegram length)
- the bus load (especially if the bus is charged with PDOs and SDOs at a low baud rate)

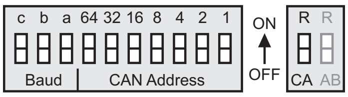
Processing time in the 8400 inverter

There are no interdependencies between parameter data and process data.

- Parameter data: approx. 5 ms (typical value)
  - For parameters concerning the motor control (e.g. C00011), the processing time may be longer (up to 30 ms).
- Process data: 1 ms

12.2 Possible settings via DIP switch

The following settings for the "CAN on board" system bus can be made via the front panel DIP switches:

DIP switch		Possible settings/detailed information
	a ... c	▸ Setting the baud rate
	1 ... 64	▸ Setting the node address
	CA	▸ Activating the bus terminating resistor

Lenze setting: All DIP switches are in the "OFF" position

Note!

- The DIP switch settings are accepted if a node address is unequal zero when the device or the 24-V supply is switched on by the DIP address.
- If all DIP switches are OFF when the device or the 24 V supply is switched on, the setting of the baud rate and node address are read out of the parameter set/parameter.

Tip!

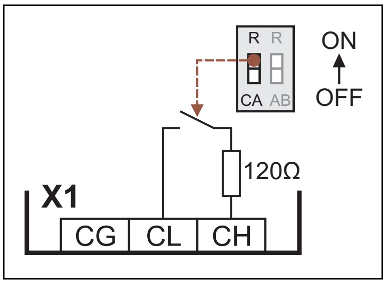
The current DIP switch settings are displayed in code C00349.

Bit 15 indicates that the setting of the DIP switches has been accepted when the device or the 24V supply has been switched on.

12.2.1 Activating the bus terminating resistor

The system bus must be terminated between CAN low and CAN high at the first and last physical node each by a resistor (120 Ω).

The 8400 inverter is provided with an integrated bus terminating resistor, which can be activated via the DIP switch labelled with "CA":



- OFF = bus terminating resistor is inactive
- ON = bus terminating resistor is active

[12-1] Activation of the integrated bus terminating resistor

12.2.2 Setting the baud rate

The baud rate can be set via code C00351 or with the DIP switches a to c:

Note!

- All DIP switches (a ... c, 1 ... 64) = OFF (Lenze setting):
  - At switching on, the settings under code C00350 (node address) and C00351 (baud rate) will become active.
- Preset baud rate: 500 kbps

DIP switch position			Baud rate
c	b	a	
ON	OFF	**ON**	20 kbps
OFF	**ON**	**ON**	50 kbps
OFF	**ON**	OFF	125 kbps
OFF	OFF	**ON**	250 kbps
OFF	OFF	OFF	**500 kbps**
ON	OFF	OFF	1000 kbit/s

12.2.3 Setting the node address

The node address can be set via code C00350 or with the DIP switches 1 to 64.

- The labelling on the housing corresponds to the values of the individual DIP switches for determining the node address.
- The valid address range depends on the number of SDO channels set in C00366:
  - 1 SDO (Lenze setting): 1 ... 127
  - 2 SDO: 1 ... 63

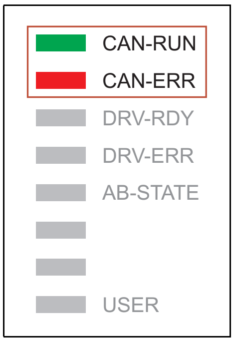
Note!

- The addresses of the nodes must differ from each other.
- All DIP switches (a ... c, 1 ... 64) = OFF (Lenze setting):
  - At switching on, the settings under code C00350 (node address) and C00351 (baud rate) will become active.

Example: Setting of the node address 23

DIP switch	64	32	16	8	4	2	1
Switch position	OFF	OFF	**ON**	OFF	**ON**	**ON**	**ON**
Value	0	0	16	0	4	2	1
Node address	**= Sum of the values = 16 + 4 + 2 + 1 = 23**						

12.3 LED status displays for the system bus



Information about the status of the system bus can be obtained quickly via LED displays "CAN-RUN" and "CAN-ERR" on the front of the inverter.

The meaning can be seen from the tables below.

Inverter is not (yet) active on the system bus

LED display	Meaning
(CAN-ERR is permanently lit)	Inverter is not active on the system bus / Bus Off
(CAN-RUN and CAN-ERR flicker)	Automatic detection of baud rate is active

Inverter is active on the system bus

- LED "CAN-RUN" signals the CANopen state:

LED display	CANopen state
(CAN-RUN is blinking every 0.2 seconds)	Pre-Operational
(CAN-RUN is permanently lit)	Operational
(CAN-RUN is blinking every second)	Stopped

- LED "CAN-ERR" signals a CANopen error:

LED display	CANopen error
(CAN-ERR is blinking once, then off for 1 second)	Warning Limit reached
(CAN-ERR is blinking twice, then off for 1 second)	Node Guard Event
(CAN-ERR is blinking three times, then off for 1 second)	Sync Message Error (only possible in the "Operational" state)

12.4 Going online via the system bus

The integrated system bus interface (CAN on board, X1 terminal) can also be used for the communication between the »Engineer« and the inverter, alternatively to the USB diagnostic adapter.

- Lenze offers the following communication accessories for connection to the PC:

Communication accessories	PC interface
PC system bus adapter 2173 incl. connection cable and voltage supply adapter • for DIN keyboard connection (EMF2173IB) • for PS/2 keyboard connection (EMF2173IBV002) • for PS/2 keyboard connection with electrical isolation (EMF2173IBV003)	Parallel interface (LPT port)
PC system bus adapter 2177 incl. connection cable (EMF2177IB)	USB (Universal Serial Bus)

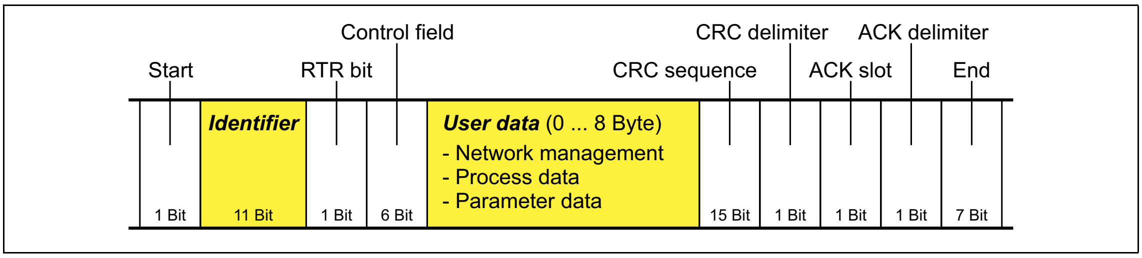
Note!

- For detailed information about the PC system bus adapter, please see the "CAN Communication Manual".
- Please observe the documentation for the PC system bus adapter!
- In the »Engineer«, go to the *Device assignment offline devices* dialog box and select the "System bus CAN" entry from the **Bus connection** list field to establish an online connection.

12.5 Reinitialising the CANopen interface

The C00002/26 = "1: On / start" device command reinitialises the CANopen interface of the inverter ("Reset node"), which is required after e.g. changing the data transfer rate, the node address or the identifiers, respectively.

12.6 Structure of the CAN data telegram



[12-2] Basic structure of the CAN telegram

The following subchapters provide a detailed description of the identifier and the user data. The other signals refer to the transfer characteristics of the CAN telegram whose description is not included in the scope of this documentation.

Tip!

Please visit the homepage of the CAN user organisation CiA (CAN in automation) for further information:
http://www.can-cia.org

12.6.1 Identifier

The principle of the CAN communication is based on a message-oriented data exchange between a transmitter and many receivers. All nodes can transmit and receive quasi-simultaneously.

The identifier, also called COB-ID (abbr. for communication object identifier), is used to control which node is to receive a transmitted message. In addition to the addressing, the identifier contains information on the priority of the message and the type of user data.

The identifier consists of a basic identifier and the node address of the node to be addressed:

Identifier (COB-ID) = basic identifier + node address (node ID)

Exception: The identifier for process data/heartbeat/emergency objects as well as network management and sync telegrams is freely assigned by the user (either manually or automatically by the network configurator), or is permanently assigned.

Node address (node ID)

Every node of the system bus network must be assigned to a node address (also called node ID) within the valid address range (1 ... 127) for unambiguous identification.

- Assigning a node address more than once within a network is impermissible.
- The own node address can be configured via the DIP switches or via code C00350. ▶ Setting the node address (📖 809)

Identifier assignment

The system bus is message-oriented instead of node-oriented. Every message has an unambiguous identification, the identifier. For CANopen, node-oriented transfer is achieved by the fact that every message has only one transmitter.

- The basic identifiers for network management (NMT) and the basic SDO channel (SDO1) are defined in the CANopen protocol and cannot be changed.
- The basic identifiers of the PDOs are preset in the Lenze setting according to the "Predefined Connection Set" of DS301 V4.02.

Object		Basis identifier (CANBaseID)		Direction
Network management (NMT)		0	0x0000	
Sync [1)]		128	0x0080	
Emergency [1)]		128	0x0080	Device ➞ system bus
PDO1 (Process data channel 1)	TPDO1	384	0x180	Device ➞ system bus
	RPDO1	512	0x200	System bus ➞ device
PDO2 (Process data channel 2)	TPDO2	640	0x280	Device ➞ system bus
	RPDO2	768	0x300	System bus ➞ device
PDO3 (Process data channel 3)	TPDO3	896	0x380	Device ➞ system bus
	RPDO3	1024	0x400	System bus ➞ device
PDO4 [2)] (Process data channel 4)	TPDO4	1152	0x480	Device ➞ system bus
	RPDO4	1280	0x500	System bus ➞ device
SDO1 (Parameter data channel 1)	TSDO1	1408	0x0580	Device ➞ system bus
	RSDO1	1536	0x0600	System bus ➞ device
SDO2 (Parameter data channel 2)	TSDO2	1472	0x05C0	Device ➞ system bus
	RSDO2	1600	0x0640	System bus ➞ device
Heartbeat		1792	0x0700	Device ➞ system bus
Boot-up		1792	0x0700	Device ➞ system bus

[1)] If you set the sync identifiers manually, observe the use of the emergency telegram, since it has the same COB-ID.
[2)] From version 15.00.00

Tip!

If required, the identifiers of the PDOs and the identifiers for the sync telegram can be changed via parameters/indices.

▶ Identifiers of the process data objects (📖 833)

▶ Synchronisation of PDOs via sync telegram (📖 837)

The active identifiers of the PDOs are displayed in C00355/x.

12.6.2 User data

All nodes communicate by exchanging data telegrams via the system bus. The user data area of the CAN telegram either contains network management data or parameter data or process data:

Network management data

(NMT data)

- Control information on start, stop, reset, etc. of communication to specific nodes or to all nodes of the CAN network.

Process data

(PDOs – process data objects)

- Process data are transferred via the process data channel.
- Process data can be used to control the inverter.
- Process data are not saved in the inverter.
- Process data are transmitted between host system and nodes to ensure continuous exchange of current input and output data.
- Process data usually are unscaled/scalable raw data.
- Process data are, for instance, setpoints and actual values.
- The exact meaning of the PDO file contents is determined via the function block editor (FB Editor) in the I/O level or via the PDO mapping.

Parameter data

(SDOs – service data objects)

- Parameter data are the CANopen indexes or, in case of Lenze devices, the codes.
- Parameters are set, for instance, when the system is initially adjusted during commissioning or when the material of the production machine is changed.
- Parameter data are transmitted as SDOs via the parameter data channel. They are acknowledged by the receiver, i.e. the transmitter gets a feedback about the transmission being successful or not.
- The parameter data channel enables access to all Lenze codes and CANopen indexes.
- Parameter changes are automatically saved to the inverter until mains switching.
- In general, the parameter transfer is not time-critical.
- Parameter data are, for instance, operating parameters, diagnostic information and motor data as well as control information on the interconnection of function blocks in the I/O level of the FB Editor.

12.7 Communication phases/network management

Regarding communication via the system bus, the inverter distinguishes between the following statuses:

State	Explanation
"Initialisation" (Initialisation)	After switch-on, an initialisation run is carried out. • During this phase, the inverter is not involved in the data exchange via the bus. • The standard values are re-written to all CAN-relevant parameters. • After initialisation is completed, the inverter is automatically set to the "Pre-Operational" status.
"Pre-Operational" (before being ready for operation)	Parameter data can be received, process data are ignored.
"Operational" (ready for operation)	Parameter data and process data can be received!
"Stopped" (stopped)	Only network management telegrams can be received.

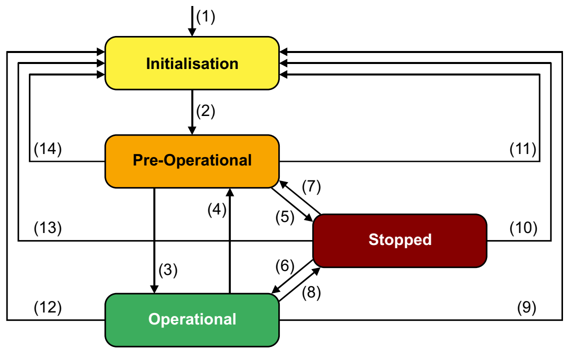
Communication object	Initialisation	Pre-Operational	Operational	Stopped
PDO			●	
SDO		●	●	
Sync		●	●	
Emergency		●	●	
Boot-up	●			
Network management (NMT)		●	●	●

Tip!

Part of the initialisation or the entire initialisation can be carried out anew in every status by transferring the corresponding network management telegrams.

The current CAN status is displayed in C00359 for diagnostic purposes.

12.7.1 State transitions

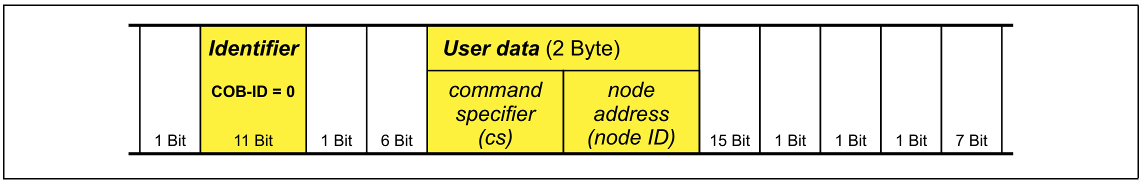


[12-3] NMT status transitions in the CAN network

Transition	NMT command	Status after change	Effects on process/parameter data after status change
(1)	-	Initialisation	Initialisation starts automatically when the mains is switched on. • During initialisation, the inverter is not involved in the data exchange. • After the initialisation is completed, the node sends a boot-up message with an individual identifier and automatically changes to the "pre-operational" status.
(2)	-	Pre-Operational	In this phase, the master determines the way in which the node(s) takes/take part in communication.
ℹ	From here, the master changes the statuses for the entire network. • A target address included in the NMT command defines the receiver(s). • If the 8400 inverter is configured as CAN master, the status is automatically changed to "Operational" after a waiting time has expired (C00356/1), and the 0x0100 ("Start remote node") NMT command is transmitted to all nodes. • Data can only be exchanged via process data objects if the status is "Operational"!		
(3), (6)	0x01 xx Start remote node	Operational	Network management/sync/emergency telegrams as well as process data (PDO) and parameter data (SDO) are active. Optional: When the status is changed, event and time-controlled process data (PDOs) are transmitted once.
(4), (7)	0x80 xx Enter Pre-Operational	Pre-Operational	Network management/sync/emergency telegrams and parameter data (SDO) are active.
(5), (8)	0x02 xx Stop remote node	Stopped	Only network management telegrams can be received.
(9), (10), (11)	0x81 xx Reset node	Initialisation	All CAN-relevant parameters (CiA DS 301) are initialised with the saved values.
(12), (13), (14)	0x82 xx Reset communication		All CAN-relevant parameters (CiA DS 301) are initialised with the saved values.
ℹ	Meaning of the node address in the NMT command: • xx = 0x00: If this assignment is selected, the telegram addresses all nodes (broadcast telegram). The status of all nodes can be changed at the same time. • xx = Node ID: If a node address is specified, only the status of the node with the corresponding address changes.		

12.7.2 Network management telegram (NMT)

The telegram for the network management contains identifier "0" and the command included in the user data which consists of the command byte and the node address:



[12-4] Network management telegram for changing over the communication phases

Command specifier (cs)		NMT command
dec	hex	
1	0x01	Start remote node
2	0x02	Stop remote node
128	0x80	Enter Pre-Operational
129	0x81	Reset node
130	0x82	Reset communication

The communication phases are changed over by a node, the CAN master, for the entire network. The CAN master can also be a inverter. ▶ Parameterising the inverter as CAN master (📖 818)

Meaning of the node address in the user data:

- node ID = "0": The telegram addresses all nodes (broadcast telegram). The status of all nodes can be changed at the same time.
- node ID = "1" ... "127": If a node address is specified, only the status of the node with the corresponding address changes.

Example:

Data can only be exchanged via process data objects if the status is "Operational".If the CAN master is supposed to switch all nodes connected to the bus from the "Pre-Operational" communication status to the "Operational" communication status, the identifier and user data in the transmission telegram must be set as follows:

- Identifier: 0x00 (network management)
- User data: 0x0100 ("Start remote node" NMT command to all nodes)

12.7.3 Parameterising the inverter as CAN master

If the initialisation of the system bus and the associated status change from "Pre-Operational" to "Operational" is not effected by a superimposed host system, the inverter can instead be defined to be a "quasi" master to execute this task.

The inverter is configured as CAN master in C00352.

- Being the CAN master, the inverter sets all nodes connected to the bus (broadcast telegram) to the "Operational" communication status with the "Start remote node" NMT telegram. Only in this communication status, data can be exchanged via process data objects.
- A delay time can be set in C00356/1 which must expire after mains switching before the inverter transmits the "Start remote node" NMT telegram.

Parameters	Info	Lenze setting	
		Value	Unit
C00352	CAN slave/master	slave	
C00356/1	CAN delay boot-up - Operational	3000	ms

Note!

The changes of the master/slave operation in C00352 will not be activated until

- another mains switching of the inverter

or

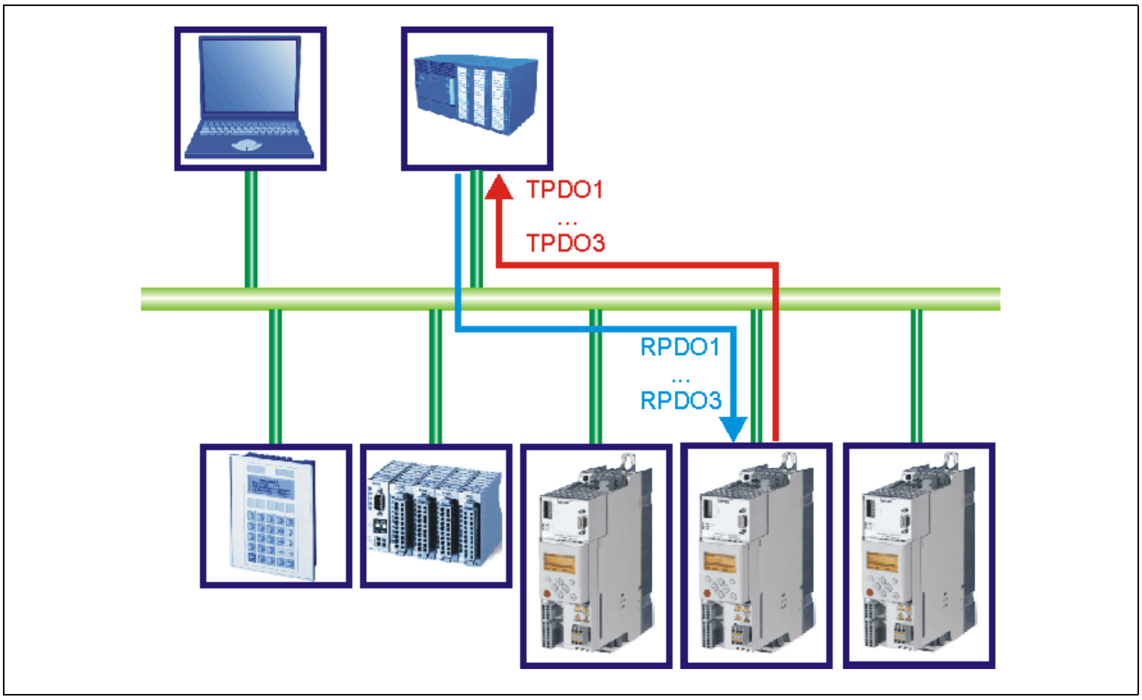
- the "Reset node" or "Reset communication" NMT telegram has been transmitted to the inverter.

The "CAN reset node" device command (C00002/26) is provided as an alternative to the "Reset node" NMT telegram for the reinitialisation of the CAN-specific device parameters.

Tip!

Master functionality is only required during the initialisation phase of the drive system.

12.8 Process data transfer



[12-5] PDO data transfer from / to the higher-level host system

For transmitting process data, up to four separated process data channels (PDO1 ... PDO4) are available depending on the device version and software version.

Definitions

- Process data telegrams between the host system and the devices are distinguished in terms of direction as follows:
  - Process data telegrams to the device (RPDO)
  - Process data telegrams from the device (TPDO)
- The CANopen process data objects are designated as seen from the node's view:
  - Receive PDOs (RPDOx): Process data object received by a node
  - Transmit PDOs (TPDOx): Process data object sent by a node

Note!

Data can only be exchanged via process data objects if the status is "Operational"!

▶ Communication phases/network management (📖 815)

12.8.1 Available process data objects

The number of available process data objects for inverters of the 8400 series depends on the device version:

Process data object	Version	
	BaseLine C	StateLine C HighLine C TopLine C
RPDO1 | Port block "LP_CanIn1"	●	●
RPDO2 | "LP_CanIn2" port block	●	●
RPDO3 | "LP_CanIn3" port block		●
RPDO4 | "LP_CanIn4" port block		● (from version 15.00.00)
TPDO1 | "LP_CanOut1" port block	●	●
TPDO2 | "LP_CanOut2" port block	●	●
TPDO3 | "LP_CanOut3" port block		●
TPDO4 | "LP_CanOut4" port block		● (from version 15.00.00)

Receive PDOs (RPDOs)

The process data objects transmitted from the system bus to the drive are processed via the LP_CanIn1 ... LP_CanIn4 port blocks.

- Every port block provides 4 words (2 bytes/word). The data of every first word are provided in a bit decoded manner (bit 0 ... 15).
- The first word of the LP_CanIn1 port block is defined as control word *wCtrl*. The *wCtrl* control word does not have a permanent connection to the device control and can be used as required. The predefined assignment of the *wCtrl* control word in the C00007 = "30: CAN" control mode depends on the technology application selected in C00005:
  - **TA "Actuating drive speed":**
    Process data assignment for fieldbus communication (📖 474)
  - **TA "Table positioning":**
    Process data assignment for fieldbus communication (📖 538)
  - **TA "Abschaltpositionierung":**
    Process data assignment for fieldbus communication (📖 564)

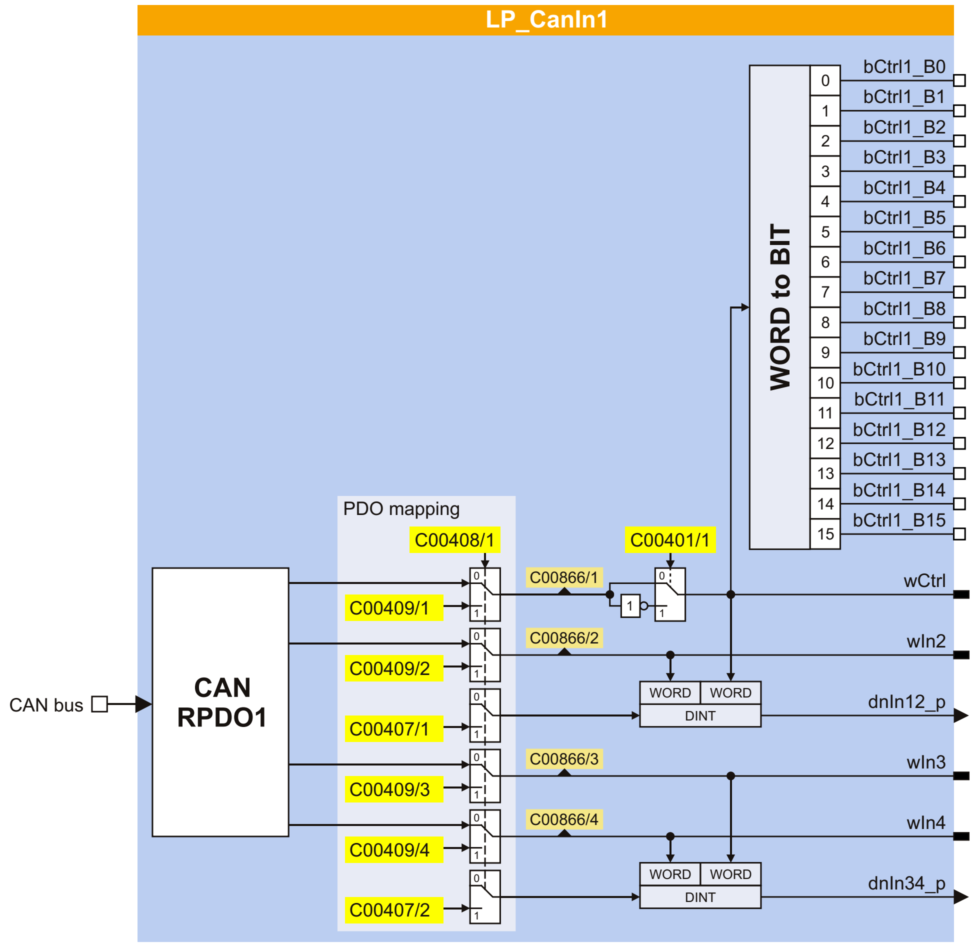
Transmit PDOs (TPDOs)

The process data transmitted from the drive to the system bus are processed via the LP_CanOut1 ... LP_CanOut4 port blocks.

- Every port block receives 4 words (2 bytes/word). The data of every first word are transmitted bit by bit (bit 0 ... 15).
- The first word of the LP_CanOut1 port block is defined as the *wState* status word. The *wState* status word does not have a permanent connection to the device control and can be used as required.
  - For a predefined assignment, see the wDeviceStatusWord status word of the drive interface.

12.8.1.1 RPDO1 | Port block "LP_CanIn1"

The **LP_CanIn1** port block maps process data object RPDO1 in the FB Editor.



dnIn12_p output is available from version 02.00.00!

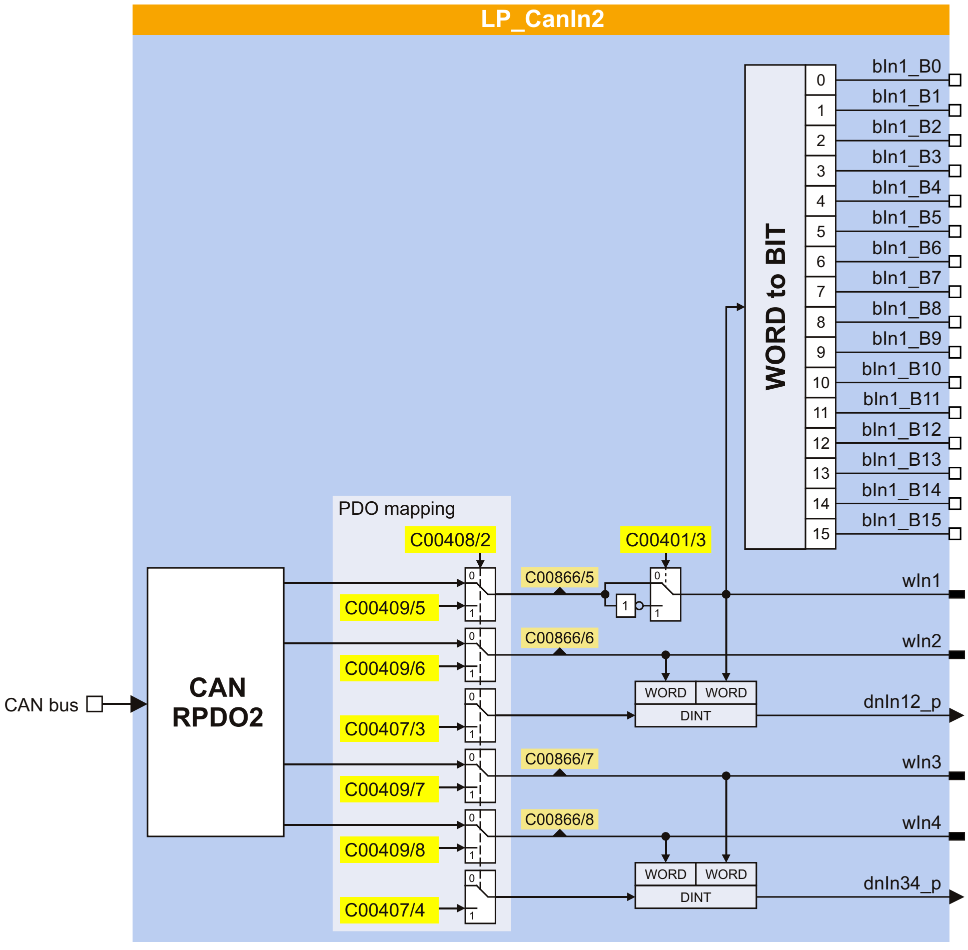
Short overview of the parameters for LP_CanIn1:

Parameters	Info	Lenze setting
C00401/1	LP_CanIn1: Inversion bCtrl1_B0..15	0x0000
C00866/1	LP_CanIn1: wCtrl	-
C00866/2	LP_CanIn1: wIn2	-
C00866/3	LP_CanIn1: wIn3	-
C00866/4	LP_CanIn1: wIn4	-
PDO mapping		
C00408/1	LP_CanIn1: Mapping selection	CanIn
C00409/1	LP_CanIn1: wCtrl MapVal	0
C00409/2	LP_CanIn1: wIn2 MapVal	0
C00409/3	LP_CanIn1: wIn3 MapVal	0
C00409/4	LP_CanIn1: wIn4 MapVal	0
Greyed out = display parameter		

Parameters	Info	Lenze setting
C00407/1	LP_CanIn1: dwIn12 MapVal • From version 12.00.00 • The mapping for the double word is ORed with the mapping setting in C00409/1 and C00409/2.	0
C00407/2	LP_CanIn1: dwIn34 MapVal • From version 12.00.00 • The mapping for the double word is ORed with the mapping setting in C00409/3 and C00409/4.	0
Greyed out = display parameter		

12.8.1.2 RPDO2 | "LP_CanIn2" port block

The **LP_CanIn2** port block maps process data object RPDO2 in the FB Editor.



dnIn12_p output is available from version 02.00.00!

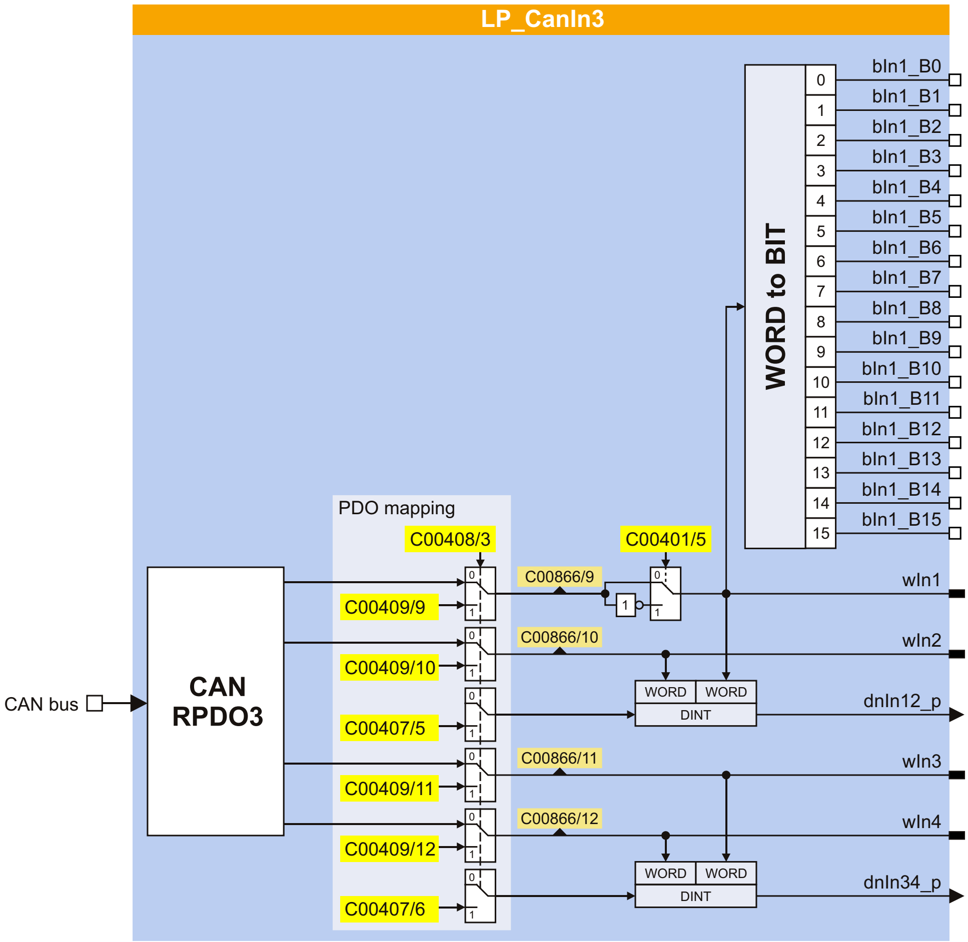
Short overview of the parameters for LP_CanIn2:

Parameters	Info	Lenze setting
C00401/3	LP_CanIn2: Inversion bIn1_B0..15	0x0000
C00866/5	LP_CanIn2: wIn1	-
C00866/6	LP_CanIn2: wIn2	-
C00866/7	LP_CanIn2: wIn3	-
C00866/8	LP_CanIn2: wIn4	-
PDO mapping		
C00408/2	LP_CanIn2: Mapping selection	CanIn
C00409/5	LP_CanIn2: wIn1 MapVal	0
C00409/6	LP_CanIn2: wIn2 MapVal	0
C00409/7	LP_CanIn2: wIn3 MapVal	0
C00409/8	LP_CanIn2: wIn4 MapVal	0
Greyed out = display parameter		

Parameters	Info	Lenze setting
C00407/3	LP_CanIn2: dwIn12 MapVal • From version 12.00.00 • The mapping for the double word is ORed with the mapping setting in C00409/5 and C00409/6.	0
C00407/4	LP_CanIn2: dwIn34 MapVal • From version 12.00.00 • The mapping for the double word is ORed with the mapping setting in C00409/7 and C00409/8.	0
Greyed out = display parameter		

12.8.1.3 RPDO3 | "LP_CanIn3" port block

The **LP_CanIn3** port block maps process data object RPDO3 in the FB Editor.



dnIn12_p output is available from version 02.00.00!

Short overview of the parameters for LP_CanIn3:

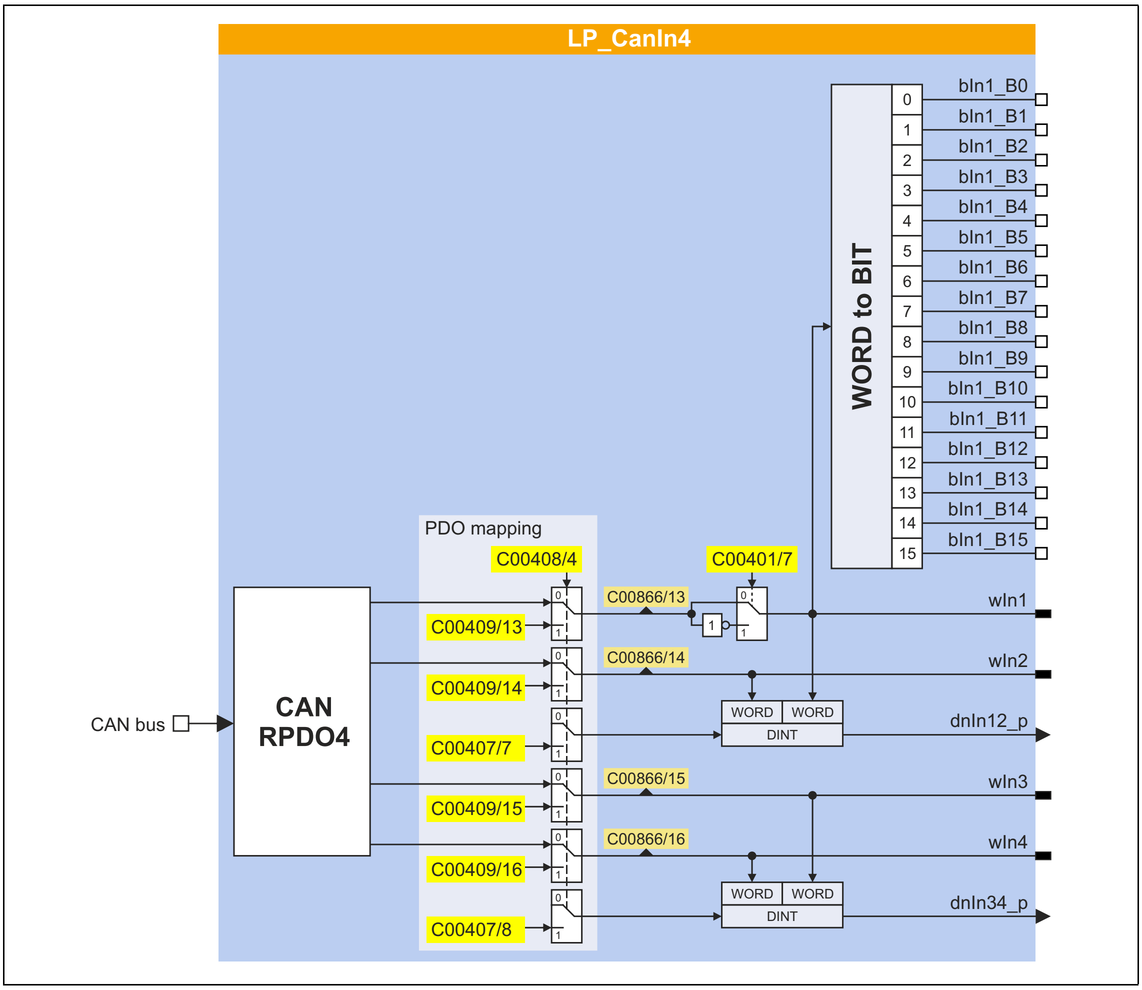
Parameters	Info	Lenze setting
C00401/5	LP_CanIn3: Inversion bIn1_B0..15	0x0000
C00866/9	LP_CanIn3: wIn1	-
C00866/10	LP_CanIn3: wIn2	-
C00866/11	LP_CanIn3: wIn3	-
C00866/12	LP_CanIn3: wIn4	-
PDO mapping		
C00408/3	LP_CanIn3: Mapping selection	CanIn
C00409/9	LP_CanIn3: wIn1 MapVal	0
C00409/10	LP_CanIn3: wIn2 MapVal	0
C00409/11	LP_CanIn3: wIn3 MapVal	0
C00409/12	LP_CanIn3: wIn4 MapVal	0
Greyed out = display parameter		

Parameters	Info	Lenze setting
C00407/5	LP_CanIn3: dwIn12 MapVal • From version 12.00.00 • The mapping for the double word is ORed with the mapping setting in C00409/9 and C00409/10.	0
C00407/6	LP_CanIn3: dwIn34 MapVal • From version 12.00.00 • The mapping for the double word is ORed with the mapping setting in C00409/11 and C00409/12.	0
Greyed out = display parameter		

12.8.1.4 RPDO4 | "LP_CanIn4" port block

This function extension is available from version 15.00.00!

The **LP_CanIn4** port block maps process data object RPDO4 in the FB Editor.



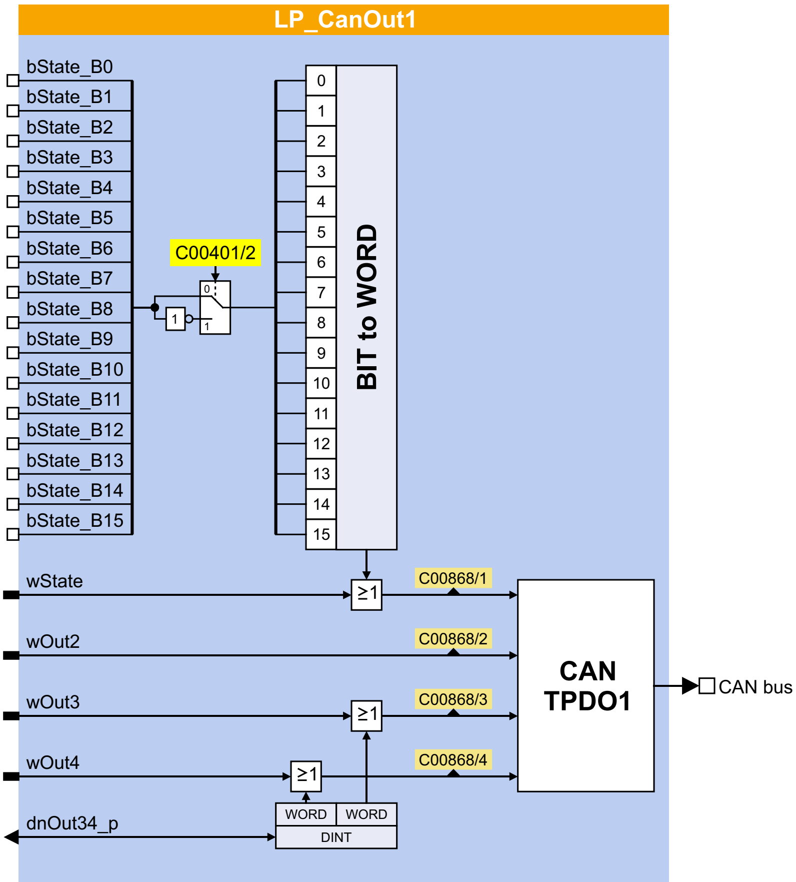
Short overview of the parameters for LP_CanIn4:

Parameters	Info	Lenze setting
C00401/7	LP_CanIn4: Inversion bIn1_B0..15	0x0000
C00866/13	LP_CanIn4: wIn1	-
C00866/14	LP_CanIn4: wIn2	-
C00866/15	LP_CanIn4: wIn3	-
C00866/16	LP_CanIn4: wIn4	-
PDO mapping		
C00408/4	LP_CanIn4: Mapping Selection	CanIn
C00409/13	LP_CanIn4: wIn1 MapVal	0
C00409/14	LP_CanIn4: wIn2 MapVal	0
C00409/15	LP_CanIn4: wIn3 MapVal	0
C00409/16	LP_CanIn4: wIn4 MapVal	0
Greyed out = display parameter		

Parameters	Info	Lenze setting
C00407/7	LP_CanIn4: dwIn12 MapVal • The mapping for the double word is ORed with the mapping setting in C00409/13 and C00409/14.	0
C00407/8	LP_CanIn4: dwIn34 MapVal • The mapping for the double word is ORed with the mapping setting in C00409/15 and C00409/16.	0
Greyed out = display parameter		

12.8.1.5 TPDO1 | "LP_CanOut1" port block

The **LP_CanOut1** port block maps process data object TPDO1 in the FB Editor.

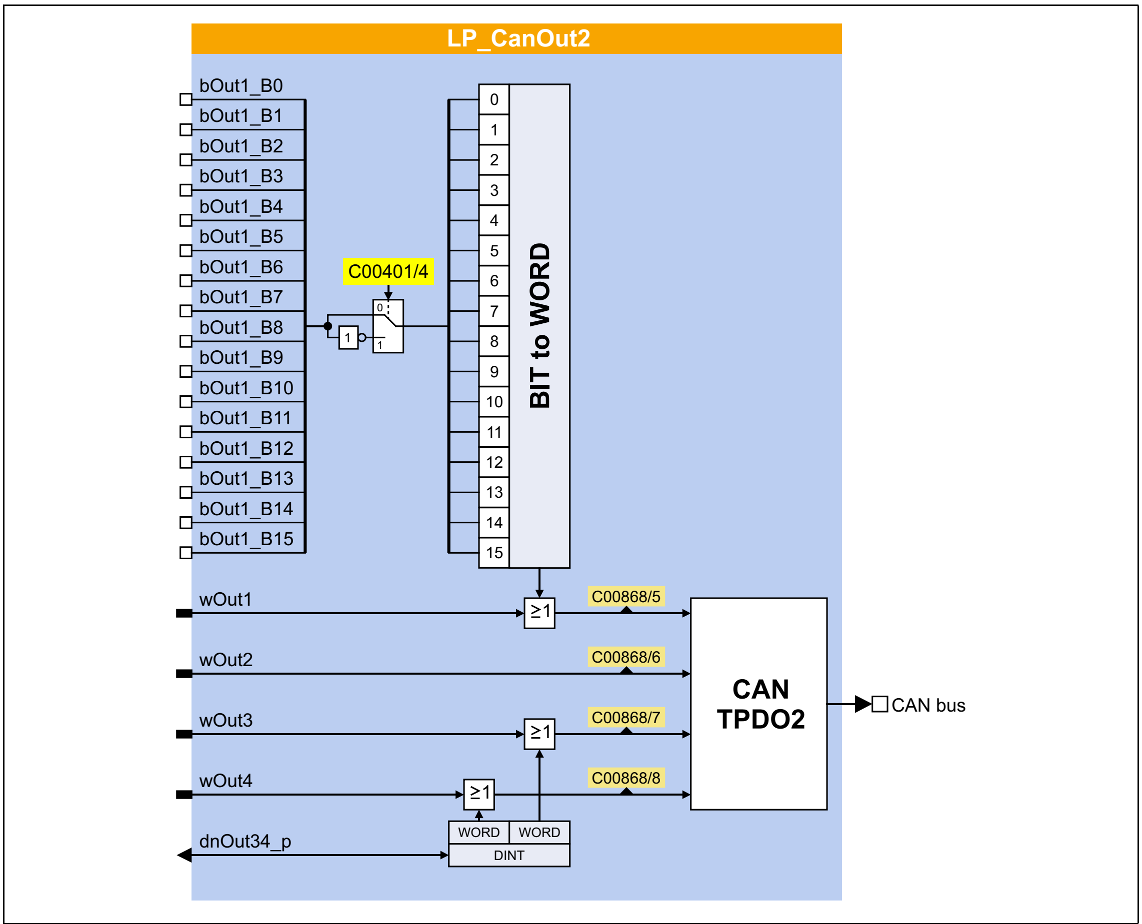


Short overview of the parameters for LP_CanOut1:

Parameters	Info	Lenze setting
C00401/2	LP_CanOut1: Inversion bState_B0..15	0x0000
C00868/1	LP_CanOut1:wState	-
C00868/2	LP_CanOut1:wOut2	-
C00868/3	LP_CanOut1:wOut3	-
C00868/4	LP_CanOut1: wOut4	-
Greyed out = display parameter		

12.8.1.6 TPDO2 | "LP_CanOut2" port block

The **LP_CanOut2** port block maps process data object TPDO2 in the FB Editor.

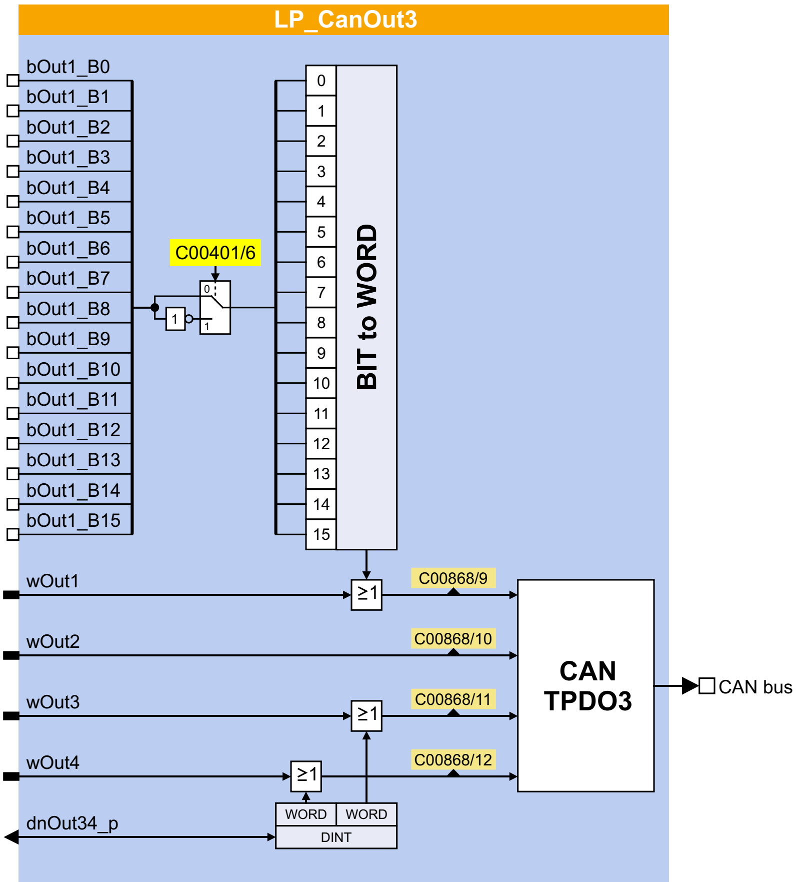


Short overview of the parameters for LP_CanOut2:

Parameters	Info	Lenze setting
C00401/4	LP_CanOut2: Inversion bOut1_B0..15	0x0000
C00868/5	LP_CanOut2: wOut1	-
C00868/6	LP_CanOut2: wOut2	-
C00868/7	LP_CanOut2: wOut3	-
C00868/8	LP_CanOut2: wOut4	-
Greyed out = display parameter		

12.8.1.7 TPDO3 | "LP_CanOut3" port block

The **LP_CanOut3** port block maps process data object TPDO3 in the FB Editor.



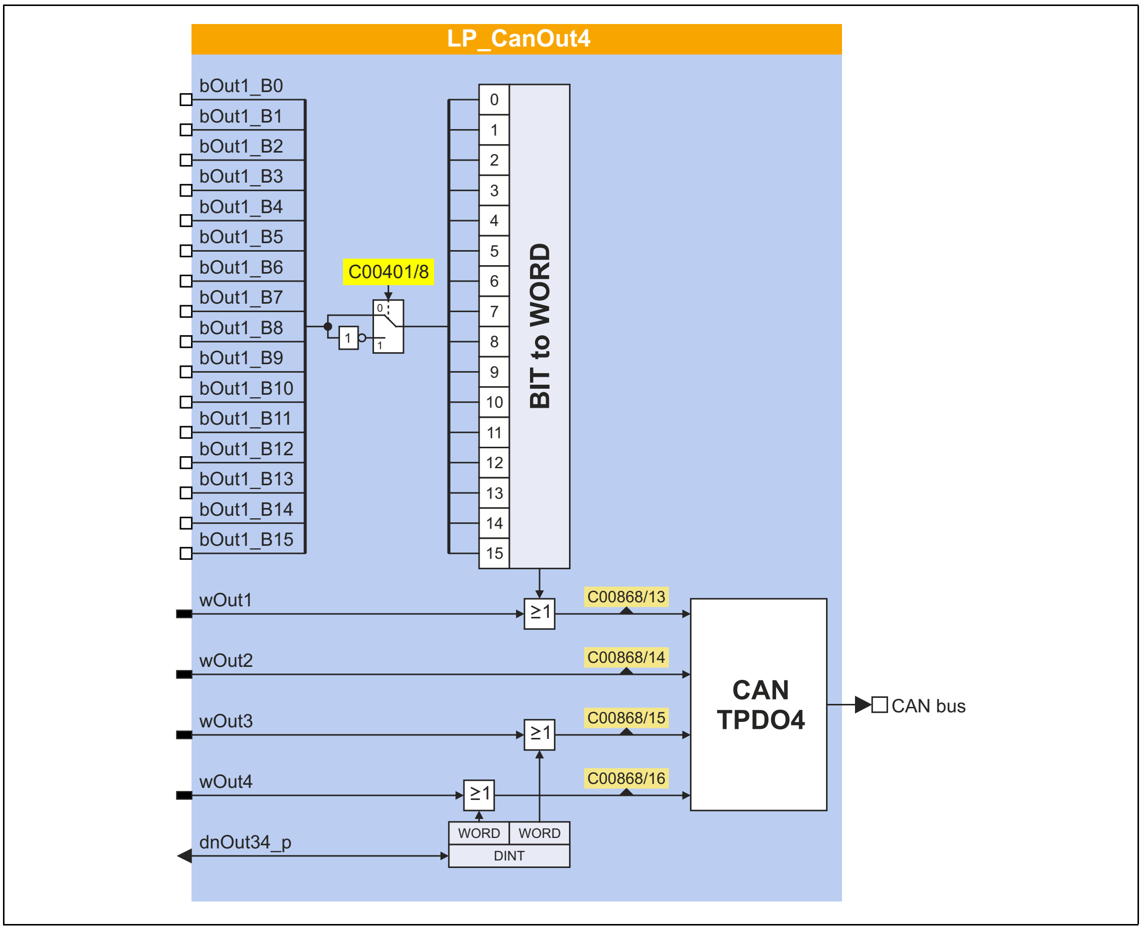
Short overview of the parameters for LP_CanOut3:

Parameters	Info	Lenze setting
C00401/6	LP_CanOut3: Inversion bOut1_B0..15	0x0000
C00868/9	LP_CanOut3: wOut1	-
C00868/10	LP_CanOut3: wOut2	-
C00868/11	LP_CanOut3: wOut3	-
C00868/12	LP_CanOut3: wOut4	-
Greyed out = display parameter		

12.8.1.8 TPDO4 | "LP_CanOut4" port block

This function extension is available from version 15.00.00!

The **LP_CanOut4** port block maps process data object TPDO4 in the FB Editor.



Short overview of the parameters for LP_CanOut4:

Parameters	Info	Lenze setting
C00401/8	LP_CanOut4: Inversion bOut1_B0..15	0x0000
C00868/13	LP_CanOut4: wOut1	-
C00868/14	LP_CanOut4: wOut2	-
C00868/15	LP_CanOut4: wOut3	-
C00868/16	LP_CanOut4: wOut4	-
Greyed out = display parameter		

12.8.2 Identifiers of the process data objects

For the process data objects PDO1 ... PDO4, three identifier assignment procedures are available in C00353/1...4 :

- Identifier (COB-ID) = node address (C00350) + CANBaseID (Lenze setting)
- Identifier (COB-ID) = node address (C00350) + LenzeBaseID
- Identifier (COB-ID) = C0354/x

Lenze setting: Basic identifiers according to the "Predefined Connection Set" of DS301 V4.02

In the Lenze setting, the identifiers for the process data objects PDO1 ... PDO4 consist of the node address set in C00350 and a basic identifier (CANBaseID) which corresponds to the "Predefined Connection Set" of DS301 V4.02:

Object			Basis identifier (CANBaseID)	
PDO1	TPDO1	COB-ID = node address (C00350) +	384	0x180
	RPDO1		512	0x200
PDO2	TPDO2		640	0x280
	RPDO2		768	0x300
PDO3	TPDO3		896	0x380
	RPDO3		1024	0x400
PDO4	TPDO4		1152	0x480
	RPDO4		1280	0x500

Tip!

The active identifiers (COB-ID) are displayed in C00355/x.

OPTIONAL: Use the basic identifier according to Lenze definition (LenzeBaseID)

If the basic identifiers shall be used instead according to the Lenze definition, go to C00353/x and select "0: COBID = C0350 + LenzeBaseID" for the coresponding PDO.

The basic Lenze identifiers which differ from the "Predefined Connection Set" are highlighted in colour in the following table:

Object			Basic identifier (LenzeBaseID)	
PDO1	TPDO1	COB-ID = node address (C00350) +	384	0x180
	RPDO1		512	0x200
PDO2	TPDO2		641	0x281
	RPDO2		640	0x280
PDO3	TPDO3		769	0x301
	RPDO3		768	0x300
PDO4	TPDO4		897	0x381
	RPDO4		896	0x380

OPTIONAL: Set identifier individually

When "2: COBID = C0354/x" in C00353/x, the identifiers for the PDOs can be individually set via the Lenze codes and CANopen indexes listed in the table below. That way, identifiers independent of the node address can be set for specific PDOs.

- If identifiers are assigned individually, all PDOs must have basic identifier values in the range of 385 ... 1407.

Object			Lenze code	CANopen index	Default setting
PDO1	TPDO1	COB-ID =	C00354/1	I-1400/1	0x201
	RPDO1		C00354/2	I-1800/1	0x181
PDO2	TPDO2		C00354/3	I-1401/1	0x301
	RPDO2		C00354/4	I-1801/1	0x281
PDO3	TPDO3		C00354/5	I-1402/1	0x401
	RPDO3		C00354/6	I-1802/1	0x381
PDO4	TPDO4		C00354/7	I-1403/1	0x501
	RPDO4		C00354/8	I-1803/1	0x481

Note!

After a node address change (C00350) and a CAN reset node afterwards, the subcodes of C00354 automatically resume the values which result from the respective basic identifier and the set node address.

12.8.3 Transmission type

Process data objects can be transmitted in an event-controlled or time-controlled manner. The below table shows that it is possible to combine the different methods by means of logic operations (AND, OR):

- Event-controlled
  The PDO is sent when a special device-internal event has occurred, e.g. when the data contents of the TPDO have changed or when a transmission cycle time has elapsed
- Synchronous transmission
  A TPDO (or RPDO) is transmitted (or received) after the device has received a sync telegram (COB-ID 0x80).
- Cyclic transmission
  The cyclic transmission of PDOs takes place when the transmission cycle time has elapsed.
- Polled via RTR
  A TPDO is transmitted when another device requests it by means of a data request telegram (RTR remote transmit request). For this purpose, the data requester (e.g. the master) sends the data request telegram with the COB-ID of the TPDO requested to be sent. The receiver recognises the RTR and transmits the corresponding PDO.

Transmission type	PDO transmission			Logic combination of different transmission types
	cyclic	synchronous	event-controlled	
0		●	●	AND
1 ... 240		●		-
254, 255	●		●	OR

Transmission type	Description
0	Synchronous and acyclic: The PDO is transmitted on an event-controlled basis with every sync (e.g. when a bit change occurs in the PDO).
1 ... 240	Synchronous and cyclic (sync-controlled with response): • Selection n = 1: The PDO is transmitted with every sync. • Selection 1 < n ≤ 240: The PDO is transmitted with every n-th sync. From version 16.00.00 onwards, the transmit PDOs are also sent when the sync telegram is generated. Precondition: the sync Tx identifier and sync Rx identifier have the same value.
241 ... 251	Reserved
252	Synchronous - RTR only
253	Asynchronous - RTR only
254, 255	Asynchronous - manufacturer-specific / device profile-specific: If this value is entered, the PDO transmission is event-controlled or cyclic. (Note: The values "254" and "255" have the same meaning). For a cyclic transmission, a cycle time must be entered for the respective PDO. In this case, cyclic transmission takes place in addition to event-controlled transmission.

The communication parameters such as the transmission mode and cycle time can be set freely for every PDO and independently of the settings of other PDOs:

Parameters	Info	Lenze setting	
		Value	Unit
CAN1_OUT			
C00322/1	Transmission mode	254	
C00324/2	Blocking time	0	ms
C00356/5	Cycle time	0	ms
C00358/1	Data length	8	Byte
CAN2_OUT			
C00322/2	Transmission mode	254	
C00324/3	Blocking time	0	ms
C00356/2	Cycle time	0	ms
C00358/2	Data length	8	Byte
CAN3_OUT			
C00322/3	Transmission mode	254	
C00324/4	Blocking time	0	ms
C00356/3	Cycle time	0	ms
C00358/3	Data length	8	Byte
CAN4_OUT (from version 15.00.00			
C00322/4	Transmission mode	254	
C00324/5	Blocking time	0	ms
C00356/6	Cycle time	0	ms
C00358/4	Data length	8	Byte
CAN1_IN ... CAN4_IN			
C00323/1...4	Transmission mode CAN1_IN ... CAN4_IN • In the case of the RPDO serves as monitoring setting in the case of sync-controlled PDOs.	254	

Blocking time

In C00324/x a "blocking time" can be set which defines the shortest transmission cycle with the transmission type"asynchronous - manufacturer-specific/device profile-specific".

Example: Cycle time = 500 ms, blocking time = 100 ms, sporadic data change:

- With a sporadic data change < 500 ms, quickest transmission takes place every 100 ms due to the set blocking time (event-controlled transmission). The transmission cycle timer is reset to 0 if the transmission has been activated in an event-controlled way.
- In the case of a sporadic data change > 500 ms, due to the cycle time set, transmission takes place every 500 ms (cyclic transmission).

Tip!

The communication parameters can also be set via the following CANopen objects:

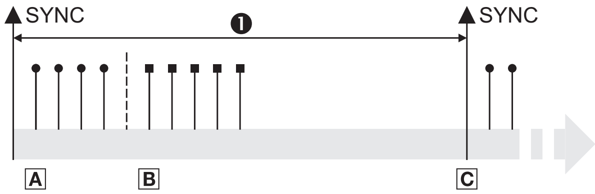
- I-1400 ... I-1403: Communication parameters for RPDO1 ... RPDO4
- I-1800 ... I-1803: Communication parameters for TPDO1 ... TPDO4

12.8.4 Synchronisation of PDOs via sync telegram

During cyclic transmission, one or more PDOs are transmitted/received in fixed time intervals. An additional specific telegram, the so-called sync telegram, is used for synchronising cyclic process data.

- The sync telegram is the trigger point for the transmission of process data from the slaves to the master and for the acceptance of process data from the master in the slaves.
- For sync-controlled process data processing, the sync telegram must be generated accordingly.
- The response to a sync telegram is determined by the transmission type selected.
  ▶ Transmission type (835)

Basic workflow



❶ Sync cycle time (C01121)

[12-6] Sync telegram

A. After the sync telegram has been received, the slaves transmit the synchronous process data to the master (TPDOs). The master reads them as process input data.
B. When the transmission process is completed, the slaves receive (RPDOs) the process output data (of the master).
   - All other telegrams (e.g. parameters or event-controlled process data) are accepted acyclically by the slaves after the transmission is completed.
   - Illustration [12-6] does not include acyclic data. However, they need to be considered when dimensioning the cycle time.
C. The data are accepted in the slave with the next sync telegram if the Rx mode is set to 1 ... 240. If the Rx mode is 254 or 255, the data are accepted in the next device cycle, irrespective of the sync telegram.

Short overview: Parameters for the synchronisation via sync telegram

Parameters	Info	Lenze setting		Assignment	
		Value	Unit	Sync master	Sync slave
C00367	CAN SYNC Rx identifier	0x0080			●
C00368	CAN SYNC Tx identifier	0x0080		●	
C00369	CAN sync transmission cycle time	0	ms	●	

Related topics:

▶ Synchronisation of the internal time base (912)

12.8.5 Monitoring of the RPDOs for data reception

For RPDO1 ... RPDO4 each, a monitoring time can be parameterised within which the RPDO must arrive. If the RPDO is not received within the monitoring time or not with the configured sync, the response parameterised for each RPDO takes place.

Short overview: Parameters for RPDO monitoring

Parameters	Info	Lenze setting	
		Value	Unit
C00357/1...4	CAN1...4_IN monitoring time	3000	ms
C00593/1...4	Resp. to CAN1...4_IN monitoring	No response	

12.8.6 Configuring exception handling of the CAN PDOs

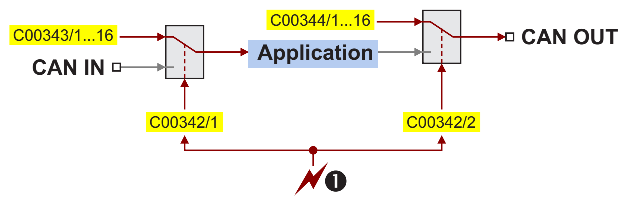
Exception handling for the CAN PDOs in the event of an error can be set via decoupling configuration and decoupling values.

- Bit coded selection is carried out in C00342/1 for the process data words read by the bus, defining the events that will trigger decoupling.
- Bit coded selection is carried out in C00342/2 for the process data words output by the application, defining the events that will trigger decoupling.

Bit	Event
Bit 0 ☐	BusOff_MsgErr
Bit 1 ☐	Warning
Bit 2 ☐	NodeStopped
Bit 3 ☐	HeartBeatEvent
Bit 4 ☐	CAN1_In_Überw.
Bit 5 ☐	CAN2_In_Überw.
Bit 6 ☐	CAN3_In_Überw.
Bit 7 ☐	CAN4_In_Überw.
Bit 8 ☐	Reserved
Bit 9 ☐	Reserved
Bit 10 ☐	Reserved
Bit 11 ☐	Reserved
Bit 12 ☐	Reserved
Bit 13 ☐	Reserved
Bit 14 ☐	Trouble
Bit 15 ☐	Fault

Finally, the following parameters define the value that the process data words are to have when they are decoupled:

Parameters	Info	Lenze setting	
		Value	Unit
C00343/1	LP_CanIn1:wCtrl DiscVal	0	
C00343/2...4	LP_CanIn1:wIn2...wIn4 DiscVal	0	
C00343/5...8	LP_CanIn2:wIn1...wIn4 DiscVal	0	
C00343/9...12	LP_CanIn3:wIn1...wIn4 DiscVal	0	
C00343/13...16	LP_CanIn4:wIn1...wIn4 DiscVal	0	
C00344/1	LP_CanOut1:wState DiscVal	0	
C00344/2...4	LP_CanOut1:wOut2...wOut4 DiscVal	0	
C00344/5...8	LP_CanOut2:wOut1...wOut4 DiscVal	0	
C00344/9...12	LP_CanOut3:wOut1...wOut4 DiscVal	0	
C00344/13...16	LP_CanOut4:wOut1...wOut4 DiscVal	0	



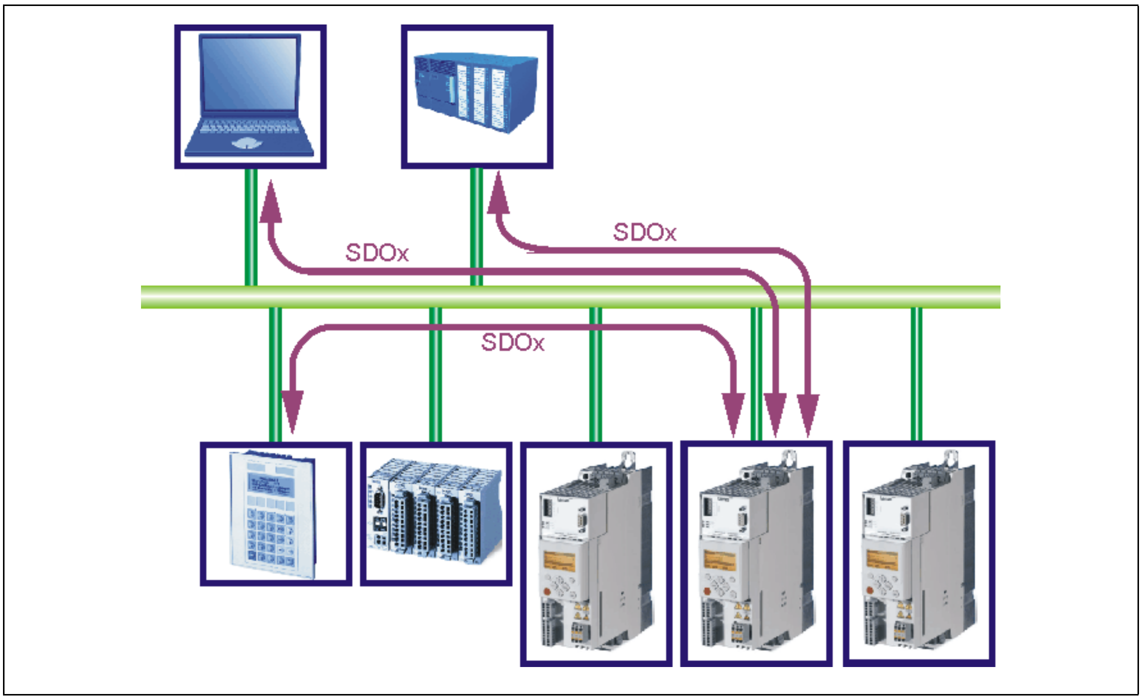
❶ Exception event (an error has occurred)

[12-7] General signal flow in the event of a configured exception

Related topics:

- Configuring exception handling of the output terminals (📖 444)

12.9 Parameter data transfer



[12-8] Parameter data transfer via the available parameter data channels

Parameters are values stored in codes on Lenze controllers.

Two parameter data channels are available for parameter setting, enabling the simultaneous connection of different devices for configuration purposes.

Parameter data are transmitted via the system bus as SDOs (*Service Data Objects*) and acknowledged by the receiver. The SDO enables read and write access to all device parameters and to the CANopen object directory integrated in the device. Indices (e.g. 0x1000) ensure access to device parameters and functions included in the object directory. To transfer SDOs, the information contained in the user data must comply with the CAN SDO protocol.

Note!

In the Lenze setting, only the parameter data channel 1 is activated according to CANopen.

- In C00366, set "2 SDO Lenze" to activate both parameter data channels.

12.9.1 Identifiers of the parameter data objects

In the Lenze setting, the basic identifiers of the SDOs are preset according to the "Predefined Connection Set".

The identifiers of the parameter data objects SDO1 and SDO2 result from the basic identifier and the node address set under code C00350:

Identifier = basic identifier + node address

Object		Direction		Lenze-Base-ID		CANopen-Base-ID	
		from device	to device	dec	hex	dec	hex
SDO1 (Parameter data channel 1)	TSDO1	●		1408	580	1408	580
	RSDO1		●	1536	600	1536	600
SDO2 (Parameter data channel 2)	TSDO2	●		1472	5C0	1472	5C0
	RSDO2		●	1600	640	1600	640
Heartbeat		●		1792	700	1792	700
Boot-up		●		1792	700	1792	700

12.9.2 User data

Structure of the user data of the parameter data telegram

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

Note!

For the user data, the Motorola format is used.

▸ Parameter data telegram examples (📖 847)

The following subchapters provide detailed information on user data.

12.9.2.1 Command

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	**Index**		**Subindex**	**Data 1**	**Data 2**	**Data 3**	**Data 4**
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

The following commands can be transmitted or received for writing and reading the parameters:

Command	1st byte hex	1st byte dec	Data length	Info
Write request	0x23	35	4 bytes	Writing of a parameter to the inverter.
	0x2B	43	2 bytes	
	0x2F	47	1 byte	
	0x21	33	Block	
Write response	0x60	96	4 bytes	Inverter acknowledges a write request.
Read request	0x40	64	4 bytes	Reading of a parameter from the inverter.
Read response	0x43	67	4 bytes	Inverter's response to a read request with the current parameter value.
	0x4B	75	2 bytes	
	0x4F	79	1 byte	
	0x41	65	Block	
Error response	0x80	128	4 bytes	Response from the inverter when the read/write request could not be executed correctly. ▸ Error messages (📖 845)

More precisely, the command byte comprises the following information:

Command	1st byte							
	Command specifier (cs)			Toggle (t)	Length*		e	s
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write request	0	0	1	0	0/1	0/1	1	1
Write response	0	1	1	0	0	0	0	0
Read request	0	1	0	0	0	0	0	0
Read response	0	1	0	0	0/1	0/1	1	1
Error response	1	0	0	0	0	0	0	0

*Bit coding of the length: 00 = 4 bytes, 01 = 3 bytes, 10 = 2 bytes, 11 = 1 byte
e: expedited (shortened block service)
s: segmented (normal block service)

Tip!

More commands are defined in CANopen specification DS301, V4.02 (e.g. segmented transfer).

12.9.2.2 Addressing by means of index and subindex

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

A parameter (a Lenze code) is addressed as per the following formula:
Index = 24575 - (Lenze code number)

Example

The C00011 parameter (motor reference speed) is to be addressed.

Calculation:

- Index:
  - Decimal: 24575 - 11 = 24564
  - Hexadecimal: 0x5FFF - 0xB = 0x5FF4
- Subindex: 0x00 (subindex 0 since the parameter does not have any subcodes)

Entries:

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	0xF4	**0x5F**	**0x00**				

12.9.2.3 Data 1 ... Data 4

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

Maximally 4 bytes are available for parameter value entries. Depending on the data format, they are assigned as follows:

5th byte	6. byte	7th byte	8th byte
Parameter value (1 byte)	0x00	0x00	0x00
Parameter value (2 bytes)		0x00	0x00
LOW byte	HIGH byte		
Parameter value (4 bytes)			
LOW word		HIGH word	
LOW byte	HIGH byte	LOW byte	HIGH byte

Note!

The "Factor" column of the Table of attributes contains a so-called scaling factor for all Lenze parameters. The scaling factor is relevant to the transfer of parameter values which have one or more decimal positions in the parameter list.

If the scaling factor is > 1, the value must be multiplied by the indicated scaling factor prior to transmission to be able to transfer the value as an integer. At the SDO client end, the integer must be divided by the scaling factor to obtain the original value including decimal positions again.

Example

A value of "123.45" is to be transmitted for a code, unit: "%" (e.g. C00039/1: "Fixed setpoint-JOG1").

In inverters of the 8400 series, parameters with the "%" unit have two decimal positions and hence a scaling factor of "100".

Calculation:

- Value to be transmitted = scaling factor x value
- Data $_{(1 \dots 4)}$ = 100 x 123.45 = 12345 (0x00 00 30 39)

Entries:

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
				0x39	**0x30**	**0x00**	**0x00**

12.9.2.4 Error messages

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	**Index**		**Subindex**	**Error code**			
0x80 (128)	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

In the event of an error, the addressed node generates a telegram with the "Error response" (0x80) command.

- The telegram includes the index and subindex of the code where the error occurred.
- The error code is entered in bytes 5 ... 8.
  - The error codes are standardised according to DS301, V4.02.
  - The representation of the error codes is provided in reverse read direction (see example below).

Example

Representation of error code "0x06 04 00 41" in bytes 5 ... 8:

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	**Index**		**Subindex**	**Error code**			
				0x41	**0x00**	**0x04**	**0x06**

Meaning of the error codes

The error codes are standardised acc. to DS301, V4.02.

Error code	Explanation
0x0503 0000	Toggle bit not changed
0x0504 0000	SDO protocol expired
0x0504 0001	Invalid or unknown client/server command specifier
0x0504 0002	Invalid block size (only block mode)
0x0504 0003	Invalid sequence number (only block mode)
0x0504 0004	CRC error (only block mode)
0x0504 0005	Not sufficient memory
0x0601 0000	Object access not supported
0x0601 0001	Attempt to read a write-only object
0x0601 0002	Attempt to write to a read-only object
0x0602 0000	Object not listed in object directory
0x0604 0041	Object not mapped to PDO
0x0604 0042	Number and length of objects to be transferred longer than PDO length.
0x0604 0043	General parameter incompatibility
0x0604 0047	General internal device incompatibility
0x0606 0000	Access denied because of hardware error
0x0607 0010	Unsuitable data type, unsuitable service parameter length
0x0607 0012	Unsuitable data type, service parameter length exceeded
0x0607 0013	Unsuitable data type, service parameter length not long enough
0x0609 0011	Subindex does not exist
0x0609 0030	Parameter value range exceeded
0x0609 0031	Parameter values too high
0x0609 0032	Parameter values too low
0x0609 0036	Maximum value falls below minimum value
0x0800 0000	General error
0x0800 0020	Data cannot be transferred/saved for application.
0x0800 0021	Data cannot be transferred/saved for application due to local control.
0x0800 0022	Data cannot be transferred/saved for application due to current device status.
0x0800 0023	Dynamic generation of object directory failed or no object directory available (e.g. object directory generated from file, generation not possible because of a file error).

12.9.3 Parameter data telegram examples

12.9.3.1 Read parameters

Task: The heatsink temperature of 43 °C (code C00061, data format INTEGER16, scaling factor 1) of the inverter with node address "5" is to be read.

Telegram to drive

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0605	0x40	0xC2	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 5 = 1541 = 0x0605 (1536 = SDO1 basic identifier to the inverter)
Command	= 0x40 = "Read request" (read request of a parameter from the inverter)
Index	= 24575 - code number = 24575 - 61 = 24514 = 0x5FC2
Subindex	= 0 (code C00061 does not have any subcodes)

Response telegram from drive (if data have been correctly transmitted)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	-	-
0x0585	0x4B	0xC2	0x5F	0x00	0x2B	0x00	-	-

Explanations on the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 5 = 1413 = 0x0585 (1408 = SDO1 basic identifier from the inverter)
Command	= 0x4B = "Read Response" (response to the read request with current value)
Index	as in telegram to the drive
Subindex	
Data 1 ... 2	= 0x002B = 43 [°C]

12.9.3.2 Write parameters

Task: The rated current of the connected motor is to be entered with I_N = 10.20 A (code C00088) into the inverter with node address "2".

Data 1 ... 4	Calculation
Value for motor current, (data type U16; display factor 1/100)	10.20 x 100 = 1020 (0x03 FC)

Telegram to drive

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0602	0x23	0xA7	0x5F	0x00	0xFC	0x03	0x00	0x00

Explanations on the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 2 = 1538 = 0x0602 (1536 = SDO1 basic identifier to the inverter)
Command	= 0x23 = "Write request" (write request of a parameter to the inverter)
Index	= 24575 - code number = 24575 - 88 = 24487 = 0x5FA7
Subindex	= 0 (code C00088 does not have any subcodes)
Data 1 ... 4	= 10.20 x 100 = 1020 = 0x000003FC (motor current value; data type U32; display factor 1/100)

Response telegram from drive (if data have been correctly transmitted)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0582	0x60	0xA7	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 2 = 1410 = 0x0582 (1408 = SDO1 basic identifier from the inverter)
Command	= 0x60 = "Write response" (Acknowledgement of the write access from the inverter)
Index	as in telegram to the drive
Subindex	

12.9.3.3 Read block parameters

Task: The firmware version (code C00099) is to be read from the parameter set of the inverter with node address "12". The firmware version has a length of 11 ASCII characters which are transmitted as a block parameter. Depending on the block, the data width from the 2nd to 8th byte is assigned within the user data.

Telegram 1 to the drive: Read request

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x060C	0x40	0x9C	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 12 = 1548 = 0x060C (1536 = SDO1 basic identifier to the inverter)
Command	= 0x40 = "Read request" (read request of a parameter from the inverter)
Index	= 24575 - code number = 24575 - 99 = 24476 = 0x5F9C
Subindex	= 0 (code C00099 does not have any subcodes)

Response telegram 1 from the drive: Indication of the block length (11 characters)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x058C	0x41	0x9C	0x5F	0x00	0x0B	0x00	0x00	0x00

Explanations on the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 12 = 1420 = 0x058C (1408 = SDO1 basic identifier from the inverter)
Command	= 0x41 = "Read response" (response is block telegram)
Index	as in telegram to the drive
Subindex	
Data 1 ... 4	= 0x0000000B = data length of 11 characters in the ASCII format

Telegram 2 to the drive: Request of the 1st data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x060C	0x60	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram to the drive	
Command	= 0x60 = "Read segment request" (request: read data block) • Bit 4 = 0 (toggle bit)
	Influence of the toggle bit on the request command The single blocks are toggled one after another, i.e. first the request is made with the "0x60" command (= 0b011**0***0000), then with the "0x70" command (= 0b011**1***0000), and then again with the "0x60" command, etc. * Toggle bit

Response telegram 2 from the drive: Transmission of the 1st data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x058C	0x00	0x30	0x31	0x2E	0x30	0x30	0x2E	0x30
		0_{asc}	1_{asc}	$._{asc}$	0_{asc}	0_{asc}	$._{asc}$	0_{asc}

Explanations on the telegram to the drive	
Command	= 0x00 = 0b00000000 • Bit 4 = 0 (toggle bit)
	Influence of the toggle bit on the transmission command • The 1st response of the inverter in the command byte is "0b000**0***0000" if bytes 2 ... 8 are completely filled with data and other telegrams are following. • The 2nd response of the inverter in the command byte is "0b0001**1***0000" if bytes 2 ... 8 are completely filled with data and other telegrams are following. * Toggle bit
Data 1 ... 7	= "01.00.0" (ASCII representation)

Telegram 3 to the drive: Request of the 2nd data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x060C	0x70	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Explanations on telegram 3 to the drive	
Command	= 0x70 = "Read segment request" (request: read data block) • Bit 4 = 1 (toggle bit)

Response telegram 3 from the drive: Transmission of the 2nd data block including end identifier

Identifier	User data							
	1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x058C	0x17	0x30	0x2E	0x30	0x30	0x00	0x00	0x00
		0_{asc}	$._{asc}$	0_{asc}	0_{asc}	-	-	-

Explanations on telegram 3 from the drive	
Command	= 0x17 = 0b00010111: • Bit 0 = 1 (end of transmission) • Bit 1 ... bit 3 = 0b011 (3 bytes do not contain any data) • Bit 4 = 1 (toggle bit)
	Influence of the final bit and the residual data length on the transmission command • The end of transmission is signalled via the set final bit 0. • Bits 1 ... 3 reveal the number of bytes that do not contain any data anymore. * Toggle bit
Data 1 ... 7	= "0.00" (ASCII representation) The result of the data block transmission is: "01.00.00.00"

12.10 Monitoring

12.10.1 Integrated error detection

If a node detects an error, it rejects the telegram bits received so far and transmits an error flag. The error flag consists of 6 consecutive bits with the same logic value.

The following errors are detected:

Error	Description
Bit error	The sending node follows the transmission on the bus and interrupts the transmission if it receives a different logic value than the value transmitted. With the next bit, the sending node starts the transmission of an error flag. In the arbitration phase, the transmitter only detects a bit error if a dominantly sent bit is received as recessive bit. In the ACK slot as well, the dominant overwriting of a recessive bit is not indicated as a bit error.
Stuff-bit error	If more than 5 consecutive bits have the same logic value before the ACK delimiter in the telegram, the previously transmitted telegram will be rejected and an error flag will be sent with the next bit.
CRC error	If the received CRC checksum does not correspond to the checksum calculated in the bus controller, the bus controller will send an error flag after the ACK delimiter and the previously transmitted telegram will be annulled.
Acknowledgement error	If the sent ACK slot recessively sent by the transmitting node is not dominantly overwritten by a receiver, the transmitting node will cancel the transmission. The transmitting node will annul the transmitted telegram and will send an error flags with the next bit.
Format error	If a dominant bit is detected in the CRC delimiter, in the ACK delimiter or in the first 6 bits of the EOF field, the received telegram will be rejected and an error flag will be sent with the next bit.

Tip!

The errors mentioned before indicate that a physical error has occurred in the bus system.

Possible causes are:

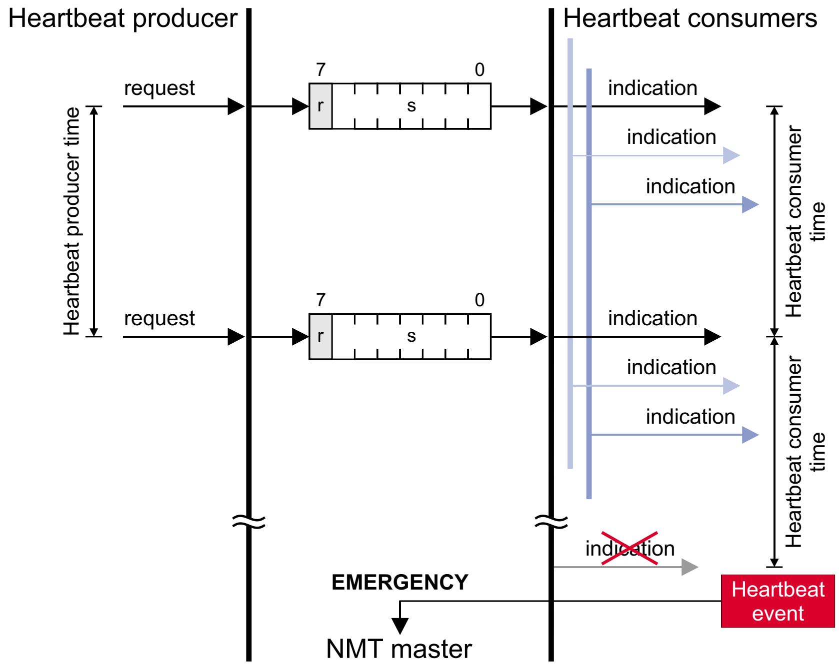
- Several nodes with identical node address
- Wrong baud rate of one or several nodes
- Too high cable length
- Too many or no terminating resistors
- Too high bus load/too many data telegrams
  (e.g. since a node permanently transmits event-controlled due to data changes of an analog signal/actual value.)
- EMC interferences on the system bus
  (e.g. since the CAN bus cable next to the motor cable is unshielded.)

C00364 displays whether such an error is active.

12.10.2 Heartbeat protocol

The heartbeat protocol can be used for node monitoring purposes within a CAN network.

Basic workflow



r: Reserved (always 0)
s: Status of the producer (0: Boot-up, 4: Stopped, 5: Operational, 127: Pre-Operational)

[12-9] Heartbeat protocol

1. A heartbeat producer cyclically transmits a so-called heartbeat telegram to one or more consumers.
2. The consumer(s) monitor the heartbeat telegram for arrival on a regular basis.

12.10.2.1 Telegram structure

- The heartbeat telegram of the producer has the following identifier:
  Identifier (COB-ID) = 1792 + producer's node address
- The user data (1 byte) contain the status (s) of the producer:

Heartbeat producer status		Data							
Communication status	Decimal value (s)	(r)	Producer status (s)						
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Boot-up	0	0	0	0	0	0	0	0	0
Stopped	4	0	0	0	0	0	1	0	0
Operational	5	0	0	0	0	0	1	0	1
Pre-Operational	127	0	1	1	1	1	1	1	1

12.10.2.2 Parameter setting

Short overview of the parameters for the "Heartbeat" monitoring function:

Parameters	Info	Lenze setting		Assignment	
		Value	Unit	Consumer	Producer
C00347/1...n	CAN status of the heartbeat producer 1 ... n	-		●	
C00381	Heartbeat producer time	0	ms		●
C00385/1...n	CAN node address of the heartbeat producer 1 ... n	0		●	
C00386/1...n	Heartbeat consumer time for the heartbeat producer 1 ... n	0	ms	●	
C00592/5	Resp. to heartbeat event	No response		●	
Greyed out = display parameter					

Heartbeat producer time

Time interval for the transmission of the heartbeat telegram to the consumer(s).

- Parameterisable in C00381 or via object I-1017. The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is sent automatically as soon as a time > 0 ms is set.

Heartbeat consumer time

Monitoring time for the nodes (producers) to be monitored.

- Parameterisable in C00386/1...n or via object I-1016.
- The parameterised time is rounded down to an integer multiple of 5 ms and must have a greater value than the heartbeat producer time of the node to be monitored.
- The maximum number of the nodes to be monitored depends on the device version:
  - "BaseLine C": 1 Heartbeat Producer can be monitored.
  - "StateLine C": Up to 7 Heartbeat Producers can be monitored.
  - "HighLine C": Up to 15 Heartbeat Producers can be monitored.
  - "TopLine C": Up to 15 Heartbeat Producers can be monitored.
- The node address(es) of the nodes to be monitored is/are set in C00385/1...n or via object I-1016, too.

Heartbeat event

The "Heartbeat event" is activated in the consumer if it does not receive any heartbeat telegram from the producer within the heartbeat consumer time:

- The consumer changes from the "Operational" communication status to the "Pre-Operational" communication status.
- The NMT master receives an emergency telegram containing emergency error code 0x8130.
- The response parameterised in C00592/5 is activated (Lenze setting: "No response").

Note!

The heartbeat monitoring will not start until the first heartbeat telegram of a monitored producer has been received successfully and the "Pre-Operational" NMT status has been assumed.

The boot-up telegram counts as the first heartbeat telegram.

12.10.2.3 Commissioning example

Task

An 8400 inverter (node 2) which is configured as heartbeat consumer is to monitor another 8400 inverter (heartbeat producer, node 1).

- The heartbeat producer is to transmit a heartbeat telegram to the heartbeat consumer every 10 ms.
- The heartbeat consumer monitors the heartbeat telegram for arrival. A response is to be activated in the event of an error.

Parameterising the heartbeat producer (node 1)

1. Set the heartbeat producer time (C00381) to 10 ms.

Parameterising the heartbeat consumer (node 2)

1. Set the CAN node address of the producer in C00385/1.
2. Set the heartbeat consumer time in C00386/1.
   - Note: The heartbeat consumer time must be greater than the heartbeat producer time of the node to be monitored set in C00381.
3. Set the desired response in C00592/5 which is to be activated should a heartbeat event in the consumer occur.

Tip!

C00347/1...n displays the heartbeat status of the monitored nodes.

Heartbeat telegram

- The heartbeat telegram of the producer has the following identifier:
  Identifier (COB-ID) = 1792 + producer's node address = 1792 + 1 = 1793 = 0x701

12.10.3 Emergency telegram

If the error status changes because an internal device error occurs or has been eliminated, the NMT master receives an emergency telegram once with the following structure:

<table>
<tr><th>1st byte</th><th>2nd byte</th><th>3rd byte</th><th>4. byte</th><th>5th byte</th><th>6. byte</th><th>7th byte</th><th>8th byte</th></tr>
<tr><th colspan="2">Emergency error codes</th><th>Error register</th><th colspan="5">Manufacturer-specific error message</th></tr>
<tr><td rowspan="2">LOW byte</td><td rowspan="2">HIGH byte</td><td rowspan="2">I-1001</td><td rowspan="2">0x00
(Reserved)</td><td colspan="2">LOW word</td><td colspan="2">HIGH word</td></tr>
<tr><td>LOW byte</td><td>HIGH byte</td><td>LOW byte</td><td>HIGH byte</td></tr>
<tr><td colspan="3">See table below</td><td colspan="5">• For emergency error code 0xF000: Lenze error number (value displayed in C00168)
• All other emergency error codes have a value of "0" here.</td></tr>
</table>

<table>
<tr><th>Emergency error codes</th><th>Error register</th><th>Cause</th></tr>
<tr><td rowspan="2">0x0000</td><td>0xXX</td><td>One of several errors eliminated</td></tr>
<tr><td>0x00</td><td>One error has been eliminated (error-free status afterwards)</td></tr>
<tr><td>0x3100</td><td>0x01</td><td>Supply voltage of standard device faulty or failed</td></tr>
<tr><td>0x8100</td><td>0x11</td><td>Communication error (warning)</td></tr>
<tr><td>0x8130</td><td>0x11</td><td>Life guarding error or heartbeat error</td></tr>
<tr><td>0x8150</td><td>0x11</td><td>Collision of identifiers (COB-IDs): An identifier parameterised for reception is also used for transmission.</td></tr>
<tr><td>0x8210</td><td>0x11</td><td>PDO length shorter than expected</td></tr>
<tr><td>0x8220</td><td>0x11</td><td>PDO length greater than expected</td></tr>
<tr><td>0x8700</td><td>0x11</td><td>Monitoring of the sync telegram</td></tr>
<tr><td>0xF000</td><td>0x01</td><td>Generic error
• An error with a "Fault", "Trouble", "TroubleQSP", "Warning" error response occurred in the standard device.
• Error message is the Lenze error number (C00168).</td></tr>
</table>

The Short overview (A-Z) of error messages of the operating system includes a list of more emergency error codes. (📖 749)

Example

<table>
<tr><th>1st byte</th><th>2nd byte</th><th>3rd byte</th><th>4. byte</th><th>5th byte</th><th>6. byte</th><th>7th byte</th><th>8th byte</th></tr>
<tr><th colspan="2">Emergency error codes</th><th>Error register</th><th colspan="5">Manufacturer-specific error message</th></tr>
<tr><td>0x00</td><td>0xF0</td><td rowspan="2">0x01</td><td rowspan="2">0x00
(Reserved)</td><td colspan="4">Lenze error number</td></tr>
<tr><td colspan="2">Generic error</td><td colspan="4">▶ Error messages of the operating system
Corresponding error-free message: Value "0x00000000"</td></tr>
</table>

Tip!

A detalied description can be found in CAN specification DS301, V4.02.

12.11 CANopen objects implemented

Lenze devices can be parameterised with both Lenze codes and manufacturer-independent "CANopen objects". Fully CANopen-compliant communication can only be achieved by exclusively using CANopen objects for the parameterisation. The CANopen objects described in this chapter are defined in the DS301 V4.02 CAN specification.

Many CANopen objects can be mapped on Lenze codes. In the following table, the corresponding Lenze codes are listed in the column "Relationship to Lenze codes".

Note!

Some of the terms used here derive from the CANopen protocol.

Overview of CANopen indices and their relationship to Lenze codes

CANopen object			Relationship to Lenze code
Index	Subindex	Name	
I-1000	0	Device type	-
I-1001	0	Error register	-
I-1003	Predefined error field		
	0	Number of errors	-
	1 ... 10	Standard error field	-
I-1005	0	COB-ID SYNC message	C00367
			C00368
I-1006	0	Communication cycle period	C00369
I-1014	0	COB-ID EMCY	-
I-1016	Consumer heartbeat time		
	0	Highest subindex supported	-
	1 ... n	Consumer heartbeat time • "BaseLine C" version: n = 1 • "StateLine C" version: n = 7 • "HighLine C" version: n = 15 • "TopLine C" version: n = 15	C00385/1...n C00386/1...n
I-1017	0	Producer heartbeat time	C00381
I-1018	Identity object		
	0	Highest subindex supported	-
	1	Vendor ID	-
	2	Product code	-
	3	Revision number	-
	4	Serial number	-
I-1200	SDO1 server parameter		
	0	Highest subindex supported	-
	1	COB-ID client –> server (rx)	-
	2	COB-ID server –> client (tx)	-
I-1201	SDO2 server parameter		
	0	Highest subindex supported	-
	1	COB-ID client –> server (rx)	-
	2	COB-ID server –> client (tx)	-

CANopen object			Relationship to Lenze code
Index	Subindex	Name	
I-1400	RPDO1 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by RPDO	C00355/1
	2	Transmission type	C00323/1
I-1401	RPDO2 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by RPDO	C00355/3
	2	Transmission type	C00323/2
I-1402	RPDO3 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by RPDO	C00355/5
	2	Transmission type	C00323/3
I-1403	RPDO4 communication parameter (from version 15.00.00)		
	0	Highest subindex supported	-
	1	COB-ID used by RPDO	C00355/7
	2	Transmission type	C00323/4
I-1600	RPDO1 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00409/1...4 C00866/1...4
I-1601	RPDO2 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00409/5...8 C00866/5...8
I-1602	RPDO3 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00409/9...12 C00866/9...12
I-1603	RPDO4 mapping parameter (from version 15.00.00)		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00409/13...16 C00866/13...16
I-1800	TPDO1 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by TPDO	C00355/2
	2	Transmission type	C00322/1
	3	Inhibit time	C00324/2
	5	Event timer	C00356/5 C00369

CANopen object			Relationship to Lenze code
Index	Subindex	Name	
I-1801	TPDO2 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by TPDO	C00355/4
	2	Transmission type	C00322/2
	3	Inhibit time	C00324/3
	5	Event timer	C00356/2 C00369
I-1802	TPDO3 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by TPDO	C00355/6
	2	Transmission type	C00322/3
	3	Inhibit time	C00324/4
	5	Event timer	C00356/3 C00369
I-1803	TPDO4 communication parameter (from version 15.00.00)		
	0	Highest subindex supported	-
	1	COB-ID used by TPDO	C00355/8
	2	Transmission type	C00322/4
	3	Inhibit time	C00324/5
	5	Event timer	C00356/4 C00369
I-1A00	TPDO1 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00868/1...4
I-1A01	TPDO2 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00868/5...8
I-1A02	TPDO3 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00868/9...12
I-1A03	TPDO4 mapping parameter (from version 15.00.00)		
	0	Number of mapped application objects in PDO	-
	1 ... 4	Application object 1 ... 4	C00868/13...16

I-1000

Index **I-1000**	Name: **Device type**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Device type	0	0		4294967295	ro	U32

The CANopen index I-1000 specifies the profile for this device. Furthermore, additional information defined in the device profile itself can be stored here.

8th byte	7th byte	6. byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
HIGH word		LOW word	
HIGH byte	LOW byte	HIGH byte	LOW byte
Additional information		**ECAT: Device Profile Number**	

[12-1] Data frame assignment

In case of 8400 series inverters, the four bytes contain the following values:

- 5th and 6th byte: The data content is 0x0000, i.e. no profile definition.
- 7th byte: The data content specifies the device type: Here the value is 0x00 for inverters.
- 8th byte: The data content is 0x00.

The data content for the 8400 inverter thus is: 00 00 00 00

I-1001

Index: **I-1001**	Name: **Error register**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Error register	-	0		255	ro	U8

Error register

The error status in the data byte (U8) is bit coded. The following error states are coded in the data byte (U8):

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Error state
0	0	0	0	0	0	0	0	No error
0	0	0	0	0	0	0	1	Device error message
0	0	0	1	0	0	0	1	Communication error

I-1003

Index: **I-1003**	Name: **Predefined error field**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of errors	0	0		255	rw	U8
1 ... 10: Standard error field	-	0		4294967295	ro	U32

Error history

This object indicates that an error has occurred in the module and in the standard device.

Subindex	Meaning
0	Number of saved error messages
1 ... 10	Display of the error list The error messages (U32) consist of a 16-bit error code and a manufacturer-specific information field comprising 16 bits.

Note!

The values in the "standard error field" under subindex 1 ... 10 will be deleted if the subindex "number of recorded errors" is overwritten with the value "0".

Emergency error codes	Cause	Entry in the error register (I-1001)
0x0000	One of several errors eliminated	0xXX
	Elimination of one single error (afterwards no more errors)	0x00
0x1000	Standard device is in error status (error response "fault", "message", "warning", "error", "quick stop by trouble")	0x01
0x3100	Supply voltage of standard device faulty or failed	0x01
0x8100	Communication error (warning)	0x11
0x8130	Life guard error or heartbeat error	0x11
0x8150	Collision of COB-IDs: An ID parameterised for reception is also used for transmission.	0x11
0x8210	PDO length shorter than expected	0x11
0x8220	PDO length greater than expected	0x11
0x8700	Monitoring of the sync telegram	0x11

I-1005

Index: **I-1005**	Name: **COB-ID SYNC message**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: COB-ID SYNC message	0x0000 0080 or 0x8000 0080	0		4294967295	rw	U32

This object can be used to activate the generation of sync telegrams and to write the identifier value.

- This object relates to codes C00367 and C00368.

Creating sync telegrams

To create sync telegrams, bit 30 (see below) must be set to "1". The interval of the sync telegrams can be set with the object I-1006.

Writing identifiers

To receive PDOs, the value 0x80 must be entered in the 11-bit identifier in the Lenze setting (and according to CANopen specification) . This means that all modules are by default set to the same sync telegram.

- If sync telegrams are only to be received by certain communication modules, their identifiers can be entered with values up to and including 0x07FF.
- The identifier can only be changed if the communication module does not send any sync telegrams (bit 30 = "0").
- How to change the identifier:
  - Deactivate identifier (set bit 30 to "0").
  - Change identifier.
  - Activate identifier (set bit 30 to "1").

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
x	**0/1**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[12-2] Data frame assignment

I-1006

Index: **I-1006**	Name: **Communication cycle period**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Communication cycle period	0 µs	0	µs	65535000	rw	U32

Setting the sync telegram cycle time.

- The cycle time can be selected as "1000" or as an integer multiple of it.
- If "0 µs" is set (Lenze setting), no sync telegrams are created.
- This object relates to code C00369.

I-1014

Index: **I-1014**	Name: **COB-ID EMCY**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: COB-ID EMCY	0x80 + node ID	0		4294967295	rw	U32

When communication errors occur and are acknowledged or when internal errors occur in the communication module or inverter (e.g. "fault"), the system bus sends an error message. The telegram is sent once for every error. This function can be activated or deactivated with bit 31.

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[12-3] Data frame assignment

Bit	setting	
Bit 31	0	Emergency object is valid.
	1	Emergency object is invalid.

Note!

The identifier can only be changed in the "emergency object invalid" status (bit 31 = 1).

I-1016

Index: **I-1016**	Name: **Consumer heartbeat time**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	BaseLine C: 1 StateLine C: 7 HighLine C: 15 TopLine C: 15	- (read access only)			ro	U16
1 ... n: Consumer heartbeat time	0	0		65535	rw	U16

Monitoring time for the nodes to be monitored via heartbeat. Heartbeat protocol (853)

- The parameterised time is rounded down to an integer multiple of 5 ms and must have a greater value than the heartbeat producer time of the node to be monitored.

Subindex	Meaning	Lenze code
0	Number of nodes to be monitored	
1 ... n	Node ID and heartbeat time of the node to be monitored	Node ID: C00385/x Heartbeat time: C00386/x

8th byte	7th byte	6. byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 24	Bit 23 ... Bit 16	Bit 15 ... Bit 0	
0 (Reserved)	**Node ID**	**Heartbeat time** in [ms]	

[12-4] Data frame assignment

I-1017

Index: **I-1017**	Name: **Producer heartbeat time**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Producer heartbeat time	0	0	ms	65535	rw	U16

Time interval for sending the heartbeat telegram to the consumer(s). ▶ Heartbeat protocol (📖 853)

- The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is automatically sent as soon as a time > 0 ms is entered. In this case, the "node guarding" monitoring function is deactivated.
- This object relates to code C00381.

I-1018

Index: **I-1018**	Name: **Identity object**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	see below	0		4294967295	ro	U32
1: Vendor ID						
2: Product code						
3: Revision number						
4: Serial number						

Subindex	Meaning	
1	Manufacturer's identification number • The identification number allocated to Lenze by the organisation "CAN in Automation e. V." is "0x0000003B".	
2	Product code	
	84001	8400 BaseLine C
	84002	8400 StateLine C
	84003	8400 HighLine C
	84004	8400 TopLine C
3	Main and subversion of firmware	
4	Serial number	

I-1200

Index: **I-1200**	Name: **SDO1 server parameter**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	2	2		2	ro	U8
1: COB-ID client -> server (rx)	node ID + 0x600	0		4294967295	ro	U32
2: COB-ID server -> client (tx)	node ID + 0x580	0		4294967295	ro	U32

Identifiers for SDO server channel 1 (basic SDO channel).

- According to DS301 V4.02, the basic SDO channel can neither be changed nor deactivated.

Subindex	Meaning
1	Specification of receive identifier • For SDO server channel 1: node address (C00350) + 0x600
2	Specification of send identifier • For SDO server channel 1: node address (C00350) + 0x580

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[12-5] Data frame assignment

I-1201

Index: **I-1201**	Name: **SDO2 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32

Identifiers for SDO server channel 2.

- The SDO server parameter is only valid, if bit 31 is set to "0" for both transmission directions (subindex 1 and 2).
- In the Lenze setting, the SDO server channels 2 are deactivated (bit 31 = "1").
- The identifier can only be changed if the SDO is invalid (bit 31 = "1").

Subindex	Meaning
1	Specification of receive identifier
2	Specification of send identifier

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[12-6] Data frame assignment

Bit	setting	
Bit 31	0	SDO is valid.
	1	SDO is invalid.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

Example

Parameter data channel 2 of the inverter with node address 4 shall be activated.

- For this purpose, bit 31 in the subindexes 1 and 2 of the I-1201 object must be set to the value "0" (≡ "SDO valid").
- The master must send the two "write request" commands to the nodes via the basic SDO channel.

Identifier calculation

- Identifier (COB-ID) = basic identifier + node address (node ID)
- Basic identifier SDO2 from master to drive: 1600 (0x640)
  ➞ Identifier = 0x640 + 0x4 = 0x644
- Basic identifier SDO2 from drive to master: 1472 (0x5C0)
  ➞ Identifier = 0x5C0 + 0x4 = 0x5C4

Resulting data (data 1 ... data 4)

<table>
<tr><th colspan="3">8th byte</th><th>7th byte</th><th colspan="2">6. byte</th><th>5th byte</th></tr>
<tr><th colspan="3">Data 4</th><th>Data 3</th><th colspan="2">Data 2</th><th>Data 1</th></tr>
<tr><td>Bit 31</td><td>Bit 30</td><td colspan="3">Bit 29 ... bit 11</td><td colspan="2">Bit 10 ... bit 0</td></tr>
<tr><td>0</td><td>0</td><td colspan="3">Extended identifier = 0</td><td colspan="2">11-bit identifier = 0x644</td></tr>
<tr><td colspan="3">0x00</td><td>0x00</td><td colspan="2">0x06</td><td>0x44</td></tr>
</table>

[12-7] Data telegram assignment for subindex 1

<table>
<tr><th colspan="3">8th byte</th><th>7th byte</th><th colspan="2">6. byte</th><th>5th byte</th></tr>
<tr><th colspan="3">Data 4</th><th>Data 3</th><th colspan="2">Data 2</th><th>Data 1</th></tr>
<tr><td>Bit 31</td><td>Bit 30</td><td colspan="3">Bit 29 ... bit 11</td><td colspan="2">Bit 10 ... bit 0</td></tr>
<tr><td>0</td><td>0</td><td colspan="3">Extended identifier = 0</td><td colspan="2">11-bit identifier = 0x5C4</td></tr>
<tr><td colspan="3">0x00</td><td>0x00</td><td colspan="2">0x05</td><td>0xC4</td></tr>
</table>

[12-8] Data telegram assignment for subindex 2

User data assignment

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	**Index**		**Subindex**	**Data 1**	**Data 2**	**Data 3**	**Data 4**
0x23	**0x01**	**0x12**	**0x01**	**0x44**	**0x06**	**0x00**	**0x00**

[12-9] User data assignment for writing to subindex 1

1st byte	2nd byte	3rd byte	4. byte	5th byte	6. byte	7th byte	8th byte
Command	**Index**		**Subindex**	**Data 1**	**Data 2**	**Data 3**	**Data 4**
0x23	**0x01**	**0x12**	**0x02**	**0xC4**	**0x05**	**0x00**	**0x00**

[12-10] User data assignment for writing to subindex 2

I-1400

Index: **I-1400**	Name: **RPDO1 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x200 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO1

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	RPDO1 identifier • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x200 + node ID	C00354/1
2	RPDO Transmission type according to DS301 V4.02 Transmission type (835)	C00323/1

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0/1**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[12-11] Data frame assignment

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

Description of subindex 1

Bit no.	Value	Explanation
0 ... 10	0/1	11-bit identifier
(11 ... 28)*	0	*) The extended identifier (29 bits) is not supported. Any of these bits must be "0".
29*	0	
30	0	RTR to this PDO possible (cannot be set)
	1	RTR to this PDO not possible (Lenze)
31	0	PDO active
	1	PDO not active

[12-12] I-1400 ... I-1402, subindex 1

Description of subindex 2

PDO transmission			Transmission type	Explanation
cyclic	synchronous	event-controlled		
X	X		n = 1 ... 240	When a value n is entered, this PDO will be accepted with every nth SYNC.
		X	n = 254	PDO will be accepted immediately.

[12-13] I-1400 ... I-1402, subindex 2

I-1401

Index: **I-1401**	Name: **RPDO2 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x300 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO2

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	RPDO2 identifier • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x300 + node ID	C00354/3
2	RPDO Transmission type according to DS301 V4.02 ▶ Transmission type (📖 835)	C00323/2

- For data telegram assignment and description of subindices 1 and 2, see object I-1400.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1402

Index: **I-1402**	Name: **RPDO3 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x400 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO3

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	RPDO3 identifier • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x400 + node ID	C00354/5
2	RPDO transmission type according to DS301 V4.02 Transmission type (835)	C00323/3

- For data telegram assignment and description of subindices 1 and 2, see object I-1400.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1403

From version 15.00.00

Index: **I-1403**	Name: **RPDO4 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x500 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO4

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	Identifier RPDO4 • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x500 + node ID	C00354/7
2	RPDO transmission type according to DS301 V4.02 ▸ Transmission type (📖 835)	C00323/4

- For data telegram assignment and description of subindices 1 and 2, see object I-1400.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1600

Index: **I-1600**	Name: **RPDO1 mapping parameter**				
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1600 serves to receive parameter data as RPDO1.

- This object relates to codes C00409/1...4 and C00866/1...4.
- From version 12.00.00 onwards: For 32-bit values, this object relates to codes C00407/1...2.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for RPDO1 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

8th byte	7th byte	6. byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 16		Bit 15 ... bit 8	Bit 7 ... bit 0
Index		**Subindex**	**Length**

[12-14] Data frame assignment

IEC 61131 process data words are mapped. Only whole bytes can be mapped (1-byte/mapping entry).

Related topics:

- RPDO1 | Port block "LP_CanIn1" (📖 821)

I-1601

Index: **I-1601**	Name: **RPDO2 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1601 serves to receive parameter data as RPDO2.

- This object relates to codes C00409/5...8 and C00866/5...8.
- From version 12.00.00 onwards: For 32-bit values, this object relates to codes C00407/3...4.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for RPDO2 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

- For assignment of the data telegram see object I-1600.

Related topics:

▶ RPDO2 | "LP_CanIn2" port block (📖 823)

I-1602

Index: **I-1602**	Name: **RPDO3 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1602 serves to receive parameter data as RPDO3.

- This object relates to codes C00409/9...12 and C00866/9...12.
- From version 12.00.00 onwards: For 32-bit values, this object relates to codes C00407/5...6.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for RPDO3 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

- For assignment of the data telegram see object I-1600.

Related topics:

▶ RPDO3 | "LP_CanIn3" port block (📖 825)

I-1603

From version 15.00.00

Index: **I-1603**	Name: **RPDO4 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1603 serves to receive parameter data as RPDO4.

- This object relates to codes C00409/13...16 and C00866/13...16.
- For 32-bit values, this object relates to the codes C00407/7...8.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for RPDO4 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

- For assignment of the data telegram see object I-1600.

Related topics:

▸ RPDO4 | "LP_CanIn4" port block (📖 827)

I-1800

Index: **I-1800**	Name: **TPDO1 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x180 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO1

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	TPDO1 identifier • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x180 + node ID	C00354/2
2	TPDO transmission type according to DS301 V4.02 ▸ Transmission type (835)	C00322/1
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/2
5	Cycle time for PDO transmission with transmission type "254".	C00356/5 C00369

<table>
<tr><th colspan="3">8th byte</th><th>7th byte</th><th colspan="2">6. byte</th><th>5th byte</th></tr>
<tr><th colspan="3">Data 4</th><th>Data 3</th><th colspan="2">Data 2</th><th>Data 1</th></tr>
<tr><td>Bit 31</td><td>Bit 30</td><td colspan="3">Bit 29 ... bit 11</td><td colspan="2">Bit 10 ... bit 0</td></tr>
<tr><td>0/1</td><td>0/1</td><td colspan="3">Extended identifier*</td><td colspan="2">11-bit identifier</td></tr>
<tr><td colspan="7">* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".</td></tr>
</table>

[12-15] Data frame assignment

Bit	setting	
Bit 30	0	RTR to this PDO possible (Lenze).
	1	RTR to this PDO not possible (not adjustable)
Bit 31	0	PDO active
	1	PDO inactive

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

Subindex 2 - transmission type

PDO transmission			Transmission type	Explanation
cyclic	synchronous	event-controlled		
●	●		n = 1 ... 240	When a value n is entered, this PDO will be accepted with every nth SYNC.
	●		n = 252	On sync, the PDO is filled with new data, but only sent on RTR.
		●	n = 254, 255	Event-controlled or cyclic

Subindex 3 - inhibit time

Note!

The delay time can only be changed when the PDO is not active (see subindex 1, bit 31 = 1).

The entered value multiplied by 0.1 gives the delay time in [ms]. Only integers will be considered, i.e. fractional numbers will be **rounded down** to integers.

Example:

- Entered value: 26
- Calculated time = 26 x 0.1 [ms] = 2.6 [ms] ➜ delay time = 2 [ms]

Subindex 5 - event timer

For cyclic operation (transmission type 254), the cycle time for sending the process data object on the system bus can be set under subindex 5:

The entered value corresponds to the time in [ms].

I-1801

Index: **I-1801**	Name: **TPDO2 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x280 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO2

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	TPDO2 identifier • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x280 + node ID	C00354/4
2	TPDO transmission type according to DS301 V4.02 ▶ Transmission type (835)	C00322/2
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/3
5	Cycle time for PDO transmission with transmission type "254".	C00356/2 C00369

- For data telegram assignment and description of subindices, see object I-1800.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1802

Index: **I-1802**	Name: **TPDO3 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x380 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO3

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	TPDO3 identifier • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x380 + node ID	C00354/6
2	TPDO transmission type according to DS301 V4.02 ▸ Transmission type (📖 835)	C00322/3
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/4
5	Cycle time for PDO transmission with transmission type "254".	C00356/3 C00369

- For data telegram assignment and description of subindices, see object I-1800.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1803

From version 15.00.00

Index: **I-1803**	Name: **TPDO4 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x480 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO4

Subindex	Meaning	Code
0	"5" is permanently set. • Max. 5 subindices are supported.	-
1	Identifier TPDO4 • According to the "Predefined Connection Set", the basic setting is: Identifier = 0x480 + node ID	C00354/8
2	TPDO transmission type according to DS301 V4.02 ▶ Transmission type (835)	C00322/4
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/5
5	Cycle time for PDO transmission with transmission type "254".	C00356/4 C00369

- For data telegram assignment and description of subindices, see object I-1800.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1A00

Index: **I-1A00**	Name: **TPDO1 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1A00 serves to send parameter data as TPDO1.

- This object relates to code C00868/1...4.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for TPDO1 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

8th byte	7th byte	6. byte	5th byte
Data 4	Data 3	Data 2	Data 1
Bit 31 ... bit 16		Bit 15 ... bit 8	Bit 7 ... bit 0
Index		**Subindex**	**Length**

[12-16] Data frame assignment

IEC 61131 process data words are mapped. Only whole bytes can be mapped (1-byte/mapping entry).

Related topics:

▶ TPDO1 | "LP_CanOut1" port block (📖 829)

I-1A01

Index: **I-1A01**	Name: **TPDO2 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1A01 serves to send parameter data as TPDO2.

- This object relates to code C00868/5...8.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for TPDO2 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

- For assignment of the data telegram see object I-1A00.

Related topics:

▶ TPDO2 | "LP_CanOut2" port block (📖 830)

I-1A02

Index: **I-1A02**	Name: **TPDO3 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1A02 serves to send parameter data as TPDO3.

- This object relates to code C00868/9...12.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for TPDO3 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

- For assignment of the data telegram see object I-1A00.

Related topics:

▸ TPDO3 | "LP_CanOut3" port block (📖 831)

I-1A03

Index: **I-1A03**	Name: **TPDO4 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 4: Application object 1 ... 4	0	0		4294967295	rw	U32

The object I-1A03 serves to send parameter data as TPDO4.

- This object relates to code C00868/13...16.

Subindex	Meaning
0	Number of mapped objects
1 ... 4	Mapping entries 1 ... 4 for TPDO4 • The 4th mapping entry is used for the statistic mapping. For this, there is no value available.

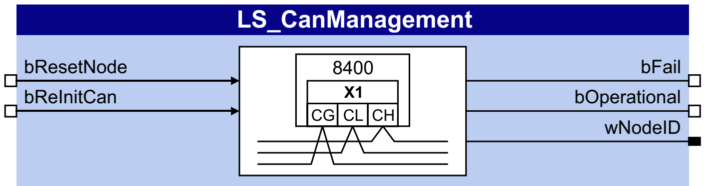
- For assignment of the data telegram see object I-1A00.

Related topics:

▸ TPDO4 | "LP_CanOut4" port block (📖 832)

12.12 Internal interfaces | System block "LS_CANManagement"

The **LS_CANManagement** system block serves to control internal functions of the CAN driver (reset node and re-initialisation) and to display the "Operational" status as well as the node address (analogous to the 9300 ServoPLC and ECS devices).



inputs

Designator (Data type)	Information/possible settings	
bResetNode (BOOL)	Reset node	
	TRUE	Carry out reset node • If the inverter is configured as CAN master in C00352, the NMT command "Start Remote Node" is sent to all nodes at the bus (broadcast telegram). ▸ Network management telegram (NMT)
bReInitCAN (BOOL)	Reinitialisation	
	TRUE	Reinitialise "CAN on board" interface.

outputs

Designator (Data type)	Value/meaning	
bFail (BOOL)	Error	
	TRUE	An event according to the error configuration in C00341 has occurred
bOperational (BOOL)	"Operational" status signal	
	TRUE	The system bus is in the "Operational" status
wNodeID (WORD)	Output of the node address	

Note!

If a "Bus off" error is detected, the "CAN on board" interface will automatically be reinitialised after 1 second.

Hence, 1 second after the "Bus off" has occurred, the inverter will automatically be active again on the system bus ("Auto bus off recovery").

13 Fieldbus interface (MCI)

The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system.

When using a communication module, the major advantage for the user is the possibility of parameterising, controlling, and diagnosing the drive system via the available fieldbus.

The following fieldbuses are supported by the 8400 TopLine:

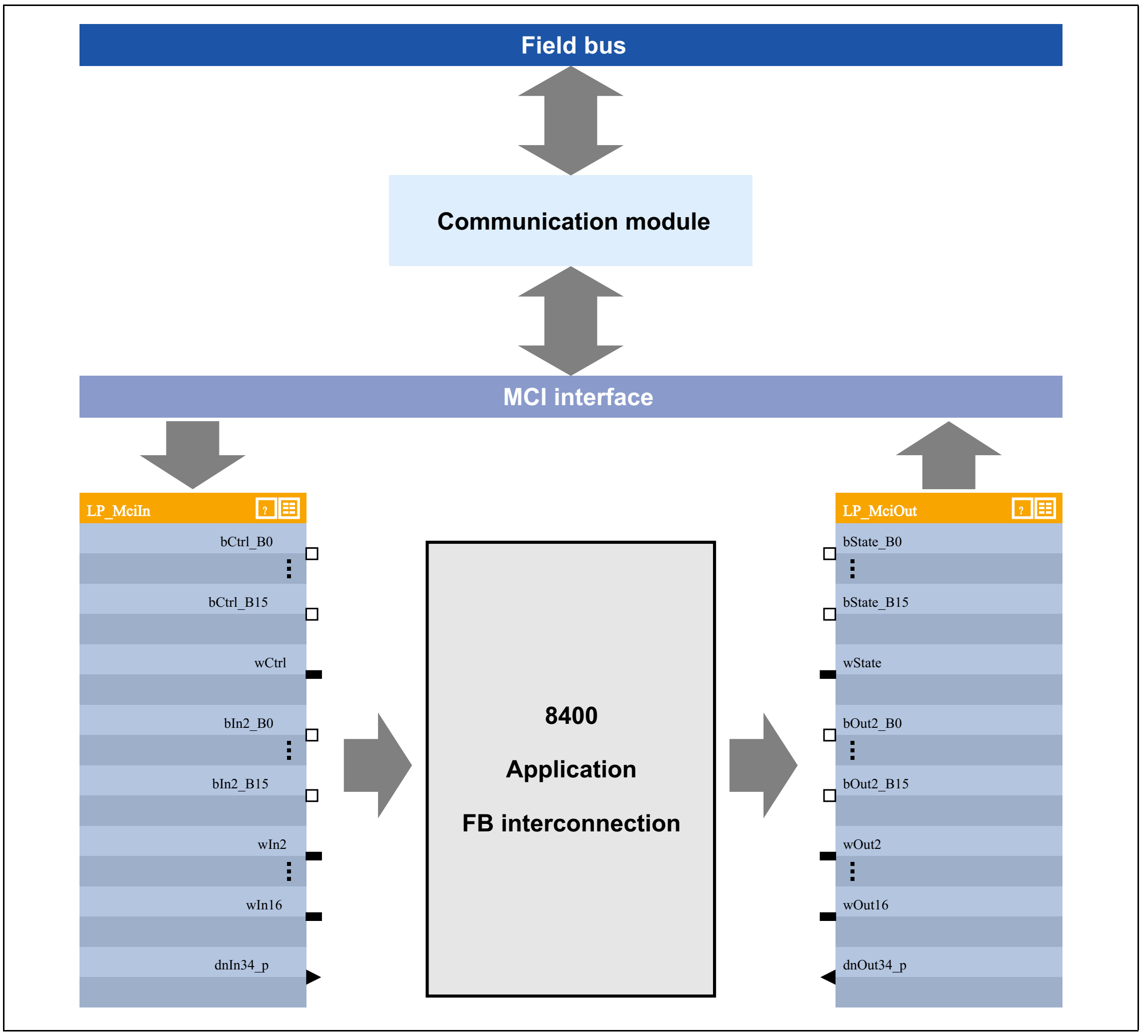
Fieldbus	Communication module (type designation)
CANopen	System bus "CAN on board"(permanently integrated in the standard device)
EtherCAT	EtherCAT® (E84AYCET)
ETHERNET POWERLINK	Ethernet POWERLINK (E84AYCEC)
EtherNet/IP	EtherNet/IP™ (E84AYCEO)
INTERBUS	INTERBUS (E84AYCIB)
PROFIBUS	PROFIBUS® (E84AYCPM)
PROFINET	PROFINET® (E84AYCER)

Detailed information is provided in the communication manual (KHB) for the respective fieldbus and in the »Engineer« online help.

13.1 Process data transfer

The process data serve to control the inverter. Thus the transfer of the process data is time-critical.

- The process cycle is 1 ms, irrespective of the respectively plugged-in bus system and the type of inverter.
- The process data transfer between the host system and the inverters is cyclical.
  - This concerns the continuous exchange of current input and output data.
  - In the case of the 8400 inverter, 16 words per direction are exchanged.
- The master computer can directly access the process data.
  Access to the process data takes place via the port blocks **LP_MciIn** and **LP_MciOut** (see FB interconnection of the »Engineer«).
  These port blocks are also called process data channels.
- The process data are not saved in the inverter.



[13-1] External and internal data transfer between bus system, inverter and function block interconnection

Voltage supply

Depending on the complexity and functional range of the fieldbus, the communication modules are supplied by the standard device or an external 24 supply at the module.

The external 24 V voltage supply of the communication module is required if the supply of the standard device fails but the communication via the bus is to continue.

Parameter setting of the communication modules

All codes which must be parameterised for establishing the fieldbus communication are saved in the memory module of the inverter.

The archived data can be addressed by all bus systems supported by the inverter.

Hotplug

The communication module (MCI module) can be plugged in/out while the inverter is switched on. When the module is plugged in, it is automatically detected and checked for plausibility regarding the function and version.

Fieldbus-specific device profiles and PDO mapping

When specific bus systems are used, the inverter is to behave according to a defined, manufacturer-spanning standard. The following definitions have been made for this:

- Definitions of the device state machine (e.g. DSP402, DriveCOM, ProfiDrive etc.)
- Definition of the bit assignment of control and status words
- Definition of signal scaling (on a limited scale)
- Definition of parameter scaling (on a limited scale)
- Definition of the process data mapping

These device profiles are not mapped in the communication module since some definitions have a strong effect on the device-internal behaviour and the device profiles are not uniform regarding this matter.

- The task of the communication modules is
  - to address parameters (SDOs),
  - to transfer PDOs and
  - the signal mapping of the PDOs.
- The process data objects (e.g. the meaning of the control word bits or the speed setpoint stipulated) are interpreted in the inverter.

13.2 Control mode "MCI"

"40: MCI" can be selected as a control mode in C00007 in order to quickly and easily set-up inverter control by means of MCI-PDOs via the fieldbus interface.

Given that the technology applications are fundamentally different and have different requirements regarding the signals sent to them, predefined assignment of the MCI-PDOs depends on the technology application selected in C00005:

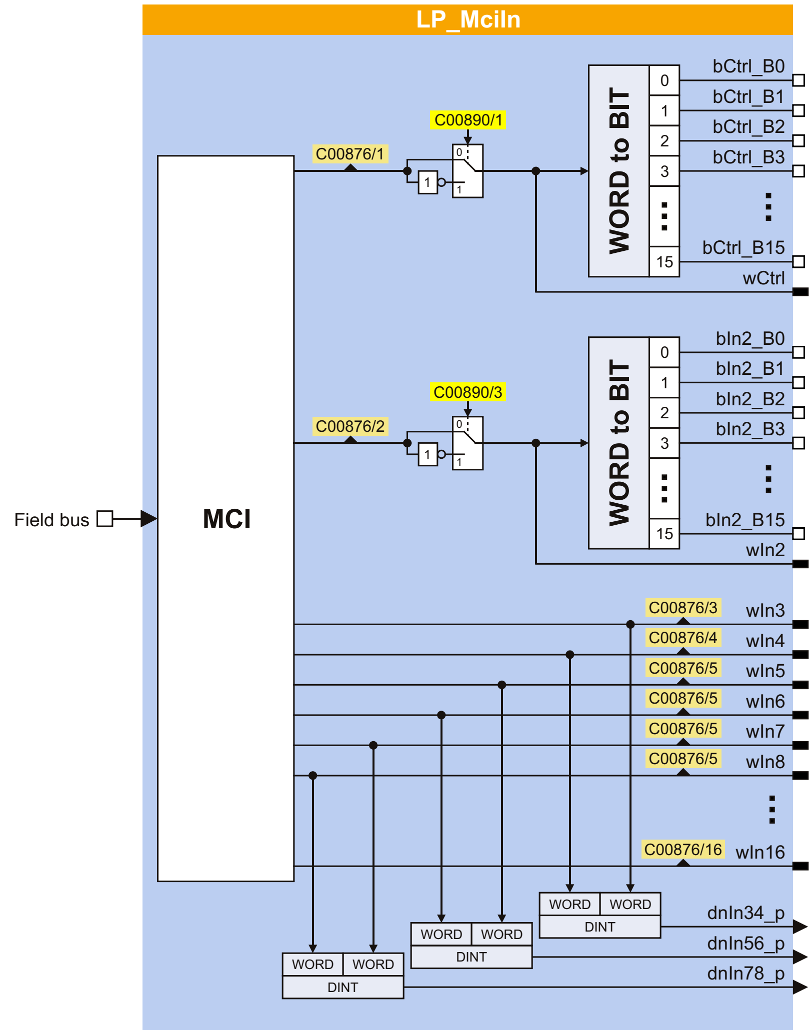
- **TA "Actuating drive speed":**
  Process data assignment for fieldbus communication (📖 474)
- **TA "Table positioning":**
  Process data assignment for fieldbus communication (📖 538)
- **TA "Abschaltpositionierung":**
  Process data assignment for fieldbus communication (📖 564)

Tip!

The predefined assignment of the MCI-PDOs can be parameterised by means of PDO mapping and can be freely configured on the I/O level in the function block editor (FB editor).

13.2.1 Port block "LP_MciIn"

The **LP_MciIn** port block maps the received MCI-PDOs in the FB Editor.



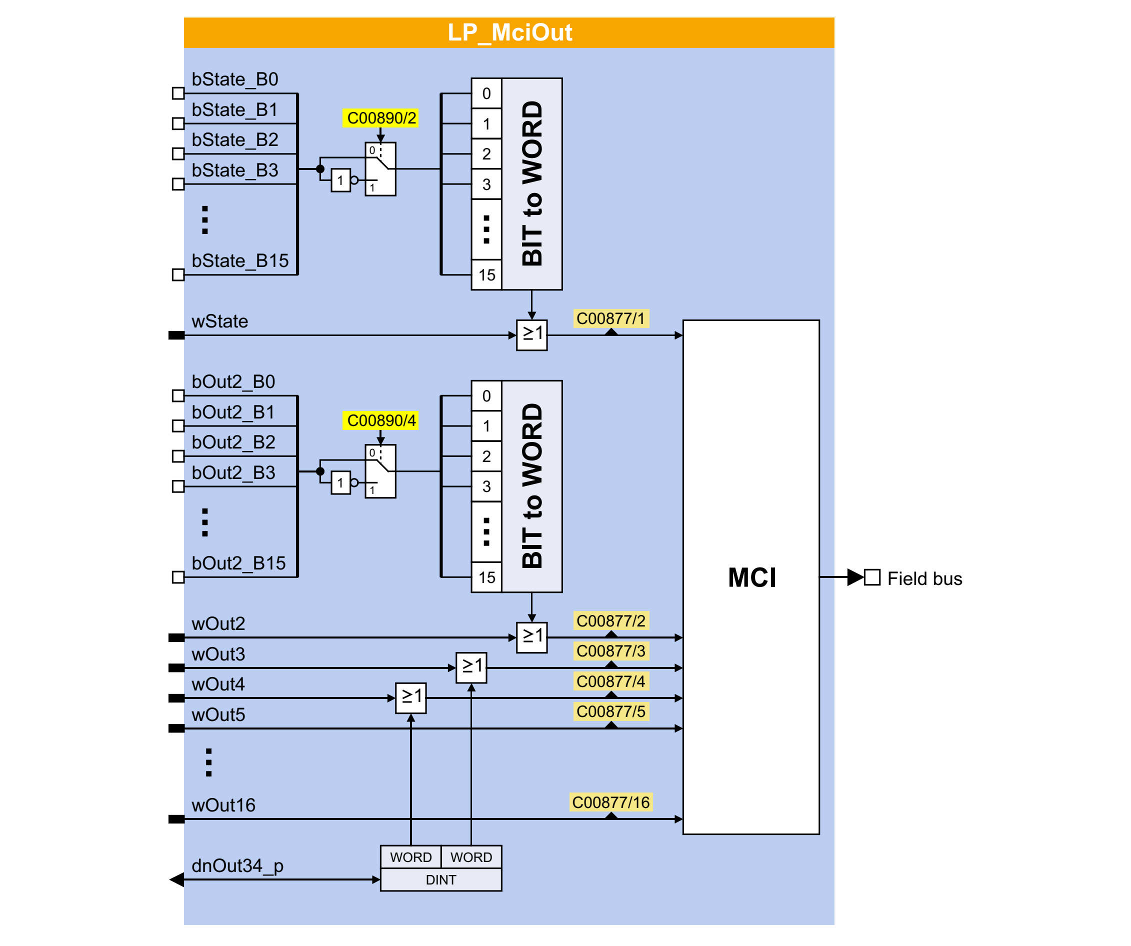
Outputs *dnIn56_p* and *dnIn78_p* are available from version 02.00.00!

Short overview of the parameters for LP_MciIn:

Parameters	Info	Lenze setting
C00876/1	LP_MciIn:wCtrl	-
C00876/2...16	LP_MciIn: wIn2 ... wIn16	-
C00890/1	LP_MciIn: Inversion bCtrl_B0..15	0x0000
C00890/3	LP_MciIn: Inversion bIn2_B0..15	0x0000
Greyed out = display parameter		

13.2.2 Port block "LP_MciOut"

The **LP_MciOut** port block maps the MCI-PDOs to be transmitted in the FB Editor.



Short overview of the parameters for LP_MciOut:

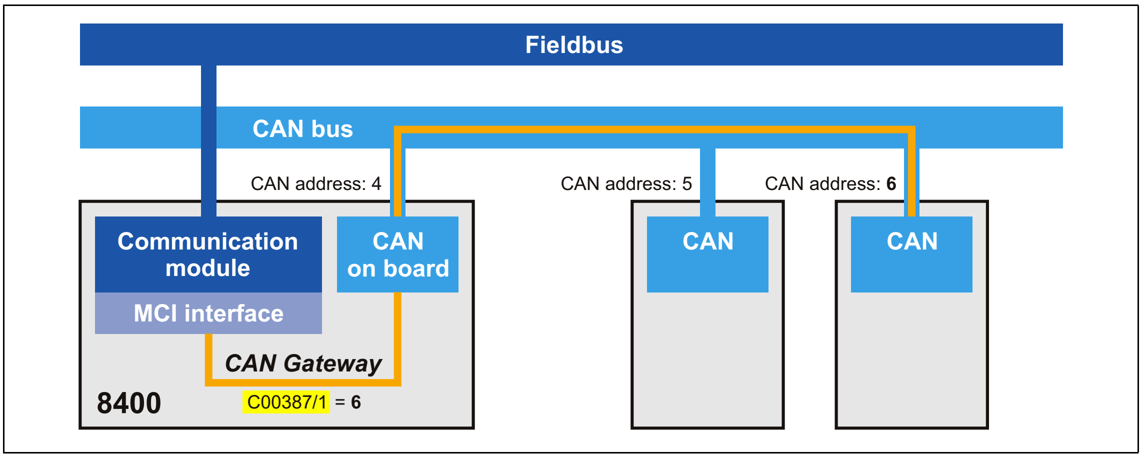
Parameters	Info	Lenze setting
C00877/1	LP_MciOut:wState	-
C00877/2...16	LP_MciOut: wOut2 ... wOut16	-
C00890/2	LP_MciOut: Inversion bState_B0..15	0x0000
C00890/4	LP_MciOut: Inversion bOut2_B0..15	0x0000
Greyed out = display parameter		

13.3 CAN gateway

This function extension is available from version 12.00.00!

The CAN gateway functionality forwards the read/write requests of the attached communication module to a connected *remote* device via the "CAN on board" system bus. It also receives the responses via this way.

- From the point of view of the attached communication module, the read/write requests are identical to accesses to its own standard device.
- Block services (reading/writing parameters of the "String" data type) can also be executes via the CAN gateway.



[13-2] Functional principle of "CAN gateway"

Activating the CAN gateway

To activate the CAN gateway, the CAN address (1 ... 127) of the connected *remote* device must be set in C00387/1.

- In the Lenze setting C00387/1 = "0", the function is switched off.
- If the CAN gateway is activated, the own inverter is the client and the *remote* device is the server.
- If the own CAN address is set, the read/write requests are forwarded to the own inverter.
- The parameters of the communication module (C13000 ... C13999) are always processed in the own inverter.
- The parameters C00387/1 (CAN gateway address) and C00350 (CAN node address) are not transferred to the *remote* device.

14 Axis bus

This function extension is available from version 02.00.00!

The 8400 TopLine inverter has an integrated axis bus which is used to couple several 8400 TopLine inverters in an axis interconnection. The main task of the axis bus is a simple cross-data exchange from axis to axis.

Axis bus operation uses two transmission media, "data transfer axis bus" and "IO axis bus" that can be used as follows:

- The data transfer axis bus is based on CAN physics and enables a high-performance data transfer of master values and control signals to other 8400 TopLine inverters. For this purpose, a synchronisation of the internal time base of the inverters via the IO axis bus is always required.
- The IO axis bus is a 1-wire bus with open-collector circuitry (5 V isolated). The IO axis bus can either be used to transfer controller errors in the interconnection ("release cord" principle) or as a pure open-collector IO function. In the first case, the inverters' internal time base can be synchronised via the IO axis bus.

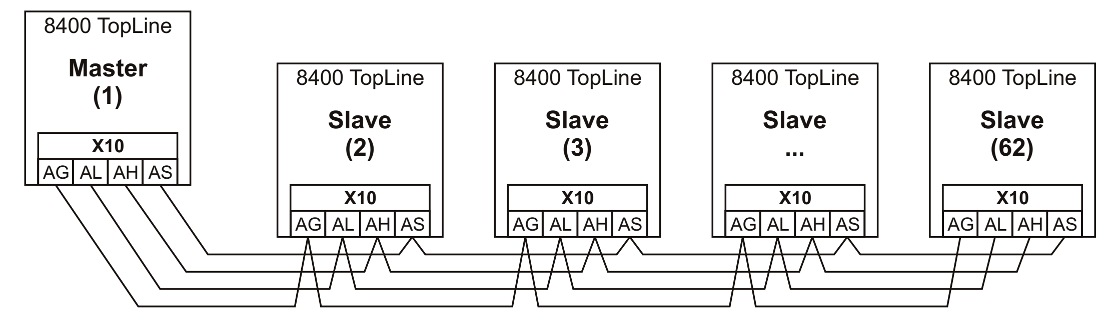
A maximum of 62 nodes can be connected to the axis bus.

Stop!

The data transfer axis bus of the 8400 TopLine inverter is designed especially for performance and simplicity. HMIs and other peripherals as well as the »Engineer« are not supported at the axis bus.

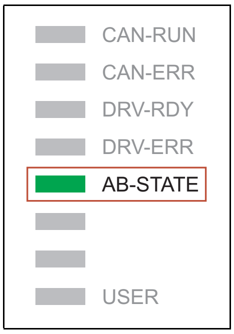
The IO axis bus of the 8400 TopLine inverter is not compatible to the state bus of the 9300/9400 device series due to different voltage levels!

Axis bus topology



X10/AL, AH	Data transfer axis bus
X10/AS	IO axis bus
X10/AG	GND, reference potential of axis bus

14.1 LED status displays of the axis bus



Information on the axis bus status can quickly be obtained via the "AB-STATE" LED display on the front of the inverter.

The meaning can be seen from the table below.

LED display	Meaning
OFF	The inverter is disconnected from the axis bus as the node address "0" is set in C02430/1.
 (blinking - 10 Hz)	The axis bus is in the "boot up" phase.
 (blinking - 2 Hz)	The axis bus is in the "Stopped" state
 (blinking - double impulse)	The axis bus is in the "Warning" state. • The error counter for incorrectly sent or received telegrams has a value > 96. • The data transfer continues to run and the error counter is tried to be reduced to 0 by error-free transfers. ▸ Axis bus data error monitoring
 (permanently on)	The axis bus is in the "PDP active" state (normal operation).

14.2 Data transfer axis bus

The data transfer axis bus is based on isolated CAN physics.

- Max. 62 nodes are supported at the bus (1 master and max. 61 slaves).
- Transfer clock = 1 ms
- Baud rate
  - Starting from version 12.00.00, the default setting of the baud rate is 800 kbps.
  - In version 2.00.00 the baud rate is 500 kbps.

Note!

The baud rate in an axis interconnection for example involving devices with version 02.00.00 must be consistently set to 500 kbps. The setting is also carried out with C2444/1, bit 15 = TRUE for devices with a higher version!

14.2.1 Topologies

There are generally two different topologies for an interconnected operation:

A. All nodes of the network are connected to the higher-level master control via MCI module and the respective fieldbus.
   - The axes can exchange data via the axis bus.
   - Moreover, the CAN bus is available.
   - For synchronous fieldbuses as e.g. EtherCAT, synchronicity of all axes towards the EtherCAT fieldbus is achieved by selecting "MCI" as sync source for the axis bus master. The synchronisation of the slaves on the EtherCAT side has to be switched off then.

B. Only the axis bus master of the network is connected to the higher-level master control via MCI module and the respective fieldbus.
   - For cost reasons, this topology only makes sense if only few information has to be transferred from the master control to the network.

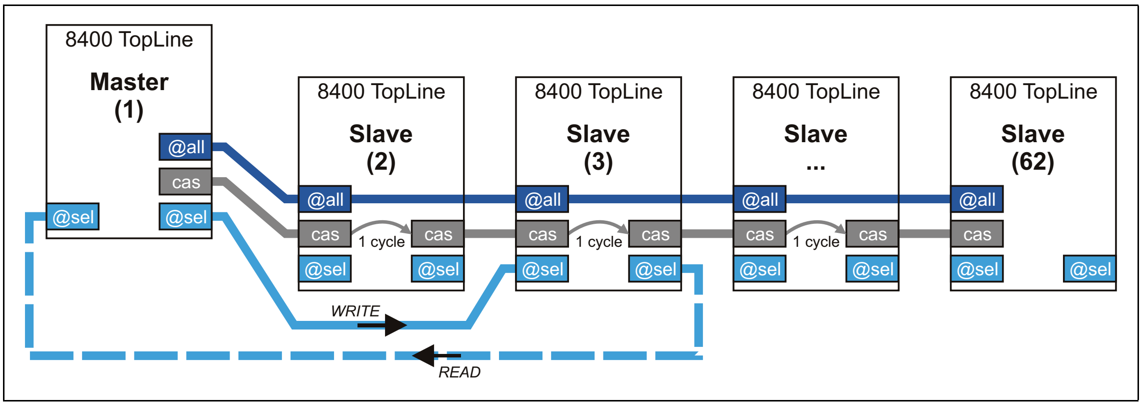
In both cases, the axes can exchange data via the axis bus and the CAN bus is available in addition.

14.2.2 Transfer mechanisms

The axis bus supports three different transfer mechanisms simultaneously according to the following methodology:

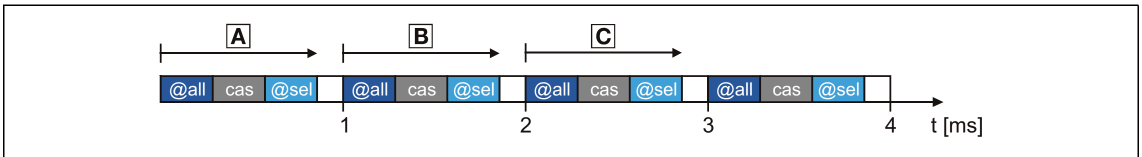
- "@all": Data transfer from master to all slaves
  - A data packet is sent from the master to all slaves in the transmission cycle.
  - Application: Data distribution according to line topology
  - Cycle time = transmission cycle = 1 ms (at baud rate = 500 kbits)
- "cas": Data transfer from node to node (cascade)
  - In the first transmission cycle, a data packet is sent from master to slave 1, in the next transmission cycle from slave 1 to slave 2, etc.
  - Application: Data distribution according to cascade topology
  - Cycle time = transmission cycle * no. of slaves+1

- "@sel": Data transfer from master to a slave
  - A data packet is sent from the master to one selected slave in the transmission cycle.
  - More options: Send to no slave, send to all slaves, and read one selected slave.
  - Application: Data distribution according to line topology for controlling one single node.
  - Cycle time = transmission cycle



[14-1] Transfer mechanisms

The axis bus data is transferred in 1-ms transmission cycle:



Case 1:	Case 2:
A Master sends setpoints	A Master requests slave to send
B Slave works with new data	B Slave sends as requested
	C Data is processed by master

[14-2] Data transfer

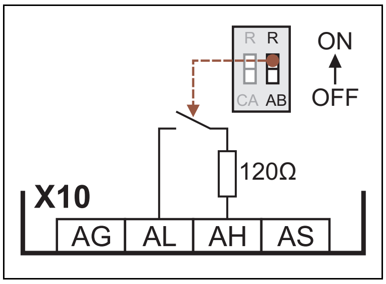
The axis bus data transfer is connected to the master/slave application via the **LS_AxisBus** systems blocks:

- For a data transfer from the master to all slaves (@all) and/or a cascaded data transfer (cas), the systems blocks LS_AxisBusIn and LS_AxisBusOut are available in the FB Editor.
- For a data transfer from the master to a certain slave (@sel), the LS_AxisBusAux system block is available.

14.2.3 Activating the bus terminating resistor

The axis bus must be terminated between axis bus low (AL) and axis bus high (AH) at the first and last physical node each by a resistor (120 Ω).

The 8400 inverter is provided with an integrated bus terminating resistor, which can be activated via the DIP switch labelled with "AB":



- OFF = bus terminating resistor is inactive
- ON = bus terminating resistor is active

[14-3] Activation of the integrated bus terminating resistor

14.2.4 Parameter setting

Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value	Unit
C01120	Sync signal source	0: Off	
C02430/1	Axis bus address • 0 ≡ Nodes disconnected from the axis bus • 1 ≡ Master • 2 ... 62 ≡ slave 1 ... slave 61	0	
C02430/2	Axis bus no. of nodes • Number of inverters connected to the axis bus.	2	
C02431/1	Axis bus decel. boot-up	3000	ms
C02431/2	Axis bus decel. BusOffRecovery	1000	ms
C02431/3	Axis bus monitoring time	1000	ms
C02440/1	Axis bus IO function	0: Off	
▶ Configuring exception handling of the received data			
C02442/1	Decoupling inputs from the axis bus • From version 12.00.00	0x0000	
C02443/1...11	Decoupling values • From version 12.00.00	0	
▶ Axis bus data error monitoring			
C00591/1	Resp. to axis bus data error • From version 12.00.00	1: Fault	
Diagnostic parameter			
C02435	Axis bus status • Display whether the node participates in the axis bus data exchange (PDO active) or is only switched on passively to the axis bus (PDO stopped).	-	
C02436	Axis bus error status	-	
C02437	Axis bus MessageError ▶ Integrated error detection	-	
C02438/1	Axis bus Tx_Error • From version 12.00.00 • Error counter for faulty sent telegrams.	-	
C02438/2	Axis bus Rx_Error • From version 12.00.00 • Error counter for faulty received telegrams.	-	
C02438/3	Axis bus transmitting meter • From version 12.00.00 • Display of the entire number of sent telegrams	-	
C02438/4	Axis bus receiving meter • From version 12.00.00 • Display of the entire number of received telegrams.	-	
Greyed out = display parameter			

14.2.4.1 Required settings at the master

For a "safe" data transfer, one inverter must be the master in the network. The master controls and synchronises the network.

Make the following two settings for the inverter which is to be the master in the network:

- Axis bus address (C02430/1) = "1"
  - With this setting, the inverter automatically takes over the control in the network.
  - Only one master is allowed in the network.
- Axis bus IO function (C02440/1) = "1: Master"
  - With this setting, the inverter outputs a synchronisation cycle to the I/O axis bus to which the slaves can orient themselves.

Optional: Synchronisation of the internal time base of the master

For the master, a sync signal source (e.g. CAN or MCI) can be selected in C01120.

- Then, the entire axis bus system behaves synchronously with this sync signal.
- The sync signal source "2: AxisBusIO" technically does not make any sense for the axis bus master and is thus ignored (same effect as with setting "0: Off").

Related topics:

▶ "Master/slave" function (📖 907)
▶ Synchronisation of the internal time base (📖 912)

14.2.4.2 Required settings for the slaves

Make the following settings for every slave inverter in the network:

- Axis bus address (C02430/1) = "2" ... "62"
  - Make sure that all inverters connected to the axis bus have different axis bus addresses.
- Sync signal source (C01120) = "2: AxisBusIO"
  - With this setting, the synchronisation cycle output by the master is used as synchronisation source.
  - Basically, only one source is allowed to synchronise the internal time base.
- Axis bus IO function (C02440/1) = "2: Slave"
  - This setting serves to define the slave.

Note!

The selection of the "2: AxisBusIO" sync signal source serves to permanently set the parameters given in the following table for synchronising the internal time base to reasonable values in order to ensure a technically perfect operation of the axis bus!

Parameters	Info	Setting for "AxisBusIO" signal source	
		Value	Unit
C01121	Sync cycle time setpoint	2000	µs
C01122	Sync phase position	200	µs
C01123	Sync window	40	µs

Related topics:

▶ "Master/slave" function (📖 907)

14.2.4.3 Boot-up phase

After the devices (master and slave) are switched on and initialised, first a boot-up phase is executed. In this phase, the master already communicates on the axis bus with the slaves which do not participate yet in the data exchange.

Only when all nodes have been synchronised at the axis bus (synchronisation of the slaves with the axis bus IO is compulsory) and the deceleration time set in C02431/1 has been expired, all nodes change to the "PDO active" bus state.

14.2.4.4 Disconnect nodes from the axis bus

The setting of the axis bus address (C02430/1) to "0" serves to switch off the corresponding node dynamically from the axis bus.

- The node does not participate in the data exchange anymore.
  - The axis bus LED "AB-STATE" of the node is switched off.
  - C02435 displays the axis bus status "4: Boot-up" for the node.
- The synchronisation and the detection and report of errors via the axis bus IO cable continues to take place.

14.2.4.5 Configuring exception handling of the received data

Certain situations require the axis to be decoupled from the received data of the axis bus, e.g. if a data error has occurred. Then it is not possible anymore, for instance, to position the slave axis to a certain target position.

Exception handling for the received data in the event of an error can be set via decoupling configuration and decoupling values in order to e.g. move the slaves to a parking position.

- Bit-coded selection is carried out in C02442 1 defining the events that will trigger decoupling of the received data.

Bit	Event
Bit 0 ☐	BusOff_MsgErr
Bit 1 ☐	Warning
Bit 2 ☐	NodeStopped • For the master, the *bStop* input at the SB LS_AxisBusAux has been set to TRUE.
Bit 3 ... 13 ☐	Reserved
Bit 14 ☐	Trouble
Bit 15 ☐	Fault

- Finally, the following parameters define the value that the received data are to have when they are decoupled:

Parameters	Info	Lenze setting Value	Unit
Decoupling values for received data at the SB LS_AxisBusIn:			
C02443/1	LS_AxisBusIn: wLine1 DiscVal	0	
C02443/2	LS_AxisBusIn: wLine2 DiscVal	0	
C02443/3	LS_AxisBusIn: wLine3 DiscVal	0	
C02443/4	LS_AxisBusIn: wCas1 DiscVal	0	
C02443/5	LS_AxisBusIn: wCas2 DiscVal	0	
C02443/6	LS_AxisBusIn: wCas3 DiscVal	0	
C02443/7	LS_AxisBusIn: wCas4 DiscVal	0	
Decoupling value for received data at the SB LS_AxisBusAux:			
C02443/8	LS_AxisBusAux: wAuxIn1 DiscVal	0	
C02443/9	LS_AxisBusAux: wAuxIn2 DiscVal	0	
C02443/10	LS_AxisBusAux: wAuxIn3 DiscVal	0	
C02443/11	LS_AxisBusAux: wAuxIn4 DiscVal	0	

14.2.4.6 Integrated error detection

Just like the system bus ("CAN on board"), the axis bus is also provided with an integrated error detection. If a node detects an error, it rejects the telegram bits received so far and transmits an error flag. The error flag consists of 6 consecutive bits with the same logic value.

The following errors are detected and displayed in C02437:

Error	Description
Bit error	The sending node follows the transmission on the bus and interrupts the transmission if it receives a different logic value than the value transmitted. With the next bit, the sending node starts the transmission of an error flag. In the arbitration phase, the transmitter only detects a bit error if a dominantly sent bit is received as recessive bit. In the ACK slot as well, the dominant overwriting of a recessive bit is not indicated as a bit error.
Stuff-bit error	If more than 5 consecutive bits have the same logic value before the ACK delimiter in the telegram, the previously transmitted telegram will be rejected and an error flag will be sent with the next bit.
CRC error	If the received CRC checksum does not correspond to the checksum calculated in the bus controller, the bus controller will send an error flag after the ACK delimiter and the previously transmitted telegram will be annulled.
Acknowledgement error	If the sent ACK slot recessively sent by the transmitting node is not dominantly overwritten by a receiver, the transmitting node will cancel the transmission. The transmitting node will annul the transmitted telegram and will send an error flags with the next bit.
Format error	If a dominant bit is detected in the CRC delimiter, in the ACK delimiter or in the first 6 bits of the EOF field, the received telegram will be rejected and an error flag will be sent with the next bit.

Tip!

The errors mentioned before indicate that a physical error has occurred in the bus system.

Possible causes are:

- Several nodes with identical node address
- Too high cable length
- Too many or no terminating resistors
- Too high bus load/too many data telegrams
  (e.g. since a node permanently transmits event-controlled due to data changes of an analog signal/actual value.)
- EMC interferences on the axis bus
  (e.g. since the axis bus cable next to the motor cable is unshielded)

14.2.4.7 Axis bus data error monitoring

Operating mode

There are two error counters, one for incorrectly sent telegrams (C02438/1) and one for incorrectly received telegrams (C02438/2).

- In an error-free normal operation, both error counters are = 0.
- Depending on the error type, the counters are increased by 8 in case of an error detection, and reduced by 1 in case no error has occurred.
  - If one of the two error counters is > 96, a warning is signalled via the LED AB-STATE (double pulse).
  - If an error counter increases or remains unchanged, a watchdog counter is increased in a 1-ms cycle.
  - If an error counter is reduced, the watchdog counter remains unchanged.
  - If the error counters are = 0, the watchdog counter is reset to 0.
- If the watchdog counter reaches the monitoring time set in C02431/3:
  - The error response set in C00591/1 is activated (Lenze setting: "Fault").
  - The error message "Ab01: AchsbusTimeOut" is entered into the logbook.

Parameterisation notes for "electrical shaft" application

When an "electrical shaft" is operated via an axis bus or generally via a bus system, data transfer errors may occur due to e.g. EMC interferences. In order to prevent a mechanical offset of the drives in case of a short-time interference only, the L_Interpolator_1 function block should be used in the slave drives. This function block has an internal correction mechanism which provides for an automatic offset correction in case of a missing data telegram. The error monitoring can then be set to an insensitive value (C02431/3 = approx. 10 ... 30 ms) which results in a very robust transfer system.

What happens in case of a cable break, loose contact or short circuit of the axis bus cable?

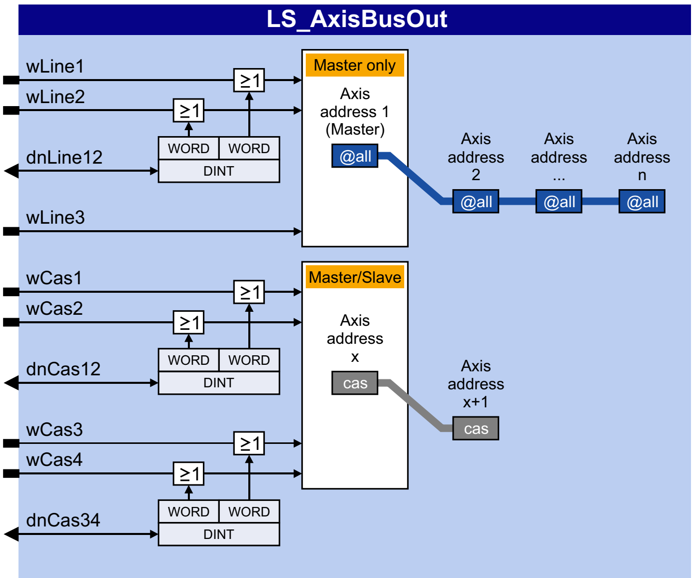
Such severe errors mostly cause a bus-off error in one or even several devices of the system.

- In this case as well, the data error monitoring is triggered.
- The "BusOff" state will be automatically reset after the time set in C02431/2 has expired if the error cause has been removed. Then, the axis bus changes to normal operation again.

14.2.5 Internal interfaces | System block "LS_AxisBusOut"

This system block is used to send line data from the master to all slaves (@all) and/or send the cascaded data (cas).

- The line data is sent as process data object (PDO) per transmission cycle to all slaves and forwarded to the application. Moreover, this PDO serves to transmit an internal control word (thus, only three words are available for data transfer).
- The cascaded data is also sent as PDO per transmission cycle from one slave to the next. The control which slave has to send when and at what time a slave in the chain has to accept the data, is executed by the internal control word in the PDO of the line data.

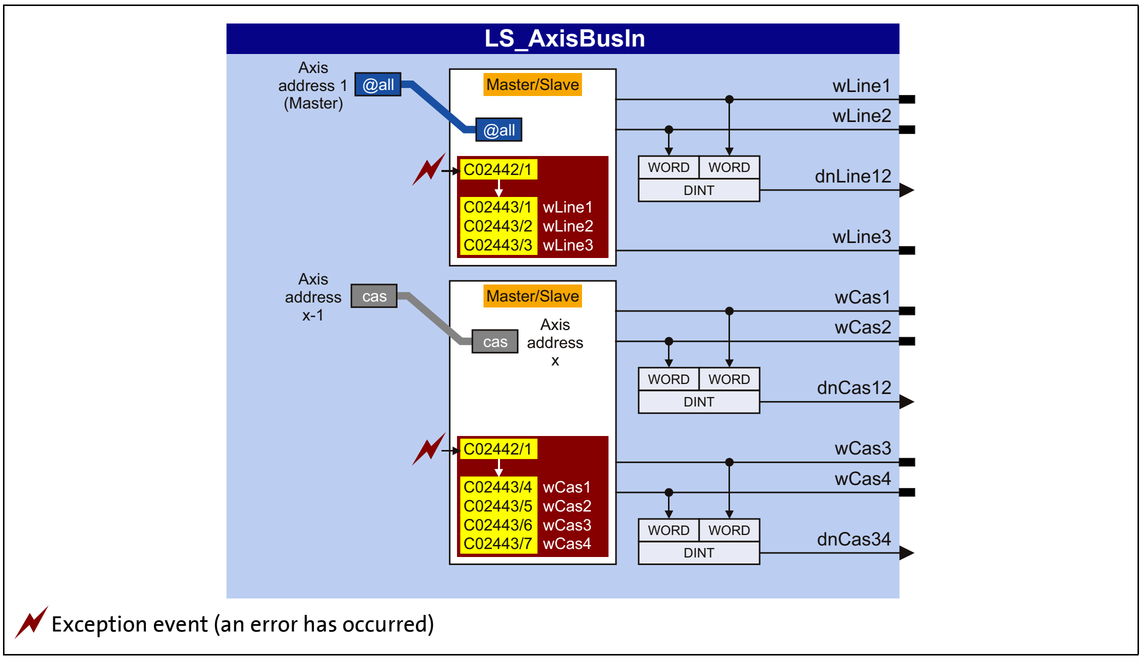


inputs

Input Data type	Information/possible settings
wLine1/wLine2 WORD	Line data - word 1 and word 2 • Selection as separate words or as double word. • The double word is OR'd with word 1 and word 2.
dnLine12 DINT	
wLine3 WORD	Line data - word 3
wCas1/wCas2 WORD	Cascaded data - word 1 and word 2 • Selection as separate words or as double word. • The double word is OR'd with word 1 and word 2.
dnCas12 DINT	
wCas3/wCas4 WORD	Cascaded data - word 3 and word 4 • Selection as separate words or as double word. • The double word is OR'd with word 3 and word 4.
dnCas34 DINT	

14.2.6 Internal interfaces | System block "LS_AxisBusIn"

This system block is used to receive the (line) data from the master (@all) and/or receive the cascaded data (cas).



outputs

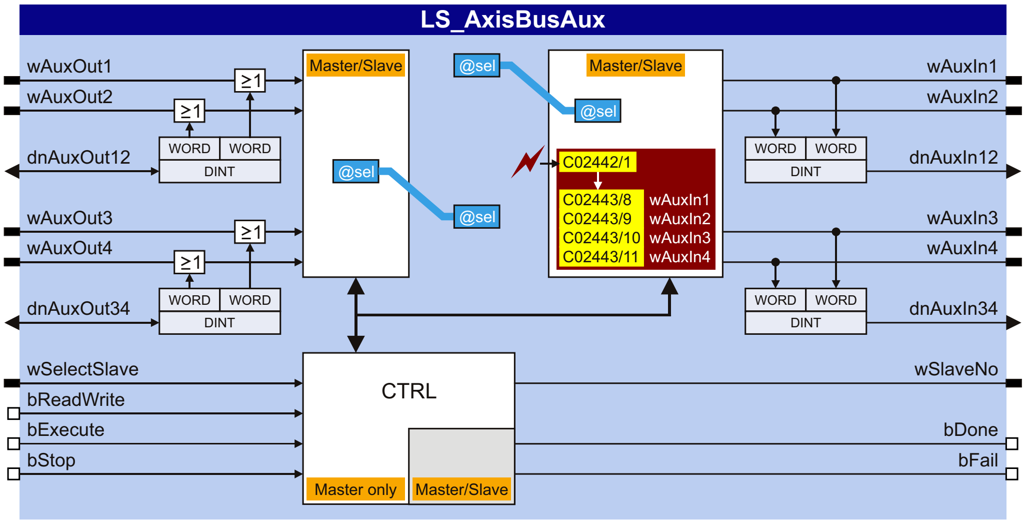
Output Data type	Value/meaning
wLine1/wLine2 WORD	Line data - word 1 and word 2
dnLine12 DINT	Line data - word 1 and word 2 as double word
wLine3 WORD	Line data - word 3
wCas1/wCas2 WORD	Cascaded data - word 1 and word 2
dnCas12 DINT	Cascaded data - word 1 and word 2 as double word
wCas3/wCas4 WORD	Cascaded data - word 3 and word 4
dnCas34 DINT	Cascaded data - word 3 and word 4 as double word

Related topics:

▶ Configuring exception handling of the received data (📖 899)

14.2.7 Internal interfaces | System block "LS_AxisBusAux"

This system block is used for data transfer between the master and one selected slave (@sel). Data can either be received from the slave ("read") or sent to the slave ("write"). By selecting the axis bus address "63", data can also be sent to all slaves at the axis bus.



Exception event (an error has occurred)

inputs

<table>
<tr><th>Input
Data type</th><th colspan="2">Information/possible settings</th></tr>
<tr><td>wAuxOut1/wAuxOut2
WORD</td><td colspan="2" rowspan="2">Data to be sent - word 1 and word 2
• Selection as separate words or as double word.
• The double word is OR'd with word 1 and word 2.</td></tr>
<tr><td>dnAuxOut12
DINT</td></tr>
<tr><td>wAuxOut3/wAuxOut4
WORD</td><td colspan="2" rowspan="2">Data to be sent - word 3 and word 4
• Selection as separate words or as double word.
• The double word is OR'd with word 3 and word 4.</td></tr>
<tr><td>dnAuxOut34
DINT</td></tr>
<tr><td colspan="3">The following inputs are only relevant for the master except for bExecute.
Via these inputs, the master controls the data acceptance of the "@sel" data in the slaves.</td></tr>
<tr><td>wSelectSlave
WORD</td><td colspan="2">Axis bus address of the slave data is to be received of or data is to be sent to.
• "0" ≡ data is sent to no slave.
• "63" ≡ data is sent to all slaves.</td></tr>
<tr><td rowspan="3">bReadWrite
BOOL</td><td colspan="2">Read/write access</td></tr>
<tr><td>FALSE</td><td>Data is sent to selected slave.</td></tr>
<tr><td>TRUE</td><td>Data is received by selected slave.</td></tr>
<tr><td rowspan="3">bExecute
BOOL</td><td colspan="2">Data acceptance</td></tr>
<tr><td rowspan="2">TRUE</td><td>For the master: The pending data is accepted and the read/write access is executed.</td></tr>
<tr><td>For the slave: The @sel data received by the master are provided via the outputs wAuxIn1...4 of this function block.</td></tr>
</table>

Input Data type	Information/possible settings	
bStop BOOL	Stop data transfer	
	TRUE	The axis bus changes to the "Stopped" state. • The slaves neither accept any data nor send any data anymore. • The master continues to send the line data. • The bit 2 ("NodeStopped") in C02442/1 serves to set whether the last received data remain active or the decoupling values are set. ▶ Configuring exception handling of the received data

outputs

Output Data type	Value/meaning	
wAuxIn1/wAuxIn2 WORD	Received data - word 1 and word 2	
dnAuxIn12 DINT	Received data - word 1 and word 2 as double word	
wAuxIn3/wAuxIn4 WORD	Received data - word 3 and word 4	
dnAuxIn34 DINT	Received data - word 3 and word 4 as double word	
wSlaveNo WORD	Display which slave has sent its data. • Output only relevant for master.	
bDone BOOL	Status signal "Data transfer completed"	
	FALSE	The master has sent a new read request to a slave.
	TRUE	Data has been received correctly.
bFail BOOL	"Fault" status signal	
	TRUE	Data has not been received correctly (e.g. bus error). ▶ Axis bus data error monitoring

Related topics:

▶ Configuring exception handling of the received data (📖 899)

14.3 IO axis bus

Stop!

The IO axis bus of the 8400 TopLine inverter is not compatible to the state bus of the 9300/9400 device series due to different voltage levels!

In order to use the IO axis bus, first decide whether the IO axis bus is to be used for transferring controller errors in the network ("release cord" principle) or as a pure open-collector IO function. The required function of the IO axis bus must then be set for all nodes in C02440/1.

- "Master/slave" function ("release cord" principle)
  - In this setting, the IO axis bus only knows the two statuses "OK" and "Error".
  - Each node connected to the IO axis bus can set the IO axis bus into the "Error" status.
  - In the "Error" status, all nodes start their adjustable response, e.g. a synchronised braking of the drive system or the master only brakes the network to standstill.
  - The "Error" status can only be reset by the node defined as "master".
  - The internal time base of the inverters can be synchronised via the IO axis bus.
- "IO" function
  - With this setting, the IO axis bus is used as a pure IO transmission medium.
  - Each node connected to the IO axis bus can transmit a TRUE signal to all nodes.

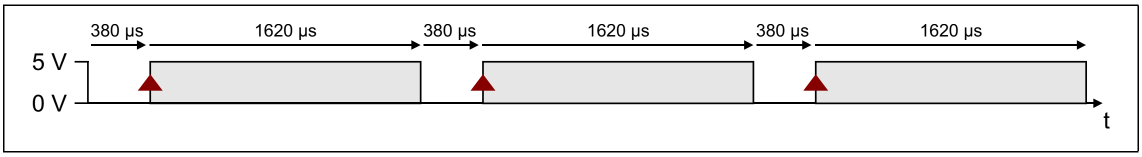
Short overview of the relevant parameters:

Parameters	Info	Lenze setting Value | Unit
C00591/2	Resp. to axis bus IO error • From version 12.00.00	1: Fault
C01120	Sync signal source	0: Off
C02440/1	Axis bus IO function	0: Off
C02444/1	Axis bus settings (bit coded) • From version 12.00.00 • Bit 0 serves to reconfigure the LS_AxisBusIO.*bSetFail_DigOut* input from level-sensitive to edge-sensitive.	0x0000

14.3.1 "Master/slave" function

With this setting, the IO axis bus can be used as "release cord", i.e. each node can set all other nodes into an error status.

One IO axis bus node takes over the master role. The master's task is to output a synchronisation cycle to the IO axis bus to which the slaves can orient themselves:

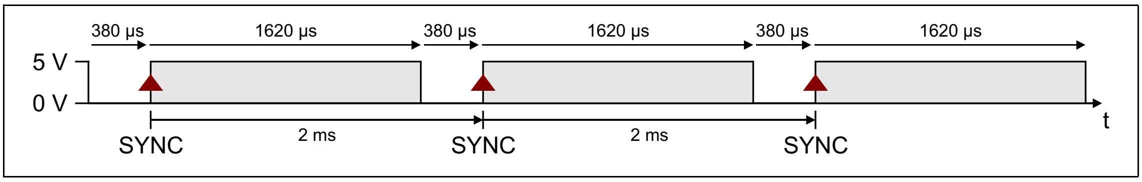


[14-4] IO axis bus: Synchronisation cycle from the master (error-free status)

The inverter is configured as master or slave in C02440/1 or by selecting "1: Master" or "2: Slave".

Synchronisation of the internal time base

The internal time base of the slaves can be synchronised via the IO axis bus. The slaves orient themselves to the synchronisation cycle output by the master:



[14-5] Synchronisation cycle

- In order that the IO axis bus is used as synchronisation source, all slaves must be set to "AxisBusIO" in C01120.
  - The internal time base is synchronised based on the HIGH edge in a 2-ms cycle (see illustration above).
  - Internal timing and trimming of the phase position are set automatically.
- Basically, only one source is allowed to synchronise the internal time base. If "AxisBusIO" is selected, this is the IO axis bus master.

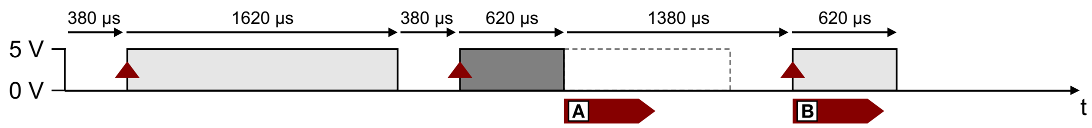
More information on this subject can be found in the following main chapter "Synchronisation of the internal time base". (📖 912)

Error-free status

In the error-free status, all nodes are error-active, i.e. each node can set the IO axis bus into the "Error" status.

Error status

By setting the LS_AxisBusIO.*bSetFail_DigOut* input to TRUE, each node can set the IO axis bus into the "Error" status. The node signalises this by reducing the HIGH pulse length to 620 µs:



A A node signalises "Error" by a reduced HIGH pulse length
B "Error" status is active for all nodes

[14-6] Synchronisation cycle (error status)

All other nodes detect the "error status" due to the changed pulse length which has the following effect:

- The LS_AxisBusIO.*bFail_DigIn* output is set to TRUE. This digital signal can be used for any application within the function block interconnection.
- From version 12.00.00:
  - The error response set in C00591/2 is activated (Lenze setting: "Fault").
  - The error message "Ab02: Axis bus IO error" is entered into the logbook.
- The nodes are now error-passive, i.e. they cannot signalise any further errors in order that the HIGH edge for synchronisation can be detected after 2 ms.

Reset "error" status

Note!

The "error" status can only be reset by the master!

A FALSE/TRUE edge at the LS_AxisBusIO.*bResetFail* input (for the master) serves to reset the "error" status again. The master signalises this by providing a LOW pulse for 3 ms.

All other nodes detect the error reset due to this signal, which has the following effect:

- The LS_AxisBusIO.*bFail_DigIn* output is reset to FALSE.
- After a delay time of 2 ms, the LS_AxisBusIO.*bResetFailIn* output is set for 3 ms to TRUE. This digital signal can be used within the function block interconnection to e.g. reset errors.

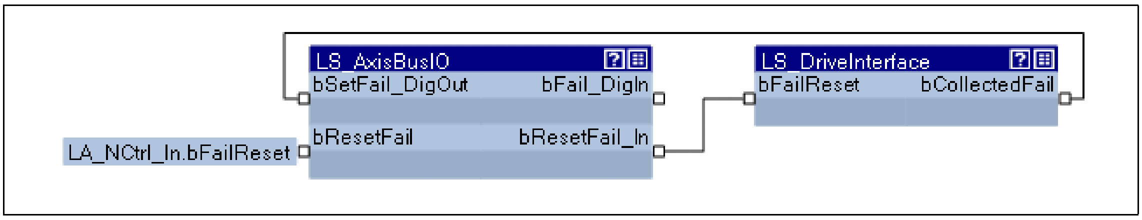
Application notes

In the event of an error in a network, a slave drive should never trigger the "Fault error response as this causes the pulse inhibit. This deactivates the coupling to the master and hence the network.

Possible solution:

1. Reconfigure the "Fault" error response to "Warning" in the slave axes.
2. Use the "master/slave" function ("release cord") to report errors and warnings to the master. The master can then initiate the respective response for the entire network.

In the following interconnection example, the slave drive uses the *bCollectedFail* group error output of the SB LS_DriveInterface for "pulling the release cord". Non-relevant inputs/outputs of the SB LS_DriveInterface are hidden for an easier presentation.



[14-7] Configuration for "master/slave" function ("release cord")

In order that this mechanism works, the following parameter setting is additionally required for the slave drive:

1. Go to C00148 and set bit 6 to "1" in order that warnings are also provided via the *bCollectedFail* group error output of the SB LS_DriveInterface.
2. Go to C02444/1 and set bit 0 to "1" in order that the *bSetFail_DigOut* input is edge-sensitive instead of level-sensitive. Otherwise, the error set input is "latched" and the slave drive does not respond anymore to an error reset by the master.

14.3.2 "IO" function

With this setting, the IO axis bus is used as a pure IO transmission medium, just like it the case for the 9300 and 9400 device series.

The "IO" function is activated in C02440/1 by selecting "3: IO".

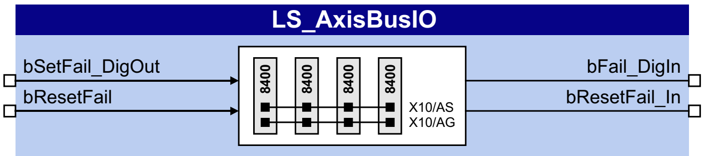
Note!

- "IO" function and Data transfer axis bus are mutually exclusive since with activated "IO" function, a synchronisation of the internal time base is not possible (which is required for a "safe" data transfer).
- The "IO" function must be set for all IO axis bus nodes.
- Triggering an error message with this function is only possible by function block interconnection.
- Resetting an error message is not possible with this function.

- By setting the LS_AxisBusIO.*bSetFail_DigOut* input to TRUE, the IO axis bus is set to the dominant level and for all nodes, the LS_AxisBusIO.*bFail_DigIn* output is set to TRUE.
- The LS_AxisBusIO.*bResetFail* input has no function.
- The LS_AxisBusIO.*bResetFail_In* output is always FALSE.

14.3.3 Internal interfaces | System block "LS_AxisBusIO"

The **LS_AxisBusIO** system block maps the IO axis bus in the FB Editor:



inputs

Designator Data type	Information/possible settings	
bSetFail_DigOut BOOL	Set IO axis bus into the "error" status / IO data exchange	
	TRUE or FALSE↗TRUE	For all IO axis bus nodes, the *bFail_DigIn* output is set to TRUE. • From version 12.00.00 onwards, bit 0 in C02444/1 can be used to reconfigure the *bSetFail_DigOut* input from level-sensitive to edge-sensitive.
bResetFail BOOL	Reset "error" status • Only possible with "Master/slave" function via the IO axis bus master.	
	FALSE↗TRUE	For all IO axis bus nodes, the *bFail_DigIn* output is reset to FALSE. After a delay time of 2 ms, the *bResetFail_In* output is set to TRUE for 3 ms for error reset.

outputs

Designator Data type	Value/meaning	
bFail_DigIn BOOL	With "master/slave" function: "Error" status With "IO" function: Data exchange	
	TRUE	An IO axis bus nodes has set the *bSetFail_Digout* input to TRUE.
bResetFail_In BOOL	"Error" status has been reset • With "IO" function, this output is always FALSE.	
	TRUE	The IO axis bus master has reset the error message. This status is only pending for 3 ms.

15 Synchronisation of the internal time base

In a drive system, synchronising the internal time bases of all inverters involved makes sense because cyclic process data should be processed synchronously in all drives.

- One of the following signal sources can be used for automatic synchronisation of the internal time base of the inverter:
  - CAN bus ("CAN on board") ➞ sync telegram
  - Axis bus ("AxisBusIO") ➞ master/slave function
  - MCI ➞ sync signal of a plugged-in communication module (EtherCAT, PROFINET or Powerlink)

Short overview of the parameters for the synchronisation of the internal time base:

Parameters	Info	Lenze setting Value	Unit
C00370/1	CAN Sync instant of transmission	-	µs
C00370/2	Sync instant of reception	-	µs
C01120	Sync signal source	Off	
C01121	Sync cycle time setpoint	1000	µs
C01122	Sync phase position	0	µs
C01123	Sync window	100	µs
C01124	Sync correction width	320	ns
Greyed out = display parameter			

Sync signal source

The synchronisation signal source can be selected in C01120. As a general rule, only one source can be used to synchronise the internal time base.

Sync cycle time setpoint

Time with which the internal phase-locking loop (PLL) expects the synchronisation signals. The time must be set in C01121 in accordance with the cycle of the synchronisation source selected in C01120.

Note!

- Only integer multiples of 1000 µs can be set in C01121.
- Intelligent communication modules usually define the cycle time setpoint derived from the bus cycle. In this case, a manual change is not possible.
- Even in case of a synchronisation via axis bus, a setting of the cycle time and the phase position is not required/possible and is executed automatically.

Example: For the CAN bus, 2 ms has been selected as interval between two synchronisation signals. If the CAN bus is to be used as synchronisation source, a cycle time setpoint of 2000 µs must be selected in C01121.

Sync window

In C01123, a time slot can be defined around the expected synchronisation time. The time slot defines the range in which the synchronisation signal is expected.

If "4:MCI" is selected in C01120, the time slot must be set to 100 µs.

Sync phase position

The phase position determines the zero-time of the internal system cycle with regard to the synchronisation signal (bus cycle). Since PDO processing is an inherent part of the system part of the application, the instant of acceptance of the PDOs is postponed as well by a changed phase position.

- If "0" is set, the internal system cycle starts at the same time as the synchronisation signal.
- If a value > 0 is set, the internal system cycle starts by the set time earlier (the phase position has a negative effect) than the synchronisation signal.
- Intelligent communication modules define the optimal time with activated synchronisation by themselves. In this case, a manual change is not possible.
- For determining C01122, the point in time where all bus nodes have valid PDOs is decisive.

Example: If the phase position is set to 550 µs, the system part of the application starts 550 µs before the arrival of the synchronisation signal.

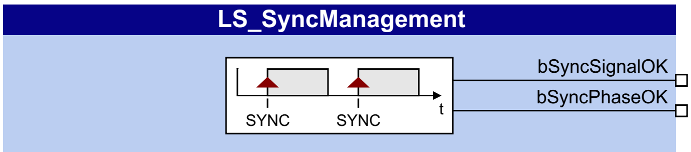
Sync correction width

If the cycle times of the synchronisation signal and the phase-locking loop (PLL) are different, the setting in C01124 defines the correction increments for the phase-locking loop.

- The recommended reset time for the CAN bus as synchronisation source in case of occurring deviations is 320 ns (Lenze setting).
- If synchronisation is not reached, select a higher correction width.
- The optimum setting depends on quartz precision and must be determined empirically if required.

15.1 Internal interfaces | System block "LS_SyncManagement"

The SB **LS_SyncManagement** provides status information for synchronising the internal time base:



outputs

Designator Data type	Value/meaning	
bSyncSignalOK BOOL	TRUE	Sync signal OK
bSyncPhaseOK BOOL	TRUE	Sync phase position OK

16 Parameter change-over

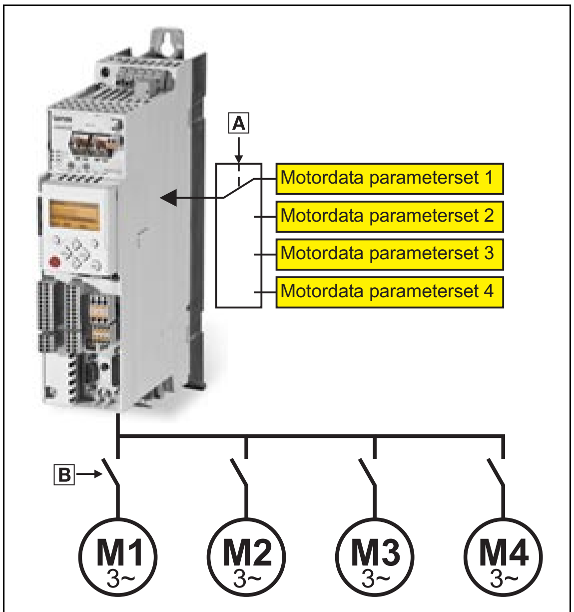
For up to 32 freely selectable parameters, this basic function provides a change-over between four sets with different parameter values.

The parameter list is created in the same way as the user menu is composed, namely by means of parameterisation. In the »Engineer«, a user-friendly parameterisation dialog with import and export functions is available for this purpose.

Motor data changeover

From version 12.00.00, an optional changeover between four parameter sets with different motor and control settings is supported in addition.

The optional motor data changeover is provided for applications/machines which have multiple axes controlled successively but which do not require the simultaneous operation of multiple motors. In this case, the very same inverter can control the motors successively. Advantages of this solution: Fewer components (inverters) and hence less energy consumption.



A Motor data changeover
B Motor changeover (e.g. via motor contactors)

Principle:

- The motor which is being controlled at the moment is connected to the inverter via motor contactors. (The contactor system can e.g. be controlled via the digital outputs of the inverter.)
- At the same time, the motor data changeover serves to activate the motor and control settings suitable for the motor in the inverter.

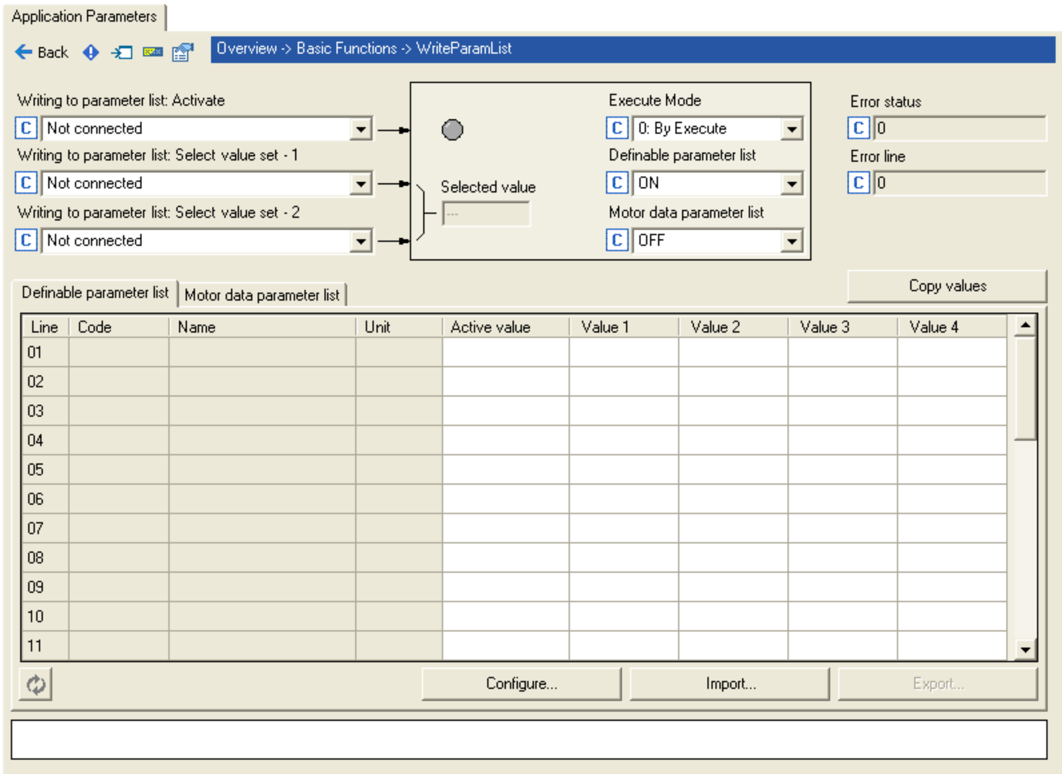
Note:
The motor data changeover is carried out within 2 ms and hence within the opening and closing times of most motor contactors.

[16-1] Principle of the selective control of multiple motors using one inverter

16.1 Configuring parameter change-over via the »Engineer« parameterisation dialog

Proceed as follows to open the dialog for parameterising the parameter change-over:

1. »Engineer« Go to the *Project view* and select the 8400 TopLine inverter.
2. Select the **Application parameters** tab from the *Workspace*.
3. Go to the *Overview* dialog level and click the "**Basic functions**" button.
4. Go to the *Overview* → *Basic functions* dialog box and click the **Parameter change-over** button.



Note!

The "parameter change-over" basic function is always processed, even if the corresponding LS_WriteParamList system block has been removed from the interconnection by the function block editor.

If you do not require this basic function anymore, delete the composed parameter list in order that no unwanted parameter write operations take place.

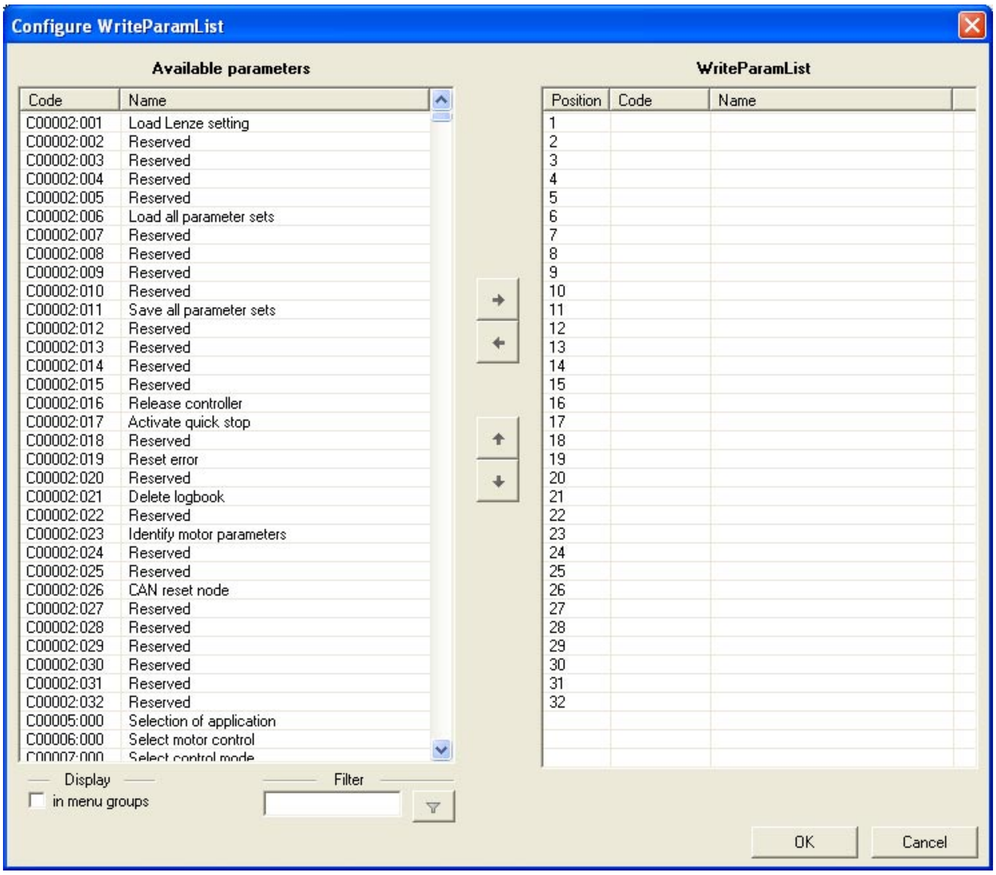
16.1.1 Configuring the parameter list(s)

Configuring the definable parameter list

In the Lenze setting, the definable parameter list does not yet contain any parameters.

How to configure the definable parameter list:

1. Click the **Adapt...** button.
   - The dialog box entitled *Configure WriteParamList* is shown:



   - On the left-hand side, all the parameters of the inverter with write and read access are shown in the list entitled **Available parameters**.
   - If the option **In menu groups** is activated, all parameters are shown assigned to their functions.
   - By clicking on the  button in the **Filter** area, you can shorten the list of available parameters. If, for example, you enter the text "ain1" and then click on the button, only those parameters whose designation contains this text are shown for selection.
2. Highlight the parameter/parameters in the **Available parameters** list that is/are to be added to the *WriteParamList*.
   - For a multi-selection you can, just like in the general Windows function, use the <**Ctrl**> and <**Shift**> key.

3. Click on the button in order to add the highlighted parameters to the *WriteParamList* on the right-hand side.
   - With the and buttons, you can alter the sequence of parameters in the *WriteParamList*.

   To remove parameters from the *WriteParamList*, proceed as follows:
   - Highlight the parameter/parameters in the **WriteParamList** that is/are to be removed from the *WriteParamList*.
   - Click on the button to remove the highlighted parameters from the *WriteParamList*.
4. Click the **OK** button to accept the configuration and close the dialog box.
   - You can call the configuration dialog again at any time in order to change or expand the *WriteParamList* retrospectively.

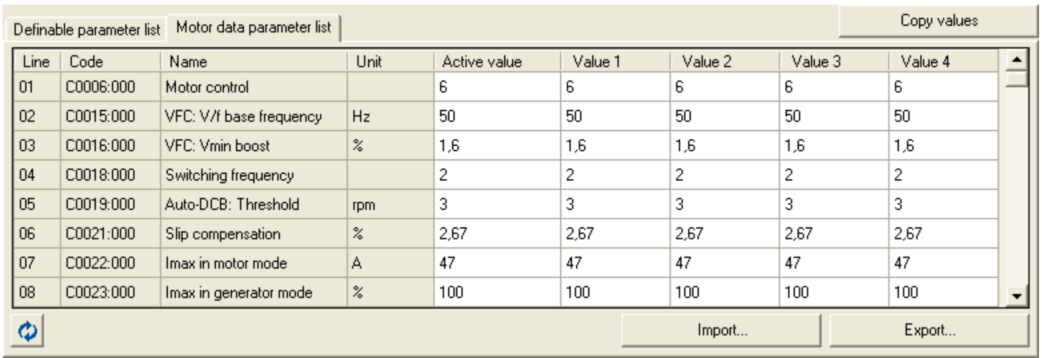
Changing the values of the definable parameter list

After the compilation of the definable parameter list, the values in the columns **1st value** ... **4th value** first correspond to the Lenze setting of the respective parameter.

- Just click one of the input fields in these columns to change the displayed value.
- If you place the cursor in an input field, the permitted value range for the corresponding parameter is shown under the table.

Changing the values of the motor data parameter list

Click the **Motor data parameter list** register to put it in the foreground:



Line	Code	Name	Unit	Active value	Value 1	Value 2	Value 3	Value 4
01	C0006:000	Motor control		6	6	6	6	6
02	C0015:000	VFC: V/f base frequency	Hz	50	50	50	50	50
03	C0016:000	VFC: Vmin boost	%	1,6	1,6	1,6	1,6	1,6
04	C0018:000	Switching frequency		2	2	2	2	2
05	C0019:000	Auto-DCB: Threshold	rpm	3	3	3	3	3
06	C0021:000	Slip compensation	%	2,67	2,67	2,67	2,67	2,67
07	C0022:000	Imax in motor mode	A	47	47	47	47	47
08	C0023:000	Imax in generator mode	%	100	100	100	100	100

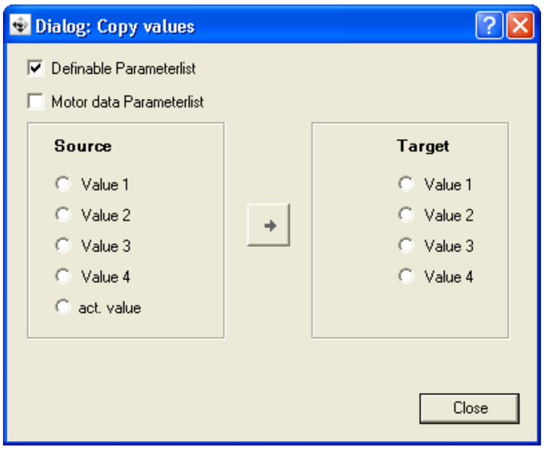
- Unlike the "definable" parameter list, the motor data parameter list has a fixed assignment to the motor and control parameters.
  - A list of the switchable motor and control parameters is provided in chapter "Configuring the motor data parameter list by means of parameterisation". (🕮 922)
- The preset values are changed in the same way as the definable parameter list.

Copying values

All the settings of a value set can be copied to another value set.

To copy values, proceed as follows:

1. Click on the **Copy values** button.
   - The *Copy values* dialog box is displayed:



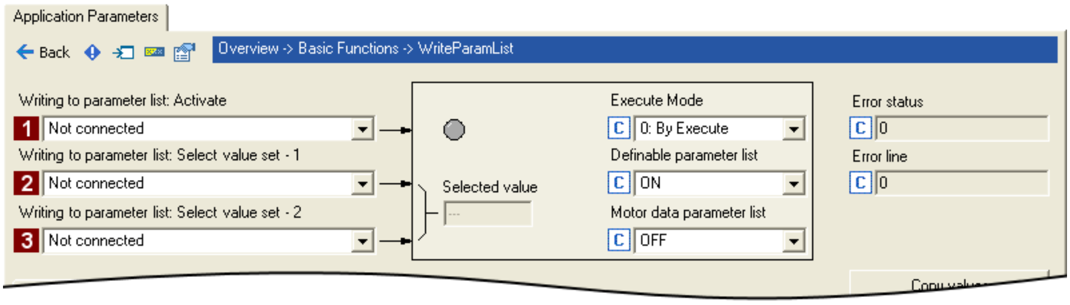
2. Define the parameter list to be copied via the two upper checkboxes.
3. Select **Source** and **Target**.
4. Click on button in order to copy the values from **Source** to **target**.

Importing/exporting the list

For cross-device reuse of the configured *WriteParamList*, you can click on the **Export** and **Import** buttons to save the parameter selection as an *.epc file and then to re-import the saved *.epc file into another 8400 inverter.

16.1.2 Configuring control inputs

The three control inputs for parameter change-over can be configured via the following parameters:

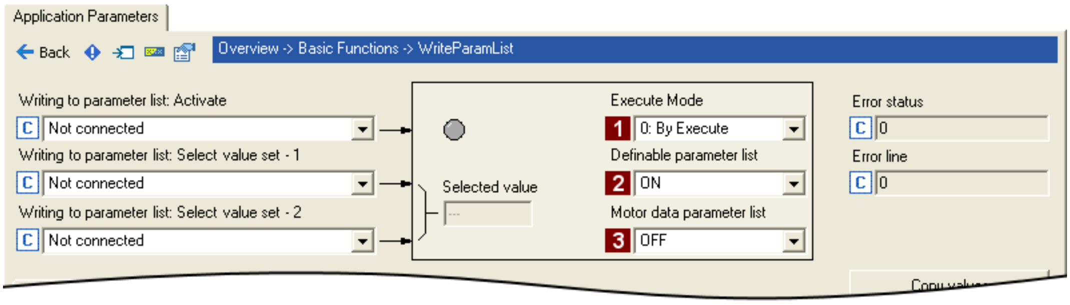


Parameters		Lenze setting	Info
1	**Activate writing** (C00621/123)	0: Not connected	Selection of the signal source that activates writing to the parameter list (for Execute Mode = "0: by Execute").
2	**Selection value set - 1** (C00621/124)	0: Not connected	Selection of the two signal sources for the binary-coded selection of the value set 1 ... 4 to be used (see following truth table).
3	**Selection value set - 2** (C00621/125)	0: Not connected	

Truth table for the selection of the value set to be used:

Selection value set - 1	Selection value set - 2	Value set used
FALSE	FALSE	Value set 1
TRUE	FALSE	Value set 2
FALSE	TRUE	Value set 3
TRUE	TRUE	Value set 4

16.1.3 Functional settings



Parameters		Lenze setting	Info
1	**Execute Mode** (C01082)	0: by Execute	For writing the parameter list, two modes are available: • 0: by Execute (Lenze setting) The writing of the parameter list is activated by a FALSE/TRUE edge at the *bExecute* control input. • 1: by Input Select The parameter list is written when the selection inputs *bSelectWriteValue_1* and *bSelectWriteValue_2* are changed and once during the initialisation of the inverter. **Note:** If the execute mode changes from "0: by Execute" to "1: by Input Select", the parameter list selected by the selection inputs is written once. In the execute mode "1: by Input Select", the parameter list then is only written again when a change occurs on the selection inputs .
2	**Definable parameter list** (C02200/1)	1: On	Switch on/off parameter changeover for the definable parameter list.
3	**Motor data parameter list** (C02200/2)	0: Off	Switch on/off parameter changeover for the motor data parameter list.

16.1.4 Error message

With every run through the main program, one parameter of the definable parameter list is written until the parameter list has been executed completely. If an error occurs, C01083 displays an error status and C01084 displays the number of the list entry causing the error (in connection with the selected value set).

Please observe the note regarding the processing time of the main program in section

▸ LS_ParReadWrite_1-6 (📖 1811)

- If several errors occur at the same time, only the first incorrect list entry will be displayed. Hence, after elimination of the displayed error and another activation, more errors may be displayed.
- The parameter list will always be processed from beginning to end, even if errors occur in the meantime.

16.2 Configuring the definable parameter list by means of parameterisation

The following application example shows the necessary procedure for configuring the list without using the »Engineer« parameterisation dialog.

Task:

The **LS_WriteParamList** SB is to be used to write to parameters C00012, C00026/1 and C00027/1.

Compiling the parameter list

In C01085/1 ... n, specify the above-named parameters in the <Code>,<Subcode> format:

- C01085/1 = 12.000
- C01085/2 = 26.001
- C01085/3 = 27.001
- C01085/4 ... n = 0.000 (no parameter)

Note!

Gaps in the parameter list (setting = 0.000) are permissible and are skipped in the process.

Invalid parameter entries are not accepted when being entered.

Entering values for the parameters (value set 1)

In C01086/1 ... n, specify the values to be used to describe the selected parameters. The values are entered according to the scaling format/scaling factor of the respective parameter.

- C01086/1 = <value> for list entry 1 (in our example: for parameter C00012)
- C01086/2 = <value> for list entry 2 (in our example: for parameter C00026/1)
- C01086/3 = <value> for list entry 3 (in our example: for parameter C00027/1)

These values are used in the writing process if the two *bSelectWriteValue_1* and *bSelectWriteValue_2* inputs are not assigned or both set to FALSE.

Entering other values for the parameters (value sets 2 ... 4)

If required, up to three other sets can be set in the same way in C01087/1 ... n to C01089/1 ... n which can optionally be written to the parameters. The decision as to which value set is finally used is dependent upon the assignment of the two bSelectWriteValue_1 and bSelectWriteValue_2 inputs:

16.3 Configuring the motor data parameter list by means of parameterisation

The setting of the values for value sets 1 ... 4 of the motor data parameter list can also be carried out directly via the codes from code C02210 (see following table; columns 1st value ... 4th value).

- Unlike the "definable" parameter list, the motor data parameter list has a fixed assignment to the motor and control parameters.
- The preset values for value sets 1 ... 4 correspond to the Lenze setting of the respective motor or control parameter.
- The values are entered according to the scaling format / scaling factor of the respective motor or control parameter.

Code	Name	Lenze setting		1st value	2nd value	3rd value	4th value
C00006	Motor control	6		C02210/1	C02210/2	C02210/3	C02210/4
C00015	VFC: V/f base frequency	50.0	Hz	C02212/1	C02212/2	C02212/3	C02212/4
C00016	VFC: Vmin boost	1.60	%	C02213/1	C02213/2	C02213/3	C02213/4
C00018	Switching frequency	2		C02214/1	C02214/2	C02214/3	C02214/4
C00019	Auto-DCB: Threshold	3	rpm	C02215/1	C02215/2	C02215/3	C02215/4
C00021	Slip comp.	2.67	%	C02216/1	C02216/2	C02216/3	C02216/4
C00022	Imax in motor mode	47.00	A	C02217/1	C02217/2	C02217/3	C02217/4
C00023	Imax in generator mode	100.00	%	C02218/1	C02218/2	C02218/3	C02218/4
C00036	DC braking: Current	50.00	%	C02219/1	C02219/2	C02219/3	C02219/4
C00070/1	SLVC: Vp speed controller	15.00		C02220/1	C02220/2	C02220/3	C02220/4
C00070/2	SC: Vp speed controller	6.00		C02220/5	C02220/6	C02220/7	C02220/8
C00070/3	SLPSM: Vp speed controller	3.00		C02220/9	C02220/10	C02220/11	C02220/12
C00071/1	SLVC: Ti speed controller	100.0	ms	C02221/1	C02221/2	C02221/3	C02221/4
C00071/2	SC: Ti speed controller	50.0	ms	C02221/5	C02221/6	C02221/7	C02221/8
C00071/3	SLPSM: Ti speed controller	100.0	ms	C02221/9	C02221/10	C02221/11	C02221/12
C00072	SC: Tdn speed controller	0.00	ms	C02222/1	C02222/2	C02222/3	C02222/4
C00073/1	VFC: Vp Imax controller	0.25		C02223/1	C02223/2	C02223/3	C02223/4
C00073/2	SLVC: Vp torque controller	1.25		C02223/5	C02223/6	C02223/7	C02223/8
C00074/1	VFC: Ti Imax controller	65	ms	C02224/1	C02224/2	C02224/3	C02224/4
C00074/2	SLVC: Ti torque controller	30	ms	C02224/5	C02224/6	C02224/7	C02224/8
C00075	Vp current controller	7.00	V/A	C02225/1	C02225/2	C02225/3	C02225/4
C00076	Ti current controller	10.61	ms	C02226/1	C02226/2	C02226/3	C02226/4
C00077	SC: Vp field controller	12.80		C02227/1	C02227/2	C02227/3	C02227/4
C00078	SC: Tn field controller	256.0	ms	C02228/1	C02228/2	C02228/3	C02228/4
C00079/1	SC: Current controller - feedforward control	0		C02229/1	C02229/2	C02229/3	C02229/4
C00079/2	SC: adapt. field weak. controller	1		C02229/5	C02229/6	C02229/7	C02229/8
C00079/3	SC: n-Ctrl Anti-Wind-Up	0		C02229/9	C02229/10	C02229/11	C02229/12
C00080	Override point of field weakening	0	Hz	C02230/1	C02230/2	C02230/3	C02230/4
C00081	Rated motor power	11.00	kW	C02231/1	C02231/2	C02231/3	C02231/4
C00082	Motor rotor resistance	276	mohm	C02232/1	C02232/2	C02232/3	C02232/4
C00084	Motor stator resistance	330	mohm	C02233/1	C02233/2	C02233/3	C02233/4
C00085	Motor stator leakage inductance	3.50	mH	C02234/1	C02234/2	C02234/3	C02234/4
C00087	Rated motor speed	1460	rpm	C02236/1	C02236/2	C02236/3	C02236/4
C00088	Rated motor current	21.00	A	C02237/1	C02237/2	C02237/3	C02237/4
C00089	Rated motor frequency	50	Hz	C02238/1	C02238/2	C02238/3	C02238/4
C00090	Rated motor voltage	400	V	C02239/1	C02239/2	C02239/3	C02239/4
C00091	Motor cosine phi	0.85		C02240/1	C02240/2	C02240/3	C02240/4
C00092	Motor magnetising inductance	81.0	mH	C02241/1	C02241/2	C02241/3	C02241/4
C00095	Motor magnetising current	8.50	A	C02242/1	C02242/2	C02242/3	C02242/4
C00106	Auto-DCB: Hold time	0.500	s	C02244/1	C02244/2	C02244/3	C02244/4
C00107	DC braking: Hold time	999.000	s	C02245/1	C02245/2	C02245/3	C02245/4

Code	Name	Lenze setting		1st value	2nd value	3rd value	4th value
C00120	Setting of motor overload (I²xt)	100.00	%	C02246/1	C02246/2	C02246/3	C02246/4
C00234	Oscillation damping influence	5.00	%	C02249/1	C02249/2	C02249/3	C02249/4
C00235	Oscillation damping filter time	32	ms	C02250/1	C02250/2	C02250/3	C02250/4
C00236	Oscillation damping field weakening	14		C02251/1	C02251/2	C02251/3	C02251/4
C00254	Kp position controller	5.00	1/s	C02252/1	C02252/2	C02252/3	C02252/4
C00273	Motor moment of inertia	0.00	kg cm^2	C02256/1	C02256/2	C02256/3	C02256/4
C00495	Speed sensor selection	0		C02260/1	C02260/2	C02260/3	C02260/4
C00576	SC: Field feedforward control	200	%	C02261/1	C02261/2	C02261/3	C02261/4
C00577	SC: Vp field weakening controller	0.0010		C02262/1	C02262/2	C02262/3	C02262/4
C00578	SC: Tn field weakening controller	20.0	ms	C02263/1	C02263/2	C02263/3	C02263/4
C00653/1	Sensibility - Setpoint feedforward control	0		C02264/1	C02264/2	C02264/3	C02264/4
C00905	Motor phase direction of rotation	0		C02272/1	C02272/2	C02272/3	C02272/4
C00909/1	Max. pos. speed	120.00	%	C02273/1	C02273/2	C02273/3	C02273/4
C00909/2	Max. neg. speed	120.00	%	C02273/5	C02273/6	C02273/7	C02273/8
C00910/1	Max. pos. output frequency	1000	Hz	C02274/1	C02274/2	C02274/3	C02274/4
C00910/2	Max. neg. output frequency	1000	Hz	C02274/5	C02274/6	C02274/7	C02274/8
C00915	Motor cable length	5.0	m	C02275/1	C02275/2	C02275/3	C02275/4
C00916	Motor cable cross-section	6.00	mm^2	C02276/1	C02276/2	C02276/3	C02276/4
C00938	PSM: Maximum motor current field weakening	30.00	%	C02278/1	C02278/2	C02278/3	C02278/4
C00939	Ultimate motor current	3000.0	A	C02279/1	C02279/2	C02279/3	C02279/4
C00965	Max. motor speed	60000	rpm	C02280/1	C02280/2	C02280/3	C02280/4
C00966	VFC: Time const. slip comp.	100	ms	C02281/1	C02281/2	C02281/3	C02281/4
C00971/1	VFC: Controller limitation V/f +encoder	10.00	Hz	C02284/1	C02284/2	C02284/3	C02284/4
C00971/2	VFC: Slip limitation V/f +encoder	100.00	Hz	C02284/5	C02284/6	C02284/7	C02284/8
C00972	VFC: Vp V/f +encoder	0.100	Hz/Hz	C02285/1	C02285/2	C02285/3	C02285/4
C00973	VFC: Ti V/f +encoder	100.0	ms	C02286/1	C02286/2	C02286/3	C02286/4
C00975	VFC-ECO: Vp CosPhi controller	0.500	Hz/Hz	C02287/1	C02287/2	C02287/3	C02287/4
C00976	VFC-ECO: Ti CosPhi controller	200.0	ms	C02288/1	C02288/2	C02288/3	C02288/4
C00977	VFC-ECO: Minimum voltage V/f	20.00	%	C02289/1	C02289/2	C02289/3	C02289/4
C00982	VFC-ECO: Voltage reduction ramp	0.8	s	C02290/1	C02290/2	C02290/3	C02290/4
C00985	SLVC: Gain of field current controller	0.50	%	C02291/1	C02291/2	C02291/3	C02291/4
C00986	SLVC: Gain of cross current controller	0.00	%	C02292/1	C02292/2	C02292/3	C02292/4
C00987	Inverter motor brake: nAdd	80	rpm	C02293/1	C02293/2	C02293/3	C02293/4
C00988	Inverter motor brake: PT1 filter time	0.0	ms	C02294/1	C02294/2	C02294/3	C02294/4
C00990	Flying restart fct.: Activate	0		C02295/1	C02295/2	C02295/3	C02295/4
C00991	Flying restart fct.: Process	2		C02296/1	C02296/2	C02296/3	C02296/4
C00992	Flying restart: Start frequency	10	Hz	C02297/1	C02297/2	C02297/3	C02297/4
C00993	Flying restart: Integration time	300.0	ms	C02298/1	C02298/2	C02298/3	C02298/4
C00994	Flying restart: Current	25.00	%	C02299/1	C02299/2	C02299/3	C02299/4
C00995/1	SLPSM: Controlled accelerating current	100.00	%	C02300/1	C02300/2	C02300/3	C02300/4
C00995/2	SLPSM: Controlled standstill current	20.00	%	C02300/5	C02300/6	C02300/7	C02300/8
C00996/1	SLPSM: Switching speed, closed-loop control	13.00	%	C02301/1	C02301/2	C02301/3	C02301/4
C00996/2	SLPSM: Switching speed, open-loop control	8.00	%	C02301/5	C02301/6	C02301/7	C02301/8
C00997	SLPSM: Filter cutoff frequency	5.00	%	C02302/1	C02302/2	C02302/3	C02302/4
C00998/1	SLPSM: Filter time rotor position	3.0	ms	C02303/1	C02303/2	C02303/3	C02303/4
C00998/2	SLPSM: Filter time actual speed value	5.0	ms	C02303/5	C02303/6	C02303/7	C02303/8
C00999	SLPSM: PLL gain	400	%	C02304/1	C02304/2	C02304/3	C02304/4
C01001/1	Manual entry of motor type	0		C02315/1	C02315/2	C02315/3	C02315/4
C02853/1	PSM: Lss saturation characteristic	100	%	C02305/1	C02305/2	C02305/3	C02305/4
C02853/2	PSM: Lss saturation characteristic	100	%	C02305/5	C02305/6	C02305/7	C02305/8
C02853/3	PSM: Lss saturation characteristic	100	%	C02305/9	C02305/10	C02305/11	C02305/12
C02853/4	PSM: Lss saturation characteristic	100	%	C02305/13	C02305/14	C02305/15	C02305/16

Code	Name	Lenze setting		1st value	2nd value	3rd value	4th value
C02853/5	PSM: Lss saturation characteristic	100	%	C02305/17	C02305/18	C02305/19	C02305/20
C02853/6	PSM: Lss saturation characteristic	100	%	C02305/21	C02305/22	C02305/23	C02305/24
C02853/7	PSM: Lss saturation characteristic	100	%	C02305/25	C02305/26	C02305/27	C02305/28
C02853/8	PSM: Lss saturation characteristic	100	%	C02305/29	C02305/30	C02305/31	C02305/32
C02853/9	PSM: Lss saturation characteristic	100	%	C02305/33	C02305/34	C02305/35	C02305/36
C02853/10	PSM: Lss saturation characteristic	100	%	C02305/37	C02305/38	C02305/39	C02305/40
C02853/11	PSM: Lss saturation characteristic	100	%	C02305/41	C02305/42	C02305/43	C02305/44
C02853/12	PSM: Lss saturation characteristic	100	%	C02305/45	C02305/46	C02305/47	C02305/48
C02853/13	PSM: Lss saturation characteristic	100	%	C02305/49	C02305/50	C02305/51	C02305/52
C02853/14	PSM: Lss saturation characteristic	100	%	C02305/53	C02305/54	C02305/55	C02305/56
C02853/15	PSM: Lss saturation characteristic	100	%	C02305/57	C02305/58	C02305/59	C02305/60
C02853/16	PSM: Lss saturation characteristic	100	%	C02305/61	C02305/62	C02305/63	C02305/64
C02853/17	PSM: Lss saturation characteristic	100	%	C02305/65	C02305/66	C02305/67	C02305/68
C02855	PSM: Imax Lss saturation characteristic	3000.0	A	C02306/1	C02306/2	C02306/3	C02306/4
C02859	PSM: Activate Ppp saturation char.	0		C02307/1	C02307/2	C02307/3	C02307/4
C02872/1	PLI without movement: adaptation of time duration	0		C02311/1	C02311/2	C02311/3	C02311/4
C02874/1	PLI without movement	0x0001		C02312/1	C02312/2	C02312/3	C02312/4
C02875/1	PLI without movement: adaptation of ident angle	0	°	C02313/1	C02313/2	C02313/3	C02313/4

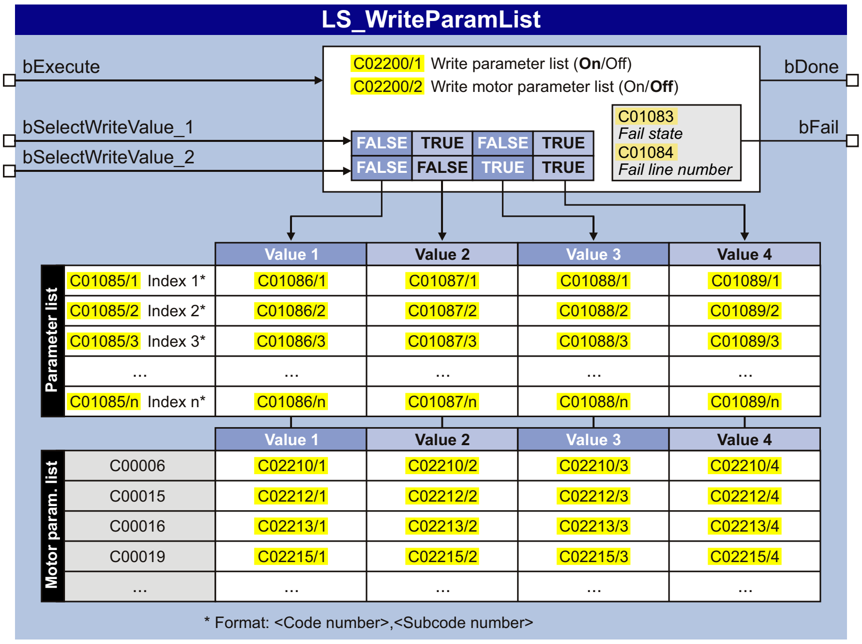
16.4 Internal interfaces | System block "LS_WriteParamList"

The **LS_WriteParamList** system block provides the internal interfaces for the basic "Parameter change-over" function.

Note!

The "parameter change-over" basic function is always processed, even if the **LS_WriteParamList** system block has been removed from the interconnection by the FB Editor.

If you do not require this basic function anymore, delete the composed parameter list in order that no unwanted parameter write operations take place.



inputs

Designator Data type	Information/possible settings		
bExecute BOOL	FALSE↗TRUE	For Execute Mode (C01082) = "0: by Execute": Activate writing of the parameter list	
bSelectWriteValue_1 bSelectWriteValue_2 BOOL	Binary coded selection of the value set 1 ... 4 to be used.		
	bSelectWrite Value_1	*bSelectWrite Value_2*	
	FALSE	FALSE	Value set 1
	TRUE	FALSE	Value set 2
	FALSE	TRUE	Value set 3
	TRUE	TRUE	Value set 4

outputs

Designator Data type	Value/meaning	
bDone BOOL	"Writing of the parameter list completed" status signal • The output is automatically reset to FALSE if writing via *bExecute* is activated again.	
	TRUE	Writing of the parameter list successfully completed.
	FALSE	The FALSE status can have the following meanings: 1. There is no active writing of the parameter list. 2. Writing of the parameter list has not been completed yet. 3. An error has occurred (if *bFail* = TRUE).
bFail BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • See display parameter (C01083) for details.

17 Parameter reference

This chapter describes all parameters which can be used for parameterising and monitoring the inverter.

Parameters which are only available in the inverter from a certain software version onwards are marked with a corresponding note in the parameter description ("from version xx.xx.xx").

The parameter descriptions are based on software version V21.00.00

Tip!

For quick reference of a parameter with a certain name simply use the **index** of the online documentation. The index always contains the corresponding code in parentheses behind the name.

General information on parameter setting can be found in the chapter "Introduction: Parameterising the inverter". (🕮 34)

For general information on how to read and change parameters, please see the online documentation for the »Engineer«.

17.1 Structure of the parameter descriptions

Each parameter is described in the Parameter list in the form of a table which consists of the following three areas:

Table header

The table header contains the following general information:

- Parameter number (Cxxxxx)
- Parameter name (display text in the »Engineer» and keypad)
- Data type
- Parameter index in decimal and hexadecimal notation for access via a fieldbus (e.g. CAN system bus).

Tip!

The parameter index is calculated as follows:

- Index [dec] = 24575 - code
- Index [hex] = 0x5FFF - code

Example for code C00005:

- Index [dec] = 24575 - 5 = 24570
- Index [hex] = 0x5FFF - 0x{5} = 0x5FFA

Table contents

The table contains further general explanations & notes on the parameter and the possible settings, which are represented in different ways depending on the parameter type:

- Parameters with read-only access
- Parameters with write access

Table footer

The table footer contains the Parameter attributes.

17.1.1 Data type

The following data types are available for parameters:

Data type	Meaning
INTEGER_16	16-bit value with sign
INTEGER_32	32-bit value with sign
UNSIGNED_8	8-bit value without sign
UNSIGNED_16	16-bit value without sign
UNSIGNED_32	32-bit value without sign
VISIBLE_STRING	String of characters of printable characters

17.1.2 Parameters with read-only access

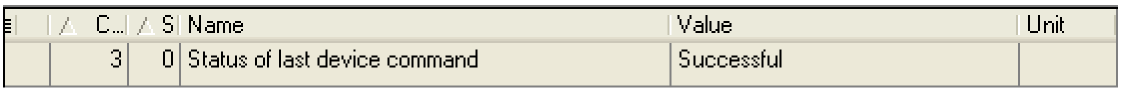
Parameters for which the "write access" attribute has not been set can only be read and not be changed by the user.

Description structure

Parameter | Name: **Cxxxxx | ______________**				Data type: _______ Index: _______
Description				
Display range **(min. value | unit | max. value)**				
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1				

Representation in the »Engineer«

The »Engineer« displays these parameters with a grey background or, with an online connection, with a pale-yellow background:



17.1.3 Parameters with write access

Only parameters with a check mark (☑) in front of the "write access" attribute can be changed by the user. The Lenze setting for these parameters is **printed in bold**.

- The settings can either be selected from a selection list or the values can be entered directly.
- Values outside the valid setting range are represented in red in the »Engineer«.

17.1.3.1 Parameters with setting range

Description structure

Parameter | Name: **Cxxxxx | ______________**				Data type: ______ Index: ______
Description				
Setting range **(min. value | unit | max. value)**			**Lenze setting**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1				

Parameter setting in the »Engineer«

In the »Engineer«, parameters are set by entering the desired value into the input field:

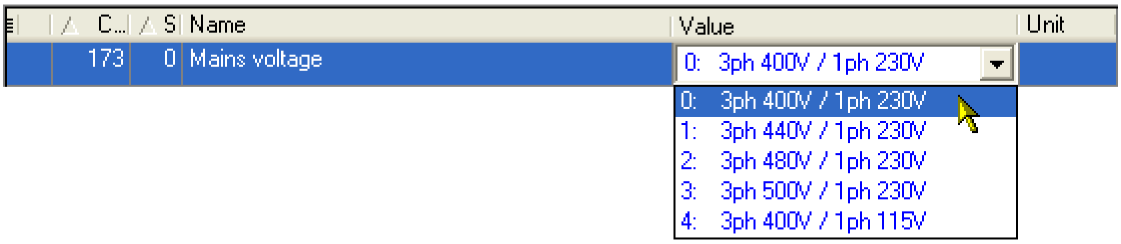
17.1.3.2 Parameters with selection list

Description structure

Parameter | Name: **Cxxxxx | ______________**			Data type: ______ Index: ______
Description			
Selection list **(Lenze setting printed in bold)**			
1			
2			
3			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

Parameter setting in the »Engineer«

In the »Engineer«, a list field is used for parameter setting:



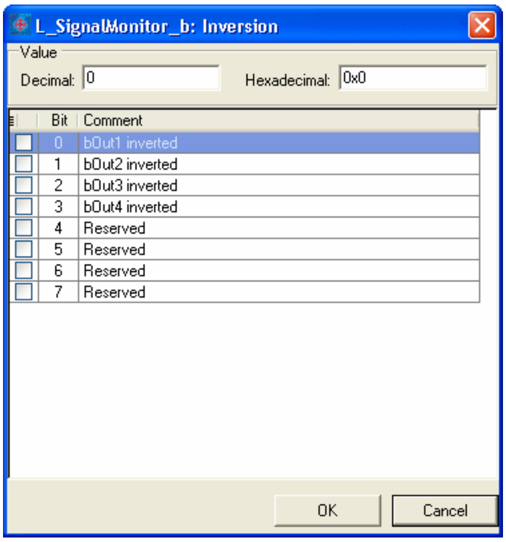
17.1.3.3 Parameters with bit-coded setting

Description structure

Parameter | Name: **Cxxxxx | ____________**		Data type: _______ Index: _______
Description		
Value is bit-coded:		
Bit 0		
...		
Bit 31		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

Parameter setting in the »Engineer«

The »Engineer« uses a dialog box for parameter setting in which the individual bits can be set or reset. Alternatively, the value can be entered as a decimal or hexadecimal value:



17.1.3.4 Parameters with subcodes

Description structure

Parameter | Name: **Cxxxxx | ____________**			Data type: _______ Index: _______
Description			
Setting range **(min. value | unit | max. value)**			
Subcodes	**Lenze setting**		
Cxxxxx/1			
Cxxxxx/2			
Cxxxxx/3			
Cxxxxx/4			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

Parameter setting in the »Engineer«

The »Engineer« parameter list displays each subcode individually. The parameters are set as described in the previous chapters.

	C...	S	Name	Value	Unit
	39	1	Fixed setpoint 1	40.00	%
	39	2	Fixed setpoint 2	60.00	%
	39	3	Fixed setpoint 3	80.00	%
	39	4	Fixed setpoint 4	0.00	%

17.1.4 Parameter attributes

The table footers contain the parameter attributes:

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1

Attribute	Meaning	
☑ Read access	Read access to parameter possible.	
☑ Write access	Write access to parameter possible. • Please also observe the following attributes:	
	☑ CINH	Parameter value can only be changed when the controller is inhibited.
	☑ PLC STOP	Parameter value can only be changed when the application is stopped.
☑ No transfer	Parameter is **not** transferred to inverter when the command Download parameter set is executed.	
☑ COM	Communication-relevant parameter • This parameter is relevant for parameter data transfer via the (CAN) system bus.	
☑ MOT	Motor control parameters	

Scaling factor

The "scaling factor" is important for parameter access via a bus system.

Signal type	Scaling factor	Resolution	Value range
Analog (scaled)	100	16 bits signed	± 199.99 %
Angular velocity	1	16 bits signed	± 32767 incr./ms
Position in [units]	10000	32 bits signed	± 214748.3647 [units]
Digital (BOOL)	1	8 bits unsigned	0 ≡ FALSE; 1 ≡ TRUE
Time	1000	16 bits unsigned	0 ... 999.000 s
Selection value	1	16 bits unsigned	0 ... 65535

Example 1: The value "654" of the parameter C00028/1 (AIN1: input voltage) read via a bus system must be divided by the corresponding scaling factor "100" to obtain the actual display value "6.54 V".

$$\frac{\text{Read value (via bus system)}}{\text{Scaling factor}} = \text{Indicated value (Engineer)}$$

[17-1] Conversion formula for read access via bus system

Example 2: In order to set the parameter C00012 (acceleration time main setpoint) to the value "123.45 %" via a bus system, the integer value "12345" must be transferred, i.e. the value to be set must be multiplied by the corresponding scaling factor "100".

$$\text{Value to be written (via bus system)} = \text{Value to be set} \cdot \text{Scaling factor}$$

[17-2] Conversion formula for write access via bus system

Character length

In case of parameters of "VISIBLE_STRING" data type, the character length is given in addition. This is also important for the parameter access via a bus system.

17.2 Parameter list

This chapter lists all parameters of the operating system in numerically ascending order.

Note!

The parameter descriptions are based on the software version V16.00.00.

C00001

Parameter | Name: **C00001 | Keypad Userlevel**		Data type: UNSIGNED_8 Index: 24574_d = $5FFE_h$
From version 12.00.00 onwards, the extent of menus, submenus and codes shown in the keypad can be adapted by selecting the "userlevel". • When the userlevel has been changed, the menus of the keypad will be reconfigured according to the selected user level. • Independently of the set userlevel, the parameters of the attached communication module will always be displayed completely.		
Selection list		**Info**
10	Standard	Only the most important menus and codes are displayed on the keypad.
20	Expert	All menus and codes are displayed on the keypad.
30	Service	Only for servicing purposes (Lenze service).
Subcodes	**Lenze setting**	**Info**
C00001/1	10: Standard	Keypad UserLevel
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00002

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_8 Index: 24573_d = $5FFD_h$
Note: • Before switching off the supply voltage after a device command has been executed, check the successful execution of the device command via the status display in C00003! • Before activating device commands by a master control, wait for the "Ready" signal of the inverter. • The device will reject a write process to C00002/x if the value is >1 and issue an error message. ▸ Drive control (DCTRL): Device commands		
Selection list		
0	Off / ready	
1	On / start	
4	Action cancelled	
5	No access	
6	No access controller inhibit	
20	20% working	
40	40% working	
60	60% working	
80	80% working	
Subcodes	**Lenze setting**	**Info**

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_8 Index: 24573_d = $5FFD_h$
C00002/1	0: Off / ready	Load Lenze setting • All parameters are reset to the Lenze setting. • Only possible when the controller is inhibited. ▶ Load Lenze setting
C00002/2	0: Off / ready	Reserved
C00002/3	0: Off / ready	Reserved
C00002/4	0: Off / ready	Reserved
C00002/5	0: Off / ready	Reserved
C00002/6	0: Off / ready	Load all parameter sets • All parameter sets are loaded by the memory module. • Only possible when the controller is inhibited. ▶ Load all parameter sets
C00002/7	0: Off / ready	Reserved
C00002/8	0: Off / ready	Reserved
C00002/9	0: Off / ready	Reserved
C00002/10	0: Off / ready	Save oscilloscope configuration
C00002/11	0: Off / ready	Save all parameter sets • All parameter sets are saved to the memory module safe against mains failure. ▶ Save all parameter sets
C00002/12	0: Off / ready	Reserved
C00002/13	0: Off / ready	Reserved
C00002/14	0: Off / ready	Reserved
C00002/15	0: Off / ready	Reserved
C00002/16	1: On / start	Enable inverter 1 ≡ Enable inverter 0 ≡ Inhibit inverter ▶ Enable/inhibit inverter
C00002/17	0: Off / ready	Activate quick stop 1 ≡ Activate quick stop 0 ≡ Deactivate quick stop ▶ Activate/deactivate quick stop
C00002/18	0: Off / ready	Reserved
C00002/19	0: Off / ready	Reset error • After the reset (acknowledgement) of the current error, further errors may be pending which must also be reset. • The status determining error is displayed in C00168. • The current error is displayed in C00170. ▶ Error messages of the operating system
C00002/20	0: Off / ready	Reserved
C00002/21	0: Off / ready	Delete logbook • All entries in the logbook of the inverter are deleted. • In the logbook, information on the error history is saved. ▶ Logbook
C00002/22	0: Off / ready	Reserved

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_8 Index: 24573_d = $5FFD_h$
C00002/23	0: Off / ready	Identify motor parameters • This device command serves to carry out automatic motor parameter identification. • The identification method can be selected in C02867. • The device command is only executed when the inverter is in the "SwitchedOn" status. • In order to identify the motor parameters, the inverter must be enabled after this device command. ▶ Automatic motor parameter identification
C00002/24	0: Off / ready	Setting Hiperface position • Ab Version 14.00.00 ▶ SinCos absolute value encoder with HIPERFACE® protocol
C00002/25	0: Off / ready	Identify resolver error • The device command is only executed when the inverter is in the "SwitchedOn" status. e ▶ Optimise resolver behaviour
C00002/26	0: Off / ready	CAN reset node • Reinitialise "CAN on board" interface. • Required when changing the baud rate, node address, or identifiers. ▶ "CAN on board" system bus
C00002/27	0: Off / ready	Device search function • This device command serves to optically locate an inverter connected online (e.g. for maintenance work). ▶ Device search function
C00002/28	0: Off / ready	Check MasterPin ▶ Unlocking the inverter with a MasterPin
C00002/29	0: Off / ready	Set binding ID ▶ Device personalisation
C00002/30	0: Off / ready	Delete binding ID ▶ Device personalisation
C00002/31	0: Off / ready	Set password ▶ Password protection
C00002/32	0: Off / ready	Check password ▶ Password protection
C00002/33	0: Off / ready	Delete password ▶ Password protection
C00002/34	0: Off / ready	Pole position identification (360°) • This device command is serves to carry out an automatic identification of the pole position. • The device command is only executed when the inverter is in the "SwitchedOn" status. • In order to identify the pole position, the inverter must be enabled after this device command. ▶ Pole position identification
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00003

Parameter | Name: **C00003 | Status of last device command**		Data type: UNSIGNED_8 Index: 24572_d = $5FFC_h$
Status of the device command executed last (C00002). **Note:** Before switching off the supply voltage after carrying out a device command, check whether the device command has been carried out successfully via the status display! ▶ Drive control (DCTRL): Device commands		
Selection list (read only)		**Info**
0	Successful	Device command has been executed successfully.
1	Command unknown	Device command implausible or unknown to the system.
2	Password protection	Unauthorised access for requested device command. ▶ Password protection
3	Time-out	Device command could not be processed in the defined time (timeout).
4	System fault	
5	Command server assigned	
6	Controller inhibit required	
7	unexpected controller inhibit	
10	Memory module binding error	▶ Device personalisation
11	Password too short	▶ Password protection
12	Wrong password	
13	Password already set	
14	Password not assigned	
15	Incorrect checksum	
20	Setting the hiperface position failed	
30	Incorrect pole position identification	
31	Pole position identification started without encoder	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00005

Parameter | Name: **C00005 | Application**		Data type: UNSIGNED_16 Index: 24570_d = $5FFA_h$
Selection of the technology application		
Selection list(Lenze setting printed in bold)		**Info**
0	Wiring has changed	This display appears if the FB interconnection has been changed in the application level using the FB Editor.
1000	**Actuating drive speed**	This technology application is used to solve speed-controlled drive tasks, e.g. conveying belts. ▶ TA "Speed actuating drive"
1100	Actuating drive speed (AC Drive Profile)	From version 13.00.00 This technology application is used to solve speed-controlled drive tasks, e.g. conveying belts. The process data word received by the master control is interpreted as "AC Drive Profile" control word. ▶ TA "Actuating drive speed (AC Drive Profile)"
2000	Table positioning	This technology application is used to solve position-controlled drive tasks which are usually controlled by a higher-level control system using a fieldbus. ▶ TA "Table positioning"
3000	Switch-off positioning	This technology application is used to solve speed-controlled drive tasks which require a pre-switch off or stopping at certain positions, e.g. roller conveyors and conveying belts. This is implemented by connecting switch-off sensors. ▶ TA "Switch-off positioning"
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00006

Parameter | Name: **C00006 | Motor control**		Data type: UNSIGNED_8 Index: 24569_d = $5FF9_h$
Selection of the motor control mode ▶ Motor control (MCTRL): Select control mode		
Selection list(Lenze setting printed in bold)		**Info**
1	SC: SM servo control	This control type is used for the servo control of a synchronous motor. • The control type requires a speed feedback via an encoder mounted to the motor! ▶ Servo control
2	SC: Servo control ASM	This control type is used for the servo control of an asynchronous motor. • The control type requires a speed feedback via an encoder mounted to the motor! ▶ Servo control
3	SLPSM: Sensorless PSM	This control type is used for the sensorless control of a synchronous motor. ▶ Sensorless control for synchronous motors
4	SLVC: Vector control	This control type is used for sensorless vector control of an asynchronous motor. • The control type requires motor parameters to be set as exactly as possible! ▶ Sensorless vector control

Parameter | Name: **C00006 | Motor control**		Data type: UNSIGNED_8 Index: 24569_d = $5FF9_h$
6	**VFCplus: V/f linear**	This control type is used for the speed control of an asynchronous motor via a linear V/f characteristic and is the simplest control type. • For setting the V/f characteristic, only the rated frequency (C00089) and the rated voltage (C00090) of the motor have to be entered. ▸ V/f characteristic control
7	VFCplus: V/f linear + encoder	This control type is used for speed control of an asynchronous motor via a linear V/f characteristic. • The control type requires a speed feedback via an encoder mounted to the motor! • For setting the V/f characteristic, only the rated frequency (C00089) and the rated voltage (C00090) of the motor have to be entered. ▸ V/f control
8	VFCplus: V/f quadr	This control type is used for speed control of an asynchronous motor via a square-law V/f characteristic. • For setting the V/f characteristic, only the rated frequency (C00089) and the rated voltage (C00090) of the motor have to be entered. ▸ V/f characteristic control
9	VFCplus: V/f quadr + encoder	This control type is used for speed control of an asynchronous motor via a square-law V/f characteristic. • The control type requires a speed feedback via an encoder mounted to the motor! • For setting the V/f characteristic, only the rated frequency (C00089) and the rated voltage (C00090) of the motor have to be entered. ▸ V/f control
10	VFCplus: V/f definable	This type of control is used for the speed control of an asynchronous motor via a user-definable characteristic with several interpolation points. • For setting the V/f characteristic, only the rated frequency (C00089) and the rated voltage (C00090) of the motor have to be entered. ▸ V/f characteristic control
11	VFCplusEco: V/f energy-saving	This control type is used for energy-saving speed control of an asynchronous motor via a linear V/f characteristic. • For setting the V/f characteristic, only the rated frequency (C00089) and the rated voltage (C00090) of the motor have to be entered. • Predestinated application areas of this control type are materials handling technology and pump and fan systems. ▸ V/f characteristic control, energy-saving
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00007

Parameter | Name: **C00007 | Control mode**		Data type: UNSIGNED_16 Index: 24568_d = $5FF8_h$
With this parameter the control mode for the technology application selected under C00005 is defined, i.e. how the inputs and outputs of the technology application are connected to the I/Os of the inverter. • How the inputs and outputs are connected in the individual control modes is described in the corresponding technology application: • "Actuating drive speed" TA • TA "actuating drive speed (AC Drive Profile)" • TA "Table positioning" • "Switch-off positioning" TA		
Selection list(Lenze setting printed in bold)		**Info**
0	Wiring has changed	This is displayed when the FB interconnection has been changed in the I/O level via the FB Editor.
10	**Terminals 0**	The technology application is controlled via the digital and analog input terminals of the inverter. • A short overview of the preconfigured terminal assignment is included in the description of the respective technology application in the "Terminal assignment of the control modes".
12	Terminals 2	
14	Terminals 11	
16	Terminal 16	
20	Keypad	The technology application is controlled via the keypad.
21	PC	The technology application is controlled via the "Free parameters" of the inverter (PC control).
30	CAN	The technology application is controlled by means of CAN-PDOs via the system bus "CAN on board". ▶ "CAN on board" system bus
40	MCI	The technology application is controlled by means of MCI-PDOs via the MCI-interface of an attached communication module (e.g. PROFIBUS).
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00008

Parameter | Name: **C00008 | Original application|control source**		Data type: UNSIGNED_16 Index: 24567_d = $5FF7_h$
Display of the originally selected technology application and the originally selected control mode. • This parameter shows the selection that was set with C00005 and C00007 before a change in the I/O level or the application level was carried out. • For diagnostic purposes, this display serves to determine whether there is a standard interconnection in the inverter or a change carried out by the user.		
Selection list (read only)		**Info**
0	Free|Free	Application: Interconnection has been changed. I/O level: Interconnection has been changed.
10	Free|Terminal0	Application: Interconnection has been changed. I/O level: "Terminals 0" control mode
12	Free|Terminal2	Application: Interconnection has been changed. I/O level: "Terminal 2" control mode
14	Free|Terminal11	Application: Interconnection has been changed. I/O level: "Terminal 11" control mode
16	Free|Terminal 16	Application: Interconnection has been changed. I/O level: "Terminal 16" control mode
20	Free|Keypad	Application: Interconnection has been changed. I/O level: "Keypad" control mode
21	Free|PC	Application: Interconnection has been changed. I/O level: "PC" control mode

Parameter | Name: **C00008 | Original application|control source**		Data type: UNSIGNED_16 Index: 24567_d = $5FF7_h$
30	Free|CAN	Application: Interconnection has been changed. I/O level: "CAN" control mode
40	Free|MCI	Application: Interconnection has been changed. I/O level: "MCI" control mode
1000	Speed|Free	Application: Actuating drive speed I/O level: Interconnection has been changed.
1010	Speed|Terminal0	Application: Actuating drive speed I/O level: "Terminals 0" control mode
1012	Speed|Terminal2	Application: Actuating drive speed I/O level: "Terminal 2" control mode
1014	Speed|Terminal11	Application: Actuating drive speed I/O level: "Terminal 11" control mode
1016	Speed|Terminal16	Application: Actuating drive speed I/O level: "Terminal 16" control mode
1020	Speed|Keypad	Application: Actuating drive speed I/O level: "Keypad" control mode
1021	Speed|PC	Application: Actuating drive speed I/O level: "PC" control mode
1030	Speed|CAN	Application: Actuating drive speed I/O level: "CAN" control mode
1040	Speed|MCI	Application: Actuating drive speed I/O level: "MCI" control mode
1100	SpeedACdrive|Free	Application: Actuating drive speed (AC Drive Profile) I/O level: Interconnection has been changed.
1110	SpeedACdrive|Terminal0	Application: Actuating drive speed (AC Drive Profile) I/O level: "Terminals 0" control mode
1112	SpeedACdrive|Klemme2	Application: Actuating drive speed (AC Drive Profile) I/O level: "Terminal 2" control mode
1114	SpeedACdrive|Klemme11	Application: Actuating drive speed (AC Drive Profile) I/O level: "Terminal 11" control mode
1116	SpeedACdrive|Klemme16	Application: Actuating drive speed (AC Drive Profile) I/O level: "Terminal 16" control mode
1120	SpeedACdrive|Keypad	Application: Actuating drive speed (AC Drive Profile) I/O level: "Keypad" control mode
1121	SpeedACdrive|PC	Application: Actuating drive speed (AC Drive Profile) I/O level: "PC" control mode
1130	SpeedACdrive|CAN	Application: Actuating drive speed (AC Drive Profile) I/O level: "CAN" control mode
1140	SpeedACdrive|MCI	Application: Actuating drive speed (AC Drive Profile) I/O level: "MCI" control mode
2000	TabPos|Free	Application: Table positioning I/O level: Interconnection has been changed.
2010	TabPos|Terminal0	Application: Table positioning I/O level: "Terminals 0" control mode
2012	TabPos|Terminal2	Application: Table positioning I/O level: "Terminal 2" control mode
2014	TabPos|Terminal11	Application: Table positioning I/O level: "Terminal 11" control mode
2016	TabPos|Terminal16	Application: Table positioning I/O level: "Terminal 16" control mode

Parameter | Name: **C00008 | Original application|control source**		Data type: UNSIGNED_16 Index: 24567_d = $5FF7_h$
2020	TabPos|Keypad	Application: Table positioning I/O level: "Keypad" control mode
2021	TabPos|PC	Application: Table positioning I/O level: "PC" control mode
2030	TabPos|CAN	Application: Table positioning I/O level: "CAN" control mode
2040	TabPos|MCI	Application: Table positioning I/O level: "MCI" control mode
3000	SwitchPos|Free	Application: Switch-off positioning I/O level: Interconnection has been changed.
3010	SwitchPos|Terminal0	Application: Switch-off positioning I/O level: "Terminals 0" control mode
3012	SwitchPos|Terminal 2	Application: Switch-off positioning I/O level: "Terminal 2" control mode
3014	SwitchPos|Terminal 11	Application: Switch-off positioning I/O level: "Terminal 11" control mode
3016	SwitchPos|Terminal 16	Application: Switch-off positioning I/O level: "Terminal 16" control mode
3020	SwitchPos|Keypad	Application: Switch-off positioning I/O level: "Keypad" control mode
3021	SwitchPos|PC	Application: Switch-off positioning I/O level: "PC" control mode
3030	SwitchPos|CAN	Application: Switch-off positioning I/O level: "CAN" control mode
3040	SwitchPos|MCI	Application: Switch-off positioning I/O level: "MCI" control mode
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00010

Parameter | Name: **C00010 | AIN1: Characteristic**			Data type: INTEGER_16 Index: 24565_d = $5FF5_h$
			▸ Analog terminals: Signal adaptation via characteristic
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00010/1	0.00 %		AIN1: (+y0) = min
C00010/2	0.00 %		AIN1: (+x0) = Dead band
C00010/3	0.00 %		AIN1: (-y0) = (-min)
C00010/4	0.00 %		AIN1: (-x0) = (-Dead band)
C00010/5	100.00 %		AIN1: (+ymax)
C00010/6	100.00 %		AIN1: (+xmax)
C00010/7	100.00 %		AIN1: (-ymax)
C00010/8	100.00 %		AIN1: (-xmax)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00011

Parameter | Name: **C00011 | Appl.: Reference speed**			Data type: UNSIGNED_16 Index: 24564_d = $5FF4_h$
Setting the reference speed • In the inverter, all speed-related signals are processed to one reference variable in percent. • Set a reference speed here that corresponds to 100 %. • The frequency that corresponds to the set reference speed is displayed in C00059. **Note:** This is not a maximum limitation! All values in percent in the inverter may be in a range of 0 ... 199.99 %. **Recommendation:** Deactivate the torque feedforward control for the speed setpoint in C00654/1 if the reference speed C00011 is 5 times higher than the rated motor speed C00087			
Setting range (min. value | unit | max. value)			**Lenze setting**
50	rpm	60000	**1500 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00012

Parameter | Name: **C00012 | Accel. time - main setpoint**			Data type: UNSIGNED_32 Index: 24563_d = $5FF3_h$
The L_NSet_1 FB: Acceleration time of the ramp generator for the main speed setpoint • Generally, this ramp generator is used for all speed-controlled technology applications.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**2.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00013

Parameter | Name: **C00013 | Decel. time - main setpoint**			Data type: UNSIGNED_32 Index: 24562_d = $5FF2_h$
The L_NSet_1 FB: Deceleration time of the ramp generator for the main speed setpoint • Generally, this ramp generator is used for all speed-controlled technology applications.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**2.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00015

Parameter | Name: **C00015 | VFC: V/f base frequency**			Data type: UNSIGNED_16 Index: 24560_d = $5FF0_h$
V/f base frequency for V/f characteristic control (VFCplus) and V/f control (VFCplus+encoder) • The motor voltage increases linearly with the frequency until the base frequency is reached. From this value on, the motor voltage remains constant, the speed increases and the maximum torque decreases. • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well.			
Setting range (min. value | unit | max. value)			**Lenze setting**
7.5	Hz	2600.0	**50.0 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00016

Parameter | Name: **C00016 | VFC: Vmin boost**			Data type: UNSIGNED_16 Index: 24559_d = $5FEF_h$
Boost of the V/f voltage characteristic in the range of small speeds or frequencies with V/f characteristic control (VFCplus) and V/f control (VFCplus+encoder) • This may increase the starting torque. • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well. ▶ Motor control (MCTRL): Setting the Vmin boost			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00018

Parameter | Name: **C00018 | Switching frequency**		Data type: UNSIGNED_8 Index: 24557_d = $5FED_h$
Selection of the pulse width modulated switching frequency transferred from the inverter to the motor • Select between an ideal setting for the drive which provides smooth running, and an optimal setting with regard to the inverter which keeps its losses to a minimum (min. Pv). • Both possibilities offer fixed and variable switching frequencies. • When a variable switching frequency is selected, the switching frequency may change as a function of the load and rotational frequency. ▶ Selection of switching frequency		
Selection list (Lenze setting printed in bold)		
1	4 kHz var./drive-optimised	
2	**8 kHz var./drive-optimised**	
3	16 kHz var./drive-optimised	
5	2 kHz constant/drive-optimised	
6	4 kHz constant/drive-optimised	
7	8 kHz constant/drive-optimised	
8	16 kHz constant/drive-optimised	
11	4 kHz var./min. Pv	
12	8 kHz var./min. Pv	
13	16 kHz var./min. Pv	
15	2 kHz constant/min. Pv	
16	4 kHz constant/min. Pv	
17	8 kHz constant/min. Pv	
18	16 kHz constant/min. Pv	
21	8 kHz var./drive-opt./4 kHz min	
22	16 kHz var./drive-opt./4 kHz min	
23	16 kHz var./drive-opt./8 kHz min	
31	8 kHz var./min. Pv/4 kHz min	
32	16 kHz var./min. Pv/4 kHz min	
33	16 kHz var./min. Pv/8 kHz min	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00019

Parameter | Name: **C00019 | Auto DCB: Threshold**			Data type: UNSIGNED_16 Index: 24556_d = $5FEC_h$
Setpoint speed threshold for automatic DC injection braking • For speed setpoints with values below the thresholds a DC current is injected or the motor is not supplied with current, depending on the setting. ▸ DC-injection braking			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	60000	**3 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00020

Parameter | Name: **C00020 | AIN2: Characteristic**			Data type: INTEGER_16 Index: 24555_d = $5FEB_h$
▸ Analog terminals: Signal adaptation via characteristic			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00020/1	0.00 %		AIN2: (+y0) = min
C00020/2	0.00 %		AIN2: (+x0) = Dead band
C00020/3	0.00 %		AIN2: (-y0) = (-min)
C00020/4	0.00 %		AIN2: (-x0) = (-Dead band)
C00020/5	100.00 %		AIN2: (+ymax)
C00020/6	100.00 %		AIN2: (+xmax)
C00020/7	100.00 %		AIN2: (-ymax)
C00020/8	100.00 %		AIN2: (-xmax)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00021

Parameter | Name: **C00021 | Slip comp.**			Data type: INTEGER_16 Index: 24554_d = $5FEA_h$
Slip compensation for V/f characteristic control (VFCplus) and sensorless vector control (SLVC) • A higher slip compensation results in a higher increase in frequency and voltage when the machine is under load. • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well. ▸ Motor control (MCTRL): Optimising the operational performance by slip compensation			
Setting range (min. value | unit | max. value)			**Lenze setting**
-100.00	%	100.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00022

Parameter | Name: **C00022 | Imax in motor mode**			Data type: UNSIGNED_16 Index: 24553_d = $5FE9_h$
Maximum current in motor mode for all motor control modes **Note:** The parameter has a value range depending on the device power. The device automatically limits the entered value to the corresponding value range.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	A	655.35	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00023

Parameter | Name: **C00023 | Imax in generator mode**			Data type: INTEGER_16 Index: 24552_d = $5FE8_h$
Maximum current in generator mode for all motor control modes • 100 % ≡ Imax in motor mode (C00022)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**100.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00024

Parameter | Name: **C00024 | LS_DriveInterface: bNActCompare**			Data type: INTEGER_16 Index: 24551_d = $5FE7_h$
Threshold for the actual speed comparison • This parameter serves to set a threshold that is compared with the actual speed value. • If the value falls below this threshold, the *bNactCompare* output sets the LS_DriveInterface system block to TRUE. • Switching hysteresis = +1 % CAUTION: There are errors in the evaluation if the sum of 199.98 % is exceeded in case of the set value combinations.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	199.99	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00025

Parameter | Name: **C00025 | LS_DriveInterface: bNActHysterese**			Data type: INTEGER_16 Index: 24550_d = $5FE6_h$
From version 16.00.00 Selection of the switching hysteresis for the actual speed value comparison in C00024. CAUTION: There are errors in the evaluation if the sum of 199.98 % is exceeded in case of the set value combinations.			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00025/1	1.00 %		LS_DriveInterface: bNActHysterese
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00026

Parameter | Name: **C00026 | AINx: Offset**			Data type: INTEGER_16 Index: 24549_d = $5FE5_h$
Offset for analog inputs ▶ Analog terminals			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00026/1	0.00 %		AIN1: Offset
C00026/2	0.00 %		AIN2: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00027

Parameter | Name: **C00027 | AINx: Gain**			Data type: INTEGER_32 Index: 24548_d = $5FE4_h$
Gain for analog inputs ▶ Analog terminals			
Setting range (min. value | unit | max. value)			
-100.0000		100.0000	
Subcodes	**Lenze setting**		**Info**
C00027/1	1.0000		AIN1: Gain
C00027/2	1.0000		AIN2: Gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C00028

Parameter | Name: **C00028 | AINx: Input voltage**			Data type: INTEGER_16 Index: 24547_d = $5FE3_h$
Display of the input voltage at the analog inputs ▶ Analog terminals			
Display range (min. value | unit | max. value)			
-10.00	V	10.00	
Subcodes			**Info**
C00028/1			AIN1: Input voltage
C00028/2			AIN2: Input voltage
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00029

Parameter | Name: **C00029 | AINx: Input current**			Data type: INTEGER_16 Index: 24546_d = $5FE2_h$
Display of the input current at the analog inputs • When the corresponding analog input is configured for current measurement (C00034/x = 1 or 2). • When C00034/x is set = 2 (4 ... 20 mA), 0 ... 16 mA is displayed. ▸ Analog terminals			
Display range (min. value | unit | max. value)			
0.00	mA	20.00	
Subcodes			**Info**
C00029/1			AIN1: Input current
C00029/2			AIN2: Input current
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00030

Parameter | Name: **C00030 | LS_DFOut: Const.**			Data type: INTEGER_16 Index: 24545_d = $5FE1_h$
From version 12.00.00 Settings for the digital frequency output (multi-encoder interface X8) ▸ Digital frequency coupling			
Setting range (min. value | unit | max. value)			
16		16384	
Subcodes	**Lenze setting**		**Information**
C00030/1	2048		LS_DFOut: number of increments • The number of increments determines after how many output increments a zero pulse will be generated. Each zero pulse defines a covered "revolution" of the rotary transducer simulated by the digital frequency output.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00033

Parameter | Name: **C00033 | AINx: Output value**			Data type: INTEGER_16 Index: 24542_d = $5FDE_h$
Display of the output value in percent of the analog input amplifier • 100 % ≡ 16384 ≡ +10 V / +20 mA ▸ Analog terminals			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00033/1			AIN1: Output value
C00033/2			AIN2: Output value
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00034

Parameter | Name: **C00034 | AINx: Configuration**		Data type: UNSIGNED_8 Index: 24541_d = $5FDD_h$
Configuration of the analog inputs for current or voltage measurement ▸ Analog terminals		
Selection list		**Info**
0	-10V..+10V	Input signal is the voltage signal -10 V ... +10 V • -10 V ... +10 V ≡ -100 % ... +100 %
1	0mA..20mA	Input signal is the current signal 0 mA ... 20 mA • 0 mA ... 20 mA ≡ 0 % ... +100 %
2	4mA..20mA	Input signal is the current signal 4 mA ... 20 mA • 4 mA ... 20 mA ≡ 0 % ... +100 % • The current loop is monitored for open circuit (I < 4 mA) by the device.
Subcodes	**Lenze setting**	**Info**
C00034/1	0: -10V..+10V	AIN1: Config.
C00034/2	0: -10V..+10V	AIN2: Config.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00036

Parameter | Name: **C00036 | DC braking: Current**			Data type: INTEGER_16 Index: 24539_d = $5FDB_h$
Braking current in [%] based on rated device current (C00098) • 100% ≡ C00098 ▸ DC-injection braking			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	200.00	**50.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

_ _

C00039

Parameter | Name: **C00039 | Fixed setpoint x (L_NSet_1 n-Fix)**			Data type: INTEGER_16 Index: 24536_d = $5FD8_h$
The L_NSet_1 FB: Fixed speed setpoints (JOG values) for the setpoint generator • 100% ≡ C00011			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00039/1	40.00 %		Preset setpoint 1
C00039/2	60.00 %		Preset setpoint 2
C00039/3	80.00 %		Preset setpoint 3
C00039/4	0.00 %		Fixed setpoint 4
C00039/5	0.00 %		Fixed setpoint 5
C00039/6	0.00 %		Fixed setpoint 6
C00039/7	0.00 %		Fixed setpoint 7
C00039/8	0.00 %		Fixed setpoint 8
C00039/9	0.00 %		Fixed setpoint 9
C00039/10	0.00 %		Fixed setpoint 10
C00039/11	0.00 %		Fixed setpoint 11
C00039/12	0.00 %		Fixed setpoint 12
C00039/13	0.00 %		Fixed setpoint 13
C00039/14	0.00 %		Fixed setpoint 14
C00039/15	0.00 %		Fixed setpoint 15
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00050

Parameter | Name: **C00050 | MCTRL: Speed setpoint**			Data type: INTEGER_32 Index: 24525_d = $5FCD_h$
Display of the speed setpoint at the speed setpoint input of the motor control			
Display range (min. value | unit | max. value)			
-120000	rpm	120000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00051

Parameter | Name: **C00051 | MCTRL: Actual speed value**			Data type: INTEGER_32 Index: 24524_d = $5FCC_h$
Display of the actual speed value of the motor shaft **Note:** The displayed value only corresponds to the real actual speed value of the motor shaft if an encoder is connected to the motor and the evaluation of the feedback signal has been set correctly ("Closed loop" operation). In case of operation without speed feedback, the signal is calculated from the motor control and thus may not correspond to the real actual speed.			
Display range (min. value | unit | max. value)			
-120000	rpm	120000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00052

Parameter | Name: **C00052 | Motor voltage**			Data type: UNSIGNED_16 Index: 24523_d = $5FCB_h$
Display of the current motor voltage/output voltage of the inverter			
Display range (min. value | unit | max. value)			
0	V	1000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00053

Parameter | Name: **C00053 | DC-bus voltage**			Data type: UNSIGNED_16 Index: 24522_d = $5FCA_h$
Display of the current DC-bus voltage			
Display range (min. value | unit | max. value)			
0	V	1000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00054

Parameter | Name: **C00054 | Motor current**			Data type: UNSIGNED_16 Index: 24521_d = $5FC9_h$
Display of the current motor current/output current of the inverter			
Display range (min. value | unit | max. value)			
0.00	A	300.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00055

Parameter | Name: **C00055 | Actual values**			Data type: INTEGER_16 Index: 24520_d = $5FC8_h$
Actual value of the encoder/feedback system **Note:** When a single-track feedback has been selected for the HTL encoder (C00115 = 1 or 3), the sign of the actual speed value is created from the sign of the speed setpoint. In C00055/1 und C00055/2, a positive speed is always displayed. ▶ Encoder/feedback system			
Display range (min. value | unit | max. value)			
-32767	rpm	32767	
Subcodes			**Info**
C00055/1			Actual value - HTL encoder FreqIn12
C00055/2			Actual value - HTL encoder FreqIn67
C00055/3			Actual value - multi-encoder
C00055/4			Actual value - resolver
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00056

Parameter | Name: **C00056 | Torque**			Data type: INTEGER_32 Index: 24519_d = $5FC7_h$
Display of the current torque			
Display range (min. value | unit | max. value)			
-65000.00	Nm	65000.00	
Subcodes			**Info**
C00056/1			Torque demand • Only in case of sensorless vector control (SLVC) and servo control (SC).
C00056/2			Actual torque value • Estimated actual torque for all motor control modes.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00057

Parameter | Name: **C00057 | Maximum torque**			Data type: UNSIGNED_32 Index: 24518_d = $5FC6_h$
Display of the maximum torque to be generated by the motor • The maximum torque to be generated by the motor depends on various factors, e.g. on Imax in motor mode (C00022) and the motor type used.			
Display range (min. value | unit | max. value)			
0.00	Nm	65000.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00058

Parameter | Name: **C00058 | Output frequency**			Data type: INTEGER_32 Index: 24517_d = $5FC5_h$
Display of the current output frequency			
Display range (min. value | unit | max. value)			
-1300.00	Hz	1300.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00059

Parameter | Name: **C00059 | Appl.: Reference frequency C11**			Data type: UNSIGNED_32 Index: 24516_d = $5FC4_h$
Display of the field frequency which corresponds to the reference speed set in C00011.			
Display range (min. value | unit | max. value)			
0.00	Hz	1300.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00060

Parameter | Name: **C00060 | Motor rotor position**			Data type: UNSIGNED_16 Index: 24515_d = $5FC3_h$
Display range (min. value | unit | max. value)			
0		2047	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00061

Parameter | Name: **C00061 | Heatsink temperature**			Data type: INTEGER_16 Index: 24514_d = $5FC2_h$
Display of the current heatsink temperature			
Display range (min. value | unit | max. value)			
-50	°C	150	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00062

Parameter | Name: **C00062 | Temperature inside the controller**			Data type: INTEGER_16 Index: 24513_d = $5FC1_h$
From version 02.00.00 Display of the current temperature inside the controller			
Display range (min. value | unit | max. value)			
-200	°C	200	
Subcodes			**Info**
C00062/1			Interior temperature of CU
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00063

Parameter | Name: **C00063 | Motor temperature**			Data type: INTEGER_16 Index: 24512_d = $5FC0_h$
Display of the current motor temperature			
Display range (min. value | unit | max. value)			
-200	°C	250	
Subcodes			**Information**
C00063/1			Motor temperature
C00063/2			Motor temperature via resolver
C00063/3			Motor temperature via MultiEncoder
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00064

Parameter | Name: **C00064 | Device utilisation (Ixt)**			Data type: INTEGER_16 Index: 24511_d = $5FBF_h$
Display of the device utilisation Ixt in different time resolutions • If the value displayed here exceeds the threshold set in C00123, the fault message "OC5: Device overload (Ixt)" is output and the fault response set in C00604 is executed (default setting: "Warning").			
Display range (min. value | unit | max. value)			
0.00	%	250.00	
Subcodes			**Info**
C00064/1			Device utilisation (Ixt) • Maximum value of the pulse utilisation (C00064/2) and permanent utilisation (C00064/3).
C00064/2			Device utilisation (Ixt) 15s • Pulse utilisation over the last 15 seconds (only for loads >160 %).
C00064/3			Device utilisation (Ixt) 3 min • Permanent utilisation over the last 3 minutes.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00065

Parameter | Name: **C00065 | Supply voltage 24V**			Data type: INTEGER_16 Index: 24510_d = $5FBE_h$
Display of the 24V supply voltage for the supply of the control electronics **Note:** The 24 V supply for the control electronics is either provided by an external supply or by the invter itself if it is connected to the mains voltage.			
Display range (min. value | unit | max. value)			
0.0	V	3276.7	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C00066

Parameter | Name: **C00066 | Thermal motor load (I²xt)**			Data type: INTEGER_16 Index: 24509_d = $5FBD_h$
Display of the thermal motor load, sensorlessly determined using a motor model • If the value displayed here exceeds "100.00 %", the error message "OC6: Thermal motor overload (I2xt)" is output and the fault response set in C00606 is executed (default setting: "Warning"). ▸ Motor overload monitoring (I2xt)			
Display range (min. value | unit | max. value)			
0.00	%	199.99	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00070

Parameter | Name: **C00070 | Vp speed controller**			Data type: UNSIGNED_16 Index: 24505_d = $5FB9_h$
Gain factor Vp of the speed controller for different motor control types			
Setting range (min. value | unit | max. value)			
0.00		600.00	
Subcodes	**Lenze setting**		**Info**
C00070/1	power-related (see table)		SLVC: Vp speed controller
C00070/2	power-related (see table)		SC: Vp speed controller • 0: The reset time is inactive.
C00070/3	power-related (see table)		SLPSM: Vp speed controller • 0: The reset time is inactive.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00071

Parameter | Name: **C00071 | Ti speed controller**			Data type: UNSIGNED_16 Index: 24504_d = $5FB8_h$
Reset time Ti of the speed controller for different motor control types			
Setting range (min. value | unit | max. value)			
0.0	ms	6000.0	
Subcodes	**Lenze setting**		**Info**
C00071/1	100.0 ms		SLVC: Ti speed controller
C00071/2	50.0 ms		SC: Ti speed controller
C00071/3	50.0 ms		SLPSM: Ti speed controller
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00072

Parameter | Name: **C00072 | SC: Tdn speed controller**			Data type: UNSIGNED_16 Index: 24503_d = $5FB7_h$
Differential time constant Tdn of the speed controller for servo control (SC) and sensorless control for synchronous motors (SLPSM)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	ms	3.00	**0.00 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00073

Parameter | Name: **C00073 | Imax/M controller gain**			Data type: UNSIGNED_16 Index: 24502_d = $5FB6_h$
Amplification factor Vp of certain controllers for different motor control modes			
Setting range (min. value | unit | max. value)			
0.00		100.00	
Subcodes	**Lenze setting**		**Info**
C00073/1	power-related (see table)		VFC: Vp Imax controller • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.
C00073/2	power-related (see table)		SLVC: Vp torque controller
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00074

Parameter | Name: **C00074 | Reset time Imax/M controller**			Data type: UNSIGNED_16 Index: 24501_d = $5FB5_h$
Reset time Ti of certain controllers for different motor control modes			
Setting range (min. value | unit | max. value)			
0	ms	9990	
Subcodes	**Lenze setting**		**Info**
C00074/1	power-related (see table)		VFC: Ti Imax controller
C00074/2	power-related (see table)		SLVC: Ti torque controller
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00075

Parameter | Name: **C00075 | Vp current controller**			Data type: UNSIGNED_16 Index: 24500_d = $5FB4_h$
Gain factor Vp of the current controller for servo control (SC) and certain inverter functions (parameter identification, flying restart circuit) • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	V/A	500.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00076

Parameter | Name: **C00076 | Ti current controller**			Data type: UNSIGNED_16 Index: 24499_d = $5FB3_h$
Reset time Ti of the current controller for servo control (SC) and certain inverter functions (parameter identification, flying restart circuit) • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	ms	500.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00077

Parameter | Name: **C00077 | SC: Vp field controller**			Data type: UNSIGNED_16 Index: 24498_d = $5FB2_h$
Gain factor Vp of the field controller for servo control (SC) • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00		500.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00078

Parameter | Name: **C00078 | SC: Tn field controller**			Data type: UNSIGNED_16 Index: 24497_d = $5FB1_h$
Reset time Tn of the field controller for servo control (SC) • After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	ms	6000.0	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00079

Parameter | Name: **C00079 | SC: Settings**		Data type: UNSIGNED_8 Index: 24496_d = $5FB0_h$
Configuration of different options for servo control (SC) and sensorless control for synchronous motors (SLPSM)		
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C00079/1	0: Off	SC: Current controller - feedforward control • Please observe the notes regarding this function in the chapter "Feedforward control of the current controller"!
C00079/2	1: On	SC: Adapt. VP in the field weakening range • Speed-dependent adaptive field weakening controller.
C00079/3	0: Off	SC: n-ctrl anti-wind-up • "Anti-wind-up" effect of the speed controller in case of an output voltage limitation in the field weakening range.
C00079/4	1: On	Field weakening for synchronous motors • From version 02.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00080

Parameter | Name: **C00080 | Override point of field weakening**			Data type: INTEGER_16 Index: 24495_d = $5FAF_h$
Offset of the override point for field weakening • In the V/f characteristic control mode (VFCplus), the stall protection function or the max. permissible current in the field weakening range can be adapted.			
Setting range (min. value | unit | max. value)			**Lenze setting**
-500	Hz	500	**0 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00081

Parameter | Name: **C00081 | Rated motor power**			Data type: UNSIGNED_16 Index: 24494_d = $5FAE_h$
This value can be obtained from the motor nameplate. After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. **Note:** It is mandatory to give the rated motor power for the sensorless vector control (SLVC) and the servo control (SC).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	kW	500.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00082

Parameter | Name: **C00082 | Motor rotor resistance**			Data type: UNSIGNED_32 Index: 24493_d = $5FAD_h$
After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	mohm	200000	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00083

Parameter | Name: **C00083 | Motor rotor time constant**			Data type: UNSIGNED_16 Index: 24492_d = $5FAC_h$
Display of the rotor time constant of the motor • This value is calculated from the rotor resistance and the rotor inductance (leakage and magnetising inductance).			
Display range (min. value | unit | max. value)			
0	ms	32767	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00084

Parameter | Name: **C00084 | Motor stator resistance**			Data type: UNSIGNED_32 Index: 24491_d = $5FAB_h$
After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	mohm	200000	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00085

Parameter | Name: **C00085 | Motor stator leakage inductance**			Data type: UNSIGNED_16 Index: 24490_d = $5FAA_h$
After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	mH	650.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00087

Parameter | Name: **C00087 | Rated motor speed**			Data type: UNSIGNED_16 Index: 24488_d = $5FA8_h$
This value can be obtained from the motor nameplate. After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. **Note:** It is mandatory to give the rated motor speed for the sensorless vector control (SLVC) and the servo control (SC).			
Setting range (min. value | unit | max. value)			**Lenze setting**
50	rpm	60000	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00088

Parameter | Name: **C00088 | Rated motor current**			Data type: UNSIGNED_16 Index: 24487_d = $5FA7_h$
This value can be obtained from the motor nameplate. After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.20	A	320.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00089

Parameter | Name: **C00089 | Rated motor frequency**			Data type: UNSIGNED_16 Index: 24486_d = $5FA6_h$
This value can be obtained from the motor nameplate. After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. **Note:** It is mandatory to give the rated motor frequency for the sensorless vector control (SLVC) and the servo control (SC).			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	Hz	1000	**50 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00090

Parameter | Name: **C00090 | Rated motor voltage**			Data type: UNSIGNED_16 Index: 24485_d = $5FA5_h$
This value can be obtained from the motor nameplate. After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	V	5000	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00091

Parameter | Name: **C00091 | Motor cosine phi**			Data type: UNSIGNED_8 Index: 24484_d = $5FA4_h$
This value can be obtained from the motor nameplate. After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.20		1.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00092

Parameter | Name: **C00092 | Motor magnetising inductance**			Data type: UNSIGNED_16 Index: 24483_d = $5FA3_h$
After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	mH	6500.0	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00093

Parameter | Name: **C00093 | Power section identification**			Data type: UNSIGNED_16 Index: 24482_d = $5FA2_h$
Display of the identification of the detected power section of the inverter			
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00095

Parameter | Name: **C00095 | Motor magnetising current**			Data type: UNSIGNED_16 Index: 24480_d = $5FA0_h$
After the motor to be used has been selected from the motor catalogue, the suitable value can be entered automatically. An automatic detection via the motor parameter identification is possible as well.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	A	320.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00097

Parameter | Name: **C00097 | Rated motor torque**			Data type: UNSIGNED_32 Index: 24478_d = $5F9E_h$
Display of the rated motor torque • The value displayed here is calculated from different parameters, e.g. the maximum current set in C00022.			
Display range (min. value | unit | max. value)			
0.00	Nm	100000.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00098

Parameter | Name: **C00098 | Device rated current**			Data type: UNSIGNED_16 Index: 24477_d = $5F9D_h$
Display of the rated inverter current which is defined by the integrated power section.			
Display range (min. value | unit | max. value)			
0.0	A	6000.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00099

Parameter | Name: **C00099 | Firmware version**	Data type: VISIBLE_STRING Index: 24476_d = $5F9C_h$
Display of the firmware version of the device as string	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 12	

C00100

Parameter | Name: **C00100 | Firmware version**			Data type: UNSIGNED_8 Index: 24475_d = $5F9B_h$
Display of the firmware version of the device, divided into subsections.			
Display range (min. value | unit | max. value)			
0		99	
Subcodes			**Info**
C00100/1			Firmware version - main version
C00100/2			Firmware version - subversion
C00100/3			Firmware version - release
C00100/4			Firmware version - build
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00101

Parameter | Name: **C00101 | Add. acceleration time x**			Data type: UNSIGNED_32 Index: 24474_d = $5F9A_h$
The L_NSet_1: FB: Additional acceleration times for the main setpoint • The additional acceleration times set here can be selected via the binary inputs *bTI1* ... *bTI8* of the L_NSet_1 FB.			
Setting range (min. value | unit | max. value)			
0.000	s	999.999	
Subcodes	**Lenze setting**		**Info**
C00101/1	0.000 s		Add. accel. time 1 ... 15
C00101/...			
C00101/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00103

Parameter | Name: **C00103 | Add. decel. time x**			Data type: UNSIGNED_32 Index: 24472_d = $5F98_h$
The L_NSet_1 FB: Additional deceleration times for the main setpoint • The additional deceleration times set here can be selected via the binary inputs *bTI1* ... *bTI8* of the L_NSet_1 FB.			
Setting range (min. value | unit | max. value)			
0.000	s	999.999	
Subcodes	**Lenze setting**		**Info**
C00103/1	0.000 s		Add. decel. time 1 ... 15
C00103/...			
C00103/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00104

Parameter | Name: **C00104 | Quick stop setting**			Data type: UNSIGNED_16 Index: 24471_d = $5F97_h$
From version 02.00.00			▶ Quick stop: Optional settings
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Standstill position-controlled		1 ≡ At standstill the position control is activated. The basic conditions are shown in the flow diagram for the quick stop function.
Bit 1	Ramp position-controlled		1 ≡ During the ramp-down phase, the position control is activated. The basic conditions are shown in the flow diagram for the quick stop function.
Bit 2	Continue following error		From version 15.00.00 1 ≡ The current following error is continued if the quick stop with position control is activated while ramping down.
Bit 3	Starting value is the setpoint speed		From version 15.00.00 0 ≡ The starting value at quick stop is the actual speed if it derives more than 5 rpm from the setpoint. This serves to prevent a jump to a strongly differing setpoint. 1 ≡ The starting value for quick stop is the setpoint speed .
Bit 4	Use position encoder		From version 15.00.00 1 ≡ In case of quick stop with position control, the position encoder is used instead of the speed encoder. The basic conditions are shown in the flow diagram for the quick stop function.

Bit 5	Standstill depending on the actual speed		From version 16.00.00
			1 ≡ Extended option for quick stop with position control at standstill.
			When this setting is activated, the standstill of the motor shaft is detected under consideration of the speed setpoint and the actual speed. The standstill of the motor shaft is reached when
			• Bit 5 = FALSE (Lenze setting)
			  • Speed setpoint = 0
			• Bit 5 = TRUE
			  • Speed setpoint = 0 AND 0 ≤ actual speed value ≤ 1 rpm

Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C00104/1	0x0000		Quick stop setting
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00105

Parameter | Name: **C00105 | Decel. time - quick stop**			Data type: UNSIGNED_32 Index: 24470_d = $5F96_h$
The set deceleration time determines the ramp slope at quick stop ▶ Activate/deactivate quick stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.900	**2.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00106

Parameter | Name: **C00106 | Auto DCB: Hold time**			Data type: UNSIGNED_32 Index: 24469_d = $5F95_h$
Hold time of the automatic DC injection brake • The DC injection brake is applied for the time set here if the value falls below the speed setpoint set in C00019. ▶ DC-injection braking			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.000	**0.500 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00107

Parameter | Name: **C00107 | DC braking: Hold time**			Data type: UNSIGNED_32 Index: 24468_d = $5F94_h$
Maximum hold time of the manual DC injection brake • A time can be set here after which the DC injection brake is switched off automatically to prevent the motor from thermal overload. • When "999.000 s" is set, the hold time is infinite. ▶ DC-injection braking			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.000	**999.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

_ _

C00114

Parameter | Name: **C00114 | DigInX: Inversion**			Data type: UNSIGNED_16 Index: 24461_d = $5F8D_h$
The polarity of each digital input of the device can be inverted via this bit field. ▶ Digital input terminals			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0000** (decimal: 0)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	DI1 inverted		Inversion of digital input 1
Bit 1 ☐	DI2 inverted		Inversion of digital input 2
Bit 2 ☐	DI3 inverted		Inversion of digital input 3
Bit 3 ☐	DI4 inverted		Inversion of digital input 4
Bit 4 ☐	DI5 inverted		Inversion of digital input 5
Bit 5 ☐	DI6 inverted		Inversion of digital input 6
Bit 6 ☐	DI7 inverted		Inversion of digital input 7
Bit 7 ☐	Reserved		
Bit 8 ☐	Reserved		
Bit 9 ☐	Reserved		
Bit 10 ☐	Reserved		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	CInh inverted		Inversion of digital input RFR (controller enable)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00115

Parameter | Name: **C00115 | DI 1/2 & 6/7: Fct.**		Data type: UNSIGNED_8 Index: 24460_d = $5F8C_h$
Function assignment of the digital terminals DI1/2 and DI6/7 ▶ Digital input terminals: Function assignment		
Selection list		**Info**
0	DI1(6)=In / DI2(7)=In	DI1/6 = digital input DI2/7 = digital input
1	DI1(6)=FreqIn / DI2(7)=In	DI1/6 = 1-track frequency input DI2/7 = digital input
2	DI1(6)&DI2(7)=FreqIn (2-track)	DI1/6 and DI2/7 = 2-track frequency input
3	DI1(6)=FreqIn / DI2(7)=direction	DI1/6 = 1-track frequency input DI2/7 = specification of direction
4	DI1(6)=CountIn / DI2(7)=In	DI1/6 = counter input DI2/7 = digital input
Subcodes	**Lenze setting**	**Info**
C00115/1	0: DI1(6)=In / DI2(7)=In	Function assignment of DI1 and DI2
C00115/2	0: DI1(6)=In / DI2(7)=In	Function assignment of DI6 and DI7
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00117

Parameter | Name: **C00117 | Status of brake output BD**		Data type: UNSIGNED_8 Index: 24458_d = $5F8A_h$
Status message of brake output		
Selection list (read only)		**Info**
0	inactive	Brake output is inactive
1	Active	Brake output is active
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00118

Parameter | Name: **C00118 | DigOutX: Inversion**			Data type: UNSIGNED_8 Index: 24457_d = $5F89_h$
The polarity of each digital output of the device can be inverted via this bit field. ▸ Digital output terminals			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x00** (decimal: 0)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	Relay inverted		Relay inversion
Bit 1 ☐	DO1 inverted		Inversion of digital output 1
Bit 2 ☐	DO2 inverted		Inversion of digital output 2
Bit 3 ☐	DO3 inverted		Inversion of digital output 3
Bit 4 ☐	High current inverted		Inversion of high-current output
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00120

Parameter | Name: **C00120 | Setting of motor overload (I²xt)**			Data type: INTEGER_16 Index: 24455_d = $5F87_h$
Operating threshold for the "OC6: Motor overload (I²xt)" error message • The response for reaching the threshold can be selected in C00606. • The current thermal motor load is displayed in C00066. ▸ Motor overload monitoring (I2xt)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	250.00	**100.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00121

Parameter | Name: C00121 | Warning threshold motor temperature			Data type: UNSIGNED_16 Index: 24454_d = $5F86_h$
Operating threshold for error message "Motor: Temperature at Res./Enc. > C121" • The response for reaching the threshold can be selected in C00583. • The current motor temperature is displayed in C00063. ▸ Encoder/feedback system: Motor temperature monitoring (KTY)			
Setting range (min. value | unit | max. value)			
0	°C	250	
Subcodes	**Lenze setting**		**Information**
C00121/1	120 °C		Warning threshold motor temperature resolver
C00121/2	120 °C		Warning threshold motor temperature MultiEncoder
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00122

Parameter | Name: C00122 | Starting value — Overload			Data type: UNSIGNED_16 Index: 24453_d = $5F85_h$
From version 12.00.00 The thermal motor load displayed in C00066 is pre-initialised with the value set here when the device is connected to the mains. • If "100.00 %" is set, the last value at switching off the device is used for the initialisation. • Recommended setting for operation according to UL: 30.00 % ▸ Motor overload monitoring (I2xt)			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C00122/1	0.00 %		Initial value motor overload (I²xt)
C00122/2	0.00 %		Starting value — Brake resistance overload
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00123

Parameter | Name: C00123 | Device utilisation threshold (Ixt)			Data type: INTEGER_16 Index: 24452_d = $5F84_h$
Operating threshold for the "OC5: Device overload (Ixt)" error message • The response for reaching the threshold can be selected in C00604. • The current device utilisation is displayed in C00064.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	200.00	**100.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00124

Parameter | Name: **C00124 | Current monitoring: Breaking current**			Data type: UNSIGNED_16 Index: 24451_d = $5F83_h$
From version 16.00.00			Current monitoring overload
Setting range (min. value | unit | max. value)			
0.00	%	200.00	
Subcodes	**Lenze setting**		**Info**
C00124/1	200.00 %		Current monitoring: Breaking current overload
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00129

Parameter | Name: **C00129 | Brake resistance value**			Data type: UNSIGNED_16 Index: 24446_d = $5F7E_h$
Resistance value of the connected brake resistor • The value to be entered can be obtained from the nameplate of the brake resistor. • For every device type, the value is preset to the minimum adapted Lenze brake resistor.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	Ohm	500.0	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00130

Parameter | Name: **C00130 | Rated power - brake resistor**			Data type: UNSIGNED_16 Index: 24445_d = $5F7D_h$
Rated power of the connected brake resistor • The value to be entered can be obtained from the nameplate of the brake resistor.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	W	65535	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00131

Parameter | Name: **C00131 | Rated heat amount of brake resist.**			Data type: UNSIGNED_16 Index: 24444_d = $5F7C_h$
Heat quantity of the brake resistor connected • Lenze brake resistor: The value is preset to the adapted minimum Lenze brake resistance. • Brake resistor from a third party manufacturer: The value to be entered can be obtained from the nameplate of the brake resistor.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	kWs	6553.5	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00133

Parameter | Name: **C00133 | Brake resistor utilisation**			Data type: UNSIGNED_16 Index: 24442_d = $5F7A_h$
Display of the utilisation of the connected brake resistor			
Display range (min. value | unit | max. value)			
0	%	65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00134

Parameter | Name: **C00134 | L_NSet_1: Ramp smoothing**		Data type: UNSIGNED_8 Index: 24441_d = $5F79_h$
The L_NSet_1 FB: Configuration of the ramp rounding for the main setpoint		
Selection list(Lenze setting printed in bold)		**Info**
0	**Off**	Ramp rounding deactivated
1	PT1 behaviour	Ramp rounding with PT1 behaviour • The corresponding S-ramp time must be set in C00182.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00136

Parameter | Name: **C00136 | Communication control words**			Data type: UNSIGNED_16 Index: 24439_d = $5F77_h$
Control words of the communication interfaces			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	SwitchOn		
Bit 1	DisableVoltage		
Bit 2	SetQuickStop		
Bit 3	EnableOperation		
Bit 4	ModeSpecific_1		
Bit 5	ModeSpecific_2		
Bit 6	ModeSpecific_3		
Bit 7	ResetFault		
Bit 8	SetHalt		
Bit 9	Reserved_1		
Bit 10	Reserved_2		
Bit 11	LenzeSpecific_1		
Bit 12	LenzeSpecific_2		
Bit 13	LenzeSpecific_3		
Bit 14	SetFail		
Bit 15	LenzeSpecific_4		
Subcodes			**Info**
C00136/1			MCI control word • Control word of the MCI communication interface (communication module)
C00136/2			CAN control word • Control word of the CAN communication interface (CAN on board)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00137

Parameter | Name: **C00137 | Device status**		Data type: UNSIGNED_16 Index: 24438_d = $5F76_h$
Display of the current device status		
Selection list (read only)		
0	FirmwareUpdate	
1	Init	
2	Ident	
3	ReadyToSwitchON	
4	SwitchedON	
5	OperationEnable	
6	Warning	
7	Trouble	
8	Fault	
9	TroubleQSP	
10	SafeTorqueOff	
11	SystemFail	
12	Reserved_1	
13	Reserved_2	
14	Reserved_3	
15	Reserved_4	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

_ _

C00138

Parameter | Name: **C00138 | Internal control signals**		Data type: UNSIGNED_16 Index: 24437_d = $5F75_h$
Bit coded display of internal control signals of different sources		
Display area (min. hex value | max. hex value)		
0x0000		0xFFFF
Value is bit-coded:		
Bit 0	Reserved	
Bit 1	DisableVoltage	
Bit 2	SetQuickStop	
Bit 3	EnableOperation	
Bit 4	InitFinishedOK	
Bit 5	ModeSpecific_2	
Bit 6	ModeSpecific_3	
Bit 7	ResetFault	
Bit 8	SetHalt	
Bit 9	FirmwareUpdate	
Bit 10	MotorIdent	
Bit 11	SetMessage	
Bit 12	SetIMP	
Bit 13	SetSystemFail	
Bit 14	SetFail	
Bit 15	SetFailQSP	
Subcodes		**Info**
C00138/1		SYS control signals
C00138/2		MCK control signals
C00138/3		FWM control signals
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT		

C00142

Parameter | Name: **C00142 | Auto-start option**			Data type: UNSIGNED_8 Index: 24433_d = $5F71_h$
The **Auto-start option** function describes the starting performance of the inverter after • connection to the mains and/or switching on the external 24-V supply (Bit 0) • Deactivation of "Trouble" or "Fault" (Bit 1 / 2) • Undervoltage (Bit 3) • Loading the Lenze setting (Bit 4) ▸ Auto-start option "inhibit at device on"			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x19** (decimal: 25)
Value is bit-coded: (☑ = bit set)			
Bit 0 ☑	Inhibit at device on		
Bit 1 ☐	Inhibit at trouble		
Bit 2 ☐	Inhibit at fault		
Bit 3 ☑	Inhibit at undervoltage		
Bit 4 ☑	Inhibit at Lenze setting		
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00144

Parameter | Name: **C00144 | Switching freq. reduct. (Temp.)**		Data type: UNSIGNED_8 Index: 24431_d = $5F6F_h$
Activation of the automatic switching frequency reduction if the temperature is too high		
Selection list(Lenze setting printed in bold)		**Info**
0	Off	Automatic switching frequency reduction deactivated
1	**On**	Automatic switching frequency reduction activated
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00148

Parameter | Name: **C00148 | LS_DriveInterface: Error message config.**			Data type: UNSIGNED_16 Index: 24427_d = $5F6B_h$
Selection of the device statuses for which the *bCollectedFail* group error output of SB LS_DriveInterface is to be set to TRUE.			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0030** (decimal: 48)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	SafeTorqueOff		
Bit 1 ☐	ReadyToSwitchOn		
Bit 2 ☐	SwitchedOn		
Bit 3 ☐	TroubleQSP		
Bit 4 ☑	Trouble		
Bit 5 ☑	Fault		
Bit 6 ☐	Warning		
Bit 7 ☐	ImpIsActive		
Bit 8 ☐	CInhIsActive		
Bit 9 ☐	Fail CAN_Management		
Bit 10 ☐	Reserved		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Lock bCollectedFail at TroubleQSP		From version 18.00.00 onwards
Bit 14 ☐	Lock bFail at TroubleQSP		From version 02.00.00 1 ≡ The *bFail* output of the SB LS_DriveInterface is also set in the "TroubleQSP" status. • Advantage: Even in the "TroubleQSP" status, an error occurred before can still be recognised.
Bit 15 ☐	Use 16BitFailNo.		1 ≡ The short 16-bit error number *wStateDetermFailNoShort* is provided at the *wStateDetermFailNoLow* output of the SB LS_DriveInterface. • In this case, the *wStateDetermFailNoHigh* output is "0". • Advantage: The bus transfer of the error numbers is possible via a data word without changing the interconnection of the technology application.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00150

Parameter | Name: **C00150 | Status word**			Data type: UNSIGNED_16 Index: 24425_d = $5F69_h$
Bit coded device status word			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	FreeStatus		Free status bit 0
Bit 1	PowerDisabled		Power switched off
Bit 2	FreeStatus		Free status bit 2
Bit 3	FreeStatus		Free status bit 3
Bit 4	FreeStatus		Free status bit 4
Bit 5	FreeStatus		Free status bit 5
Bit 6	ActSpeedIsZero		Current speed is 0
Bit 7	ControllerInhibit		Controller is inhibited
Bit 8	StatusCodeBit0		Status code bit 0
Bit 9	StatusCodeBit1		Status code bit 1
Bit 10	StatusCodeBit2		Status code bit 2
Bit 11	StatusCodeBit3		Status code bit 3
Bit 12	Warning		Warning
Bit 13	Trouble		Interference
Bit 14	FreeStatus		Free status bit 14
Bit 15	FreeStatus		Free status bit 15
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00155

Parameter | Name: **C00155 | Extended status word**			Data type: UNSIGNED_16 Index: 24420_d = $5F64_h$
Bit coded device status word 2			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Fail		Error
Bit 1	M_max		Maximum torque
Bit 2	I_max		Maximum current
Bit 3	PowerDisabled		Power switched off
Bit 4	Ready		Controller is ready for operation
Bit 5	ControllerInhibit		Controller is inhibited
Bit 6	Trouble		Interference
Bit 7	InitState		Initialisation
Bit 8	CwCcw		CW/CCW rotation
Bit 9	TroubleQSP		Quick stop due to fault is active
Bit 10	SafeTorqueOff		Safe torque off
Bit 11	AplicationRunning		Application is running
Bit 12	AplParSetBit0		Application parameter set - bit 0
Bit 13	AplParSetBit1		Application parameter set - bit 1
Bit 14	quick stop		Quick stop active
Bit 15	Motor parameter identification		Motor parameter identification is active
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00158

Parameter | Name: **C00158 | Cause of controller inhibit**			Data type: UNSIGNED_16 Index: 24417_d = $5F61_h$
Bit coded display of the cause/source of the controller inhibit			
Display area **(min. hex value | max. hex value)**			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Terminal controller enable		
Bit 1	CAN control word		
Bit 2	MCI control word		
Bit 3	SwitchOn		
Bit 4	Application		
Bit 5	Device command		
Bit 6	Error response		
Bit 7	Internal signal		
Bit 8	Reserved		
Bit 9	Energy saving mode		
Bit 10	AutoStartLock		
Bit 11	Motor parameter identification		
Bit 12	Automatic brake operation		
Bit 13	DCB-IMP		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00159

Parameter | Name: **C00159 | Cause of quick stop QSP**			Data type: UNSIGNED_16 Index: 24416_d = $5F60_h$
Bit coded display of the cause/source of the quick stop			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Reserved		
Bit 1	CAN control word		
Bit 2	MCI control word		
Bit 3	Reserved		
Bit 4	Application		
Bit 5	Device command		
Bit 6	Error response		
Bit 7	Internal signal		
Bit 8	Reserved		
Bit 9	Energy saving mode		
Bit 10	Operating system		
Bit 11	Reserved		
Bit 12	MCK		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00160

Parameter | Name: **C00160 | Status determining error (16bit)**			Data type: UNSIGNED_16 Index: 24415_d = $5F5F_h$
Display of the short 16-bit error number of the status determining error ▸ Structure of the 16-bit error number (bit coding)			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C00160/1			Status determining error (16-bit)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00161

Parameter | Name: **C00161 | LS_SetError_x: Error number**			Data type: UNSIGNED_16 Index: 24414_d = $5F5E_h$
Setting of the error number for user error messages			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C00161/1	1		LS_SetError_1: Error no.1
C00161/2	2		LS_SetError_1: Error no.2
C00161/3	3		LS_SetError_1: Error no.3
C00161/4	4		LS_SetError_1: Error no.4
C00161/5	1		LS_SetError_2: Error no.1
C00161/6	2		LS_SetError_2: Error no.2
C00161/7	3		LS_SetError_2: Error no.3
C00161/8	4		LS_SetError_2: Error no.4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00162

Parameter | Name: **C00162 | Error number masked**			Data type: UNSIGNED_32 Index: 24413_d = $5F5D_h$
From version 13.00.00 Display of the 32-bit error number of the status determining error without error type • The error number displayed here only contains the error subject area and error ID (lower 26 bits of the 32-bit error number) ▸ Structure of the 32-bit error number (bit coding)			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C00162/1			Subject area + Id statuserror
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00163

Parameter | Name: **C00163 | Logbook - binary elements**		Data type: UNSIGNED_16 Index: 24412_d = $5F5C_h$
Selection of two binary signals to be logged in the logbook		
Selection list		**Info**
0	No signal	
1	DI1: Input signal	
2	DI2: Input signal	
3	DI3: Input signal	
4	DI4: Input signal	
5	Controller inhibit signal	
6	Digital counter: Comparison bit	
7	CAN1 input bit 0	
8	CAN1 input bit 1	
9	CAN1 input bit 2	
10	CAN1 input bit 3	

Parameter | Name: **C00163 | Logbook - binary elements**		Data type: UNSIGNED_16 Index: 24412_d = $5F5C_h$
11	CAN1 input bit 4	
12	CAN1 input bit 5	
13	CAN1 input bit 6	
14	CAN1 input bit 7	
15	CAN1 input bit 8	
16	CAN1 input bit 9	
17	CAN1 input bit 10	
18	CAN1 input bit 11	
19	CAN1 input bit 12	
20	CAN1 input bit 13	
21	CAN1 input bit 14	
22	CAN1 input bit 15	
23	CAN2 input bit 0	
24	CAN2 input bit 1	
25	CAN2 input bit 2	
26	CAN2 input bit 3	
27	CAN2 input bit 4	
28	CAN2 input bit 5	
29	CAN2 input bit 6	
30	CAN2 input bit 7	
31	CAN2 input bit 8	
32	CAN2 input bit 9	
33	CAN2 input bit 10	
34	CAN2 input bit 11	
35	CAN2 input bit 12	
36	CAN2 input bit 13	
37	CAN2 input bit 14	
38	CAN2 input bit 15	
39	CAN3 input bit 0	
40	CAN3 input bit 1	
41	CAN3 input bit 2	
42	CAN3 input bit 3	
43	CAN3 input bit 4	
44	CAN3 input bit 5	
45	CAN3 input bit 6	
46	CAN3 input bit 7	
47	CAN3 input bit 8	
48	CAN3 input bit 9	
49	CAN3 input bit 10	
50	CAN3 input bit 11	
51	CAN3 input bit 12	
52	CAN3 input bit 13	
53	CAN3 input bit 14	
54	CAN3 input bit 15	

Parameter | Name: **C00163 | Logbook - binary elements**		Data type: UNSIGNED_16 Index: 24412_d = $5F5C_h$
55	MCI word1 input bit0	
56	MCI word1 input bit1	
57	MCI word1 input bit2	
58	MCI word1 input bit3	
59	MCI Word 1 Input bit 4	
60	MCI word1 input bit5	
61	MCI word1 input bit6	
62	MCI word1 input bit7	
63	MCI word1 input bit8	
64	MCI word1 input bit9	
65	MCI word1 input bit10	
66	MCI word1 input bit11	
67	MCI word1 input bit12	
68	MCI word1 input bit13	
69	MCI word1 input bit14	
70	MCI word1 input bit15	
71	MCI word2 input bit0	
72	MCI word2 input bit1	
73	MCI Word 2 Input bit 2	
74	MCI word2 input bit3	
75	MCI word2 input bit4	
76	MCI word2 input bit5	
77	MCI word2 input bit6	
78	MCI word 2 input bit 7	
79	MCI word2 input bit8	
80	MCI word2 input bit9	
81	MCI word2 input bit10	
82	MCI word2 input bit11	
83	MCI Word 2 Input bit 12	
84	MCI word2 input bit13	
85	MCI word2 input bit14	
86	MCI word2 input bit15	
87	Position controller: Limit	
88	Speed controller: Limit	
89	Speed setpoint: Limit	
90	Torque setpoint: Limit	
91	Current setpoint: Limit	
92	DC injection brake active	
93	Quick stop active	
94	Pulse inhibit active	
95	Controller inhibit active	
96	Safe status active	
97	Direction of rotation ccw	
98	Actual speed = 0	

Parameter | Name: **C00163 | Logbook - binary elements**		Data type: UNSIGNED_16 Index: 24412_d = $5F5C_h$
99	L_Or_1: Out	
100	L_DFlipFlop_1: Out	
101	L_DigitalDelay_1: Out	
102	L_Compare_1: Out	
103	L_Compare_2: Out	
104	L_NSet_1: Setpoint reached	
105	L_DigitalLogic_1: Out	
106	L_SignalMonitor_b: Out1	
107	L_SignalMonitor_b: Out2	
108	L_SignalMonitor_b: Out3	
109	L_SignalMonitor_b: Out4	
110	L_PCTRL_1: act=set	
Subcodes	**Lenze setting**	**Info**
C00163/1	0: No signal	Logbook - binary element 1
C00163/2	0: No signal	Logbook - binary element 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00164

Parameter | Name: **C00164 | Logbook - analog elements**		Data type: UNSIGNED_16 Index: 24411_d = $5F5B_h$
Selection of an analog signal to be logged in the logbook		
Selection list		**Info**
0	No signal	
1	AIN1	
2	CAN1 control word	
3	CAN1 input word 2	
4	CAN1 input word 3	
5	CAN1 input word 4	
6	CAN2 input word 1	
7	CAN2 input word 2	
8	CAN2 input word 3	
9	CAN2 input word 4	
10	CAN3 input word 1	
11	CAN3 input word 2	
12	CAN3 input word 3	
13	CAN3 input word 4	
14	Digital counter LowWord	
15	Digital counter HighWord	
16	MCI word 1	
17	MCI word 2	
18	MCI word 3	
19	MCI word 4	
20	MCI word 5	
21	MCI word 6	

Parameter | Name: **C00164 | Logbook - analog elements**		Data type: UNSIGNED_16 Index: 24411_d = $5F5B_h$
22	MCI word 7	
23	MCI word 8	
24	MCI word 9	
25	MCI word 10	
26	MCI word 11	
27	MCI word 12	
28	MCI word 13	
29	MCI word 14	
30	MCI word 15	
31	MCI word 16	
32	Current motor speed	
33	Current motor torque	
34	DC-bus voltage	
35	Current motor current	
36	Current motor voltage	
37	Current motor frequency	
38	Effective speed setpoint	
39	Device utilisation	
40	Motor utilisation	
41	L_OffsetGainPar_1: Out	
42	L_OffsetGainPar_2: Out	
43	L_OffsetGainPar_3: Out	
44	L_Aritmethik_1: Out	
45	L_AnalogSwitch_1: Out	
46	L_NSet_1: Out	
47	L_MotorPoti_1: Out	
48	L_PCTRL_1: Out	
49	L_SignalMonitor_a: Out1	
50	L_SignalMonitor_a: Out2	
51	L_SignalMonitor_a: Out3	
52	L_SignalMonitor_a: Out4	
53	L_MulDiv_1: Out	
54	L_NSet_1: Target setpoint	
Subcodes	**Lenze setting**	**Info**
C00164/1	0: No signal	Logbook - analog element 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00165

Parameter | Name: **C00165 | Error information**	Data type: VISIBLE_STRING Index: 24410_d = $5F5A_h$
Display of the error number divided into sectors in the event of an error	
Subcodes	**Info**
C00165/1	Status determining error
C00165/2	Current error
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 14	

C00166

Parameter | Name: **C00166 | Error information text**	Data type: VISIBLE_STRING Index: 24409_d = $5F59_h$
Display of details on the status determining error and on the currently pending error	
Subcodes	**Info**
C00166/1	Resp. to status det. error • Response to the status determining error
C00166/2	Subj. - status det. error • Subject area of the status determining error
C00166/3	Mess. - status det. error • Textual message of the status determining error
C00166/4	Resp. to curr. error • Response of the currently pending error
C00166/5	Subj. - curr. error • Subject area of the currently pending error
C00166/6	Mess. - curr. error • Textual message of the currently pending error
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 31	

C00167

Parameter | Name: **C00167 | Logbook data**	Data type: OCTET_STRING Index: 24408_d = $5F58_h$
This code is used device-internally and must not be written by the user side!	

C00168

Parameter | Name: **C00168 | Status determining error**			Data type: UNSIGNED_32 Index: 24407_d = $5F57_h$
Display of the 32-bit error number of the status determining error ▸ Structure of the 32-bit error number (bit coding)			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00169

Parameter | Name: **C00169 | Logbook setting**			Data type: UNSIGNED_16 Index: 24406_d = $5F56_h$
Configuration which message types are to be logged in the logbook.			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x067E** (decimal: 1662)
Value is bit-coded: (☑ = bit set)			
Bit 0 ☐	Reserved		
Bit 1 ☑	Log entry: Fault		
Bit 2 ☑	Log entry: Trouble		
Bit 3 ☑	Log entry: TroubleQuickstop		
Bit 4 ☑	Log entry: WarningLocked		
Bit 5 ☑	Log entry: Warning		
Bit 6 ☑	Log entry: Information		
Bit 7 ☐	Reserved		
Bit 8 ☐	Reserved		
Bit 9 ☑	Activation: Error counter		
Bit 10 ☑	Activation: Log line refresh		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00170

Parameter | Name: **C00170 | Current error**			Data type: UNSIGNED_32 Index: 24405_d = $5F55_h$
Display of the internal error number of the currently pending error			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00171

Parameter | Name: **C00171 | Logbook access index**	Data type: UNSIGNED_8 Index: 24404_d = $5F54_h$
This code is used device-internally and must not be written by the user side!	

C00173

Parameter | Name: **C00173 | Mains voltage**		Data type: UNSIGNED_8 Index: 24402_d = $5F52_h$
If the rated mains voltage differs from 230 V or 400 V, set the mains voltage the drive is operated with. The set mains voltage influences the brake chopper threshold, the monitoring of the device utilisation (Ixt) and the switch-off threshold in case of undervoltage in the DC bus. • The brake chopper threshold must not be smaller than the stabilised DC-bus voltage		
Selection list(Lenze setting printed in bold)		**Info**
0	**3ph 400V / 1ph 230V**	3-phase 400 V or 1-phase 230 V
1	3ph 440V / 1ph 230V	3-phase 440 V or 1-phase 230 V
2	3ph 480V / 1ph 230V	3-phase 480 V or 1-phase 230 V
3	3ph 500V / 1ph 230V	3-phase 500 V or 1-phase 230 V
4	Reserved / reserved	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00174

Parameter | Name: **C00174 | Reduc. brake chopper threshold**			Data type: UNSIGNED_8 Index: 24401_d = $5F51_h$
The threshold from which on the brake chopper is controlled is reduced by the voltage value set here. • The brake chopper threshold must not be smaller than the stabilised DC-bus voltage			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	V	150	**0 V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00175

Parameter | Name: **C00175 | Brake energy management**		Data type: UNSIGNED_8 Index: 24400_d = $5F50_h$
Selection of the braking procedure ▸ Select response if the brake resistor is controlled		
Selection list(Lenze setting printed in bold)		**Info**
0	**R_Brems**	The brake resistor is used. When the threshold voltage (C00174) is exceeded, the brake resistor is energised.
1	RfgStop	The "Ramp function generator stop" signal (*MCTRL_bRfgStop*) is used. When the threshold voltage is exceeded (C00174), the ramp function generator is stopped.
2	R_Brems + HlgStop	The brake resistor and the "Ramp function generator stop" signal are used. When the threshold voltage is exceeded (C00174), the brake resistor is energised and the ramp function generator is stopped.
3	FI_MotBrk + RfgStop	Braking is performed by a superimposed speed setpoint vibration in conjunction with "Ramp function generator stop".
4	R_Brems + FU_MotBrk + HlgStop	Braking is performed by combining all three braking procedures.
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00176

Parameter | Name: **C00176 | Undervoltage threshold for mains OFF**	Data type: UNSIGNED_16 Index: 24399_d = $5F4F_h$
This code is used device-internally and must not be written by the user side!	

C00177

Parameter | Name: **C00177 | Switching cycles**			Data type: UNSIGNED_32 Index: 24398_d = $5F4E_h$
Counter of different switching cycles and stressful situations			
Display range (min. value | unit | max. value)			
0		2147483647	
Subcodes			**Info**
C00177/1			Number of mains switching cycles
C00177/2			Number of switching cycles of the output relay
C00177/3			Short circuit counter
C00177/4			Earth fault counter
C00177/5			"Clamp" counter
C00177/6			Counter for "safe torque off" (STO) after power-on • From version 12.00.00
C00177/7			Counter for controller inhibit via terminal after power-on • From version 12.00.00
C00177/8			Counter for pulse inhibit (IMP) after power-on • From version 12.00.00
C00177/9			Service code
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00178

Parameter | Name: **C00178 | Elapsed-hour meter**			Data type: UNSIGNED_32 Index: 24397_d = $5F4D_h$
Display of operating hours in seconds			
Display range (min. value | unit | max. value)			
0	s	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00179

Parameter | Name: **C00179 | Power-on time meter**			Data type: UNSIGNED_32 Index: 24396_d = $5F4C_h$
Display of the power-on time in seconds			
Display range (min. value | unit | max. value)			
0	s	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00180

Parameter | Name: **C00180 | Running time**			Data type: UNSIGNED_32 Index: 24395_d = $5F4B_h$
Display of various running times in seconds			
Display range (min. value | unit | max. value)			
0	s	2147483647	
Subcodes			**Info**
C00180/1			Runtime - control card
C00180/2			Running time - heatsink fan
C00180/3			Running time - internal fan
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00181

Parameter | Name: **C00181 | Time settings**			Data type: UNSIGNED_16 Index: 24394_d = $5F4A_h$
Time for device search function (optical location) ▶ Device search function			
Setting range (min. value | unit | max. value)			
0	s	6000	
Subcodes	**Lenze setting**		**Info**
C00181/1	5 s		Time - device search function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00182

Parameter | Name: **C00182 | L_NSet_1: S-ramp time PT1**			Data type: INTEGER_16 Index: 24393_d = $5F49_h$
FB L_NSet_1: PT1 S-ramp time for the main setpoint ramp function generator • Only effective with activated ramp rounding (C00134 = "1").			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.01	s	50.00	**20.00 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00184

Parameter | Name: **C00184 | AutoFailReset repetition time**			Data type: UNSIGNED_16 Index: 24391_d = $5F47_h$
After the time set here has expired, an error message of an error that has occurred will be reset automatically if "AutoFailReset" had been configured correspondingly in C00188. ▶ AutoFailReset function			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	s	600	**3 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00185

Parameter | Name: **C00185 | AutoFailReset remaining time**			Data type: UNSIGNED_16 Index: 24390_d = $5F46_h$
Display of the residual runtime of the "AutoFailReset" function AutoFailReset function			
Display range (min. value | unit | max. value)			
0	s	600	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00186

Parameter | Name: **C00186 | Max. number of AutoFailReset processes**			Data type: UNSIGNED_8 Index: 24389_d = $5F45_h$
Maximum number of "AutoFailReset" procedures AutoFailReset function			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		16	**4**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00187

Parameter | Name: **C00187 | Current AutoFailReset processes**			Data type: UNSIGNED_8 Index: 24388_d = $5F44_h$
Data of the current number of "AutoFailReset" procedures AutoFailReset function			
Display range (min. value | unit | max. value)			
0		16	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00188

Parameter | Name: **C00188 | AutoFailReset configuration**		Data type: UNSIGNED_8 Index: 24387_d = $5F43_h$
Setting which error messages are to be reset automatically. AutoFailReset function		
Selection list(Lenze setting printed in bold)		**Info**
0	**Off**	No automatic error message reset
1	Fault + TroubleQSP	Error messages with the response "Fault" and "TroubleQSP" are reset automatically
2	WarningLocked	Error messages with the response "WarningLocked" are reset automatically
3	All locking	All "locking" error messages are reset automatically
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00189

Parameter | Name: **C00189 | Resp. to too frequent AutoFailReset**		Data type: UNSIGNED_8 Index: 24386_d = $5F42_h$
Response to exceeding the maximum number of "AutoFailReset" processes set in C00186. ▸ AutoFailReset function		
Selection list(Lenze setting printed in bold)		
0	No Reaction	
1	**Fault**	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00190

Parameter | Name: **C00190 | L_NSet_1: Setpoint arithmetic**		Data type: UNSIGNED_8 Index: 24385_d = $5F41_h$
The L_NSet_1 FB: Selection of arithmetics • To be able to influence the main setpoint (NSet) by an additional setpoint (NAdd).		
Selection list(Lenze setting printed in bold)		
0	**Out = Set**	
1	Out = Set + Add	
2	NOut = NSet - NAdd	
3	NOut = (NSet * NAdd) / 100%	
4	NOut = (NSet * 1%) / |NAdd|	
5	Out = (Set*100%)/(100%-Add)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00191

Parameter | Name: **C00191 | Logbook access index User**			Data type: UNSIGNED_8 Index: 24384_d = $5F40_h$
From version 12.00.00 ▸ Logbook read interface			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C00191/1	255		Logbook access index User
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00192

Parameter | Name: **C00192 | Logbook data User**			Data type: OCTET_STRING Index: 24383_d = $5F3F_h$
From version 12.00.00			▸ Logbook read interface
Display range (min. value | unit | max. value)			
Subcodes			**Info**
C00192/1			Logbook data User
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00193

Parameter | Name: **C00193 | Logbook element User**			Data type: UNSIGNED_32 Index: 24382_d = $5F3E_h$
From version 12.00.00			▸ Logbook read interface
Display range (min. value | unit | max. value)			
0		2147483647	
Subcodes			**Info**
C00193/1			Response index
C00193/2			Activity
C00193/3			Meters
C00193/4			Error type
C00193/5			Error number
C00193/6			Time stamp
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00199

Parameter | Name: **C00199 | Description data**		Data type: VISIBLE_STRING Index: 24376_d = $5F38_h$
Parameters for storing decription data for the inverter		▸ Device identification
Subcodes	**Lenze setting**	**Info**
C00199/1		Device name
C00199/2		User text • From version 12.00.00
C00199/3		User text • From version 12.00.00
C00199/4		User text • From version 12.00.00
C00199/5		User text • From version 12.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ COM ☐ MOT Character length: 32		

C00200

Parameter | Name: **C00200 | Firmware product type**	Data type: VISIBLE_STRING Index: 24375_d = $5F37_h$
Display of the firmware product type	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 19	

C00201

Parameter | Name: **C00201 | Firmware**	Data type: VISIBLE_STRING Index: 24374_d = $5F36_h$
Display of the firmware data of the control card and the power section	
Subcodes	**Info**
C00201/1	Firmware type - ctrl card
C00201/2	Firmware version - ctrl card
C00201/3	Firmware comp. file - ctrl card
C00201/4	Firmware type - power section
C00201/5	Firmware version - power sect.
C00201/6	Firmw. comp. file - power sect.
C00201/7	Firmware type CU2
C00201/8	Firmware version CU2
C00201/9	Firmware comp. dat. CU2
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 22	

C00203

Parameter | Name: **C00203 | Product type code**	Data type: VISIBLE_STRING Index: 24372_d = $5F34_h$
Display of the types of the individual device components	
Subcodes	**Info**
C00203/1	Type: Control card
C00203/2	Type: Power section
C00203/3	Type: MCI module
C00203/4	Reserved
C00203/5	Type: Memory module
C00203/6	Type: Safety card
C00203/7	Type: Standard device
C00203/8	Type: Complete device
C00203/9	Reserved
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 24	

C00204

Parameter | Name: **C00204 | Serial number**	Data type: VISIBLE_STRING Index: 24371_d = $5F33_h$
Display of the serial numbers of the individual device components	
Subcodes	**Info**
C00204/1	Serial no.: Control card
C00204/2	Serial no.: Power section
C00204/3	Serial no.: MCI module
C00204/4	Reserved
C00204/5	Serial no.: Memory module
C00204/6	Serial no.: Safety card
C00204/7	Serial no.: Standard device
C00204/8	Serial no.: Complete device
C00204/9	Reserved
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 24	

C00205

Parameter | Name: **C00205 | Info**	Data type: VISIBLE_STRING Index: 24370_d = $5F32_h$
This code is used device-internally and must not be written by the user side!	

C00206

Parameter | Name: **C00206 | Production date**	Data type: VISIBLE_STRING Index: 24369_d = $5F31_h$
This code is used device-internally and must not be written by the user side!	

C00210

Parameter | Name: **C00210 | HW version**	Data type: VISIBLE_STRING Index: 24365_d = $5F2D_h$
This code is used device-internally and must not be written by the user side!	

C00219

Parameter | Name: **C00219 | Identity**			Data type: UNSIGNED_32 Index: 24356_d = $5F24_h$
From version 12.00.00			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C00219/1			CAN manufacturer no.
C00219/2			CAN device type
C00219/3			CAN version
C00219/4			CAN count no.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00220

Parameter | Name: **C00220 | L_NSet_1: Acceleration time - add. setpoint**			Data type: UNSIGNED_32 Index: 24355_d = $5F23_h$
The L_NSet_1 FB: Acceleration time for the additional setpoint *nNAdd_a*			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**0.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00221

Parameter | Name: **C00221 | L_NSet_1: Deceleration time - add. setpoint**			Data type: UNSIGNED_32 Index: 24354_d = $5F22_h$
The L_NSet_1 FB: Deceleration time for the additional setpoint *nNAdd_a*			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**0.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00222

Parameter | Name: **C00222 | L_PCTRL_1: Vp**			Data type: INTEGER_16 Index: 24353_d = $5F21_h$
The L_PCTRL_1 FB: Gain factor Vp for the PID process controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.1		500.0	**1.0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C00223

Parameter | Name: **C00223 | L_PCTRL_1: Tn**			Data type: UNSIGNED_16 Index: 24352_d = $5F20_h$
The L_PCTRL_1 FB: Reset time Tn for the PID process controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
20	ms	6000	**400 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00224

Parameter | Name: **C00224 | L_PCTRL_1: Kd**			Data type: UNSIGNED_16 Index: 24351_d = $5F1F_h$
The L_PCTRL_1 FB: Derivative-action coefficient Kd for the PID process controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0		5.0	**0.0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C00225

Parameter | Name: **C00225 | L_PCTRL_1: MaxLimit**			Data type: INTEGER_16 Index: 24350_d = $5F1E_h$
The L_PCTRL_1 FB: Maximum output value of the PID process controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**199.99 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00226

Parameter | Name: **C00226 | L_PCTRL_1: MinLimit**			Data type: INTEGER_16 Index: 24349_d = $5F1D_h$
The L_PCTRL_1 FB: Minimum output value of the PID process controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**-199.99 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00227

Parameter | Name: **C00227 | L_PCTRL_1: Acceleration time**			Data type: UNSIGNED_32 Index: 24348_d = $5F1C_h$
The L_PCTRL_1 FB: Acceleration time for the output value of the PID process controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**0.010 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00228

Parameter | Name: **C00228 | L_PCTRL_1: Deceleration time**			Data type: UNSIGNED_32 Index: 24347_d = $5F1B_h$
The L_PCTRL_1 FB: Deceleration time for the output value of the PID process controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**0.010 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00231

Parameter | Name: **C00231 | L_PCTRL_1: Operating range**			Data type: INTEGER_16 Index: 24344_d = $5F18_h$
The L_PCTRL_1 FB: Operating range for the PID process controller			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00231/1	199.99 %		L_PCTRL_1: Pos. maximum
C00231/2	0.00 %		L_PCTRL_1: Pos. minimum
C00231/3	0.00 %		L_PCTRL_1: Neg. minimum
C00231/4	199.99 %		L_PCTRL_1: Neg. maximum
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00233

Parameter | Name: **C00233 | L_PCTRL_1: Root function**			Data type: UNSIGNED_8 Index: 24342_d = $5F16_h$
The L_PCTRL_1 FB: Use of the root function at the actual value input			
Selection list(Lenze setting printed in bold)		**Info**	
0	**Off**	Root function inactive • The actual value *nAct_a* remains unchanged for further processing	
1	On	Root function active • The root is extracted of the actual value *nAct_a* for further processing	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00234

Parameter | Name: **C00234 | Oscillation damping influence**			Data type: UNSIGNED_16 Index: 24341_d = $5F15_h$
From a device power of 2.2 kW: 50 % ▶ Oscillation damping			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	250.00	**5.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00235

Parameter | Name: **C00235 | Oscillation damping filter time**			Data type: UNSIGNED_8 Index: 24340_d = $5F14_h$
▶ Oscillation damping			
Setting range (min. value | unit | max. value)			**Lenze setting**
2	ms	250	**32 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00236

Parameter | Name: **C00236 | Field weakening oscillation damping**			Data type: UNSIGNED_8 Index: 24339_d = $5F13_h$
Oscillation damping for idling machines ▶ Oscillation damping			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		40	**14**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00241

Parameter | Name: **C00241 | L_NSet_1: Hyst. NSet reached**			Data type: INTEGER_16 Index: 24334_d = $5F0E_h$
The L_NSet_1 FB: Hysteresis window for the zero detection of the speed output setpoint • The speed threshold for the zero detection is 1 %			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**0.50 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00242

Parameter | Name: **C00242 | L_PCTRL_1: Operating mode**		Data type: UNSIGNED_8 Index: 24333_d = $5F0D_h$
The L_PCTRL_1 FB: Selection of the operating mode • Depending on the selection, the blue switches in the displayed signal flow are set accordingly in the Engineer on the **Application parameters** tab for the L_PCTRL_1 FB.		
Selection list(Lenze setting printed in bold)		**Info**
0	**Off**	The input setpoint *nNSet_a* is output without any changes at the output *nOut_a*.
1	nNSet + nNSet_PID	*nNSet_a* and *nAct_a* are used as PID input values. The arriving *nNSet_a* is additively linked to the value output by the PID element.
2	nSet_PID	*nSet_a* and *nAct_a* are used as PID input values. The input *nNSet_a* is not considered.
3	nNSet_PID	*nNSet_a* and *nAct_a* are used as PID input values. The input *nSet_a* is not considered.
4	nNSet + nSet_PID	*nSet_a* and *nAct_a* are used as PID input values. The arriving *nNSet_a* setpoint is additively linked to the value output by the PID element.
5	nNSet || nSet_PID	*nSet_a* and *nAct_a* are used as PID input values. The setpoint *nNSet_a* is output at the output *nOut_a* . The PID output value is output at the output *nPIDOut_a*.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00243

Parameter | Name: **C00243 | L_PCTRL_1: Acceleration time influence**			Data type: UNSIGNED_32 Index: 24332_d = $5F0C_h$
The L_PCTRL_1 FB: Acceleration time for showing the PID output value			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**5.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00244

Parameter | Name: **C00244 | L_PCTRL_1: Deceleration time influence**			Data type: UNSIGNED_32 Index: 24331_d = $5F0B_h$
The L_PCTRL_1 FB: Deceleration time for masking out the PID output value			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**5.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00245

Parameter | Name: **C00245 | L_PCTRL_1: PID output value**			Data type: INTEGER_16 Index: 24330_d = $5F0A_h$
The L_PCTRL_1 FB: Display of the output value of the PID process controller			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00246

Parameter | Name: **C00246 | L_PCTRL_1: nAct_a internal**			Data type: INTEGER_16 Index: 24329$_d$ = 5F09$_h$
FB L_PCTRL_1: Display of the internal actual value			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00247

Parameter | Name: **C00247 | L_PCTRL_1: Window setpoint reached**			Data type: INTEGER_16 Index: 24328$_d$ = 5F08$_h$
FB L_PCTRL_1: Window for comparison operation "actual value = setpoint"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**2.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00249

Parameter | Name: **C00249 | L_PT1_1: Time constant**			Data type: UNSIGNED_16 Index: 24326$_d$ = 5F06$_h$
FB L_PT1_1: Time constant Tn			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	5000	**2000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00250

Parameter | Name: **C00250 | L_PT1 2-3: Time constant**			Data type: INTEGER_16 Index: 24325$_d$ = 5F05$_h$
Setting range (min. value | unit | max. value)			
0	ms	5000	
Subcodes	**Lenze setting**		**Info**
C00250/1	2000 ms		L_PT1_2: Time constant
C00250/2	2000 ms		L_PT1_3: Time constant
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00251

Parameter | Name: **C00251 | L_DT1_1: Time constant**			Data type: INTEGER_16 Index: 24324$_d$ = 5F04$_h$
FB L_DT1_1: Time constant Tn			
Setting range (min. value | unit | max. value)			**Lenze setting**
10	ms	5000	**1000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00252

Parameter | Name: **C00252 | L_DT1_1: Gain**			Data type: INTEGER_16 Index: 24323$_d$ = 5F03$_h$
FB L_DT1_1: Gain factor Vp			
Setting range (min. value | unit | max. value)			**Lenze setting**
-320.00		320.00	**1.00**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00253

Parameter | Name: **C00253 | L_DT1_1: Sensitivity**		Data type: UNSIGNED_8 Index: 24322$_d$ = 5F02$_h$
FB L_DT1_1: Selection of sensitivity • Depending on the selection, the number of indicated higher-order bits is evaluated. **Note:** The most significant bit determines the sign of the value, the remaining bits determine the numerical value.		
Selection list(Lenze setting printed in bold)		**Info**
1	**15 bits**	Bit 0 ... bit 14 are evaluated
2	14 Bit	Bit 0 ... bit 13 are evaluated
3	13 bits	Bit 0 ... bit 12 are evaluated
4	12 bits	Bit 0 ... bit 11 are evaluated
5	11 Bit	Bit 0 ... bit 10 are evaluated
6	10 Bit	Bit 0 ... bit 9 are evaluated
7	9 Bit	Bit 0 ... bit 8 are evaluated
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00254

Parameter | Name: **C00254 | Kp position controller**			Data type: UNSIGNED_16 Index: 24321$_d$ = 5F01$_h$
Gain for following error compensation			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	1/s	500.00	**5.00 1/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00265

Parameter | Name: **C00265 | SLVC: Filtering of setpoint and actual currents**	Data type: UNSIGNED_8 Index: 24310$_d$ = 5EF6$_h$
This code is used device-internally and must not be written by the user side!	

C00270

Parameter | Name: **C00270 | SC: Freq. current setpoint filter**			Data type: UNSIGNED_16 Index: 24305$_d$ = 5EF1$_h$
Frequency to be inhibited by the current setpoint filter at servo control (SC) and sensorless control for synchronous motors (SLPSM).			
Setting range (min. value | unit | max. value)			**Lenze setting**
40.0	Hz	1000.0	**200.0 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00271

Parameter | Name: **C00271 | SC: Width of current setpoint filter**			Data type: UNSIGNED_16 Index: 24304_d = $5EF0_h$
Frequency width of the current setpoint filter at servo control (SC) and sensorless control for synchronous motors (SLPSM) • Width around the frequency to be inhibited (C00270).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	Hz	500.0	**0.0 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00272

Parameter | Name: **C00272 | SC: Depth of current setpoint filter**			Data type: UNSIGNED_16 Index: 24303_d = $5EEF_h$
Damping of the current setpoint filter at servo control (SC) and sensorless control for synchronous motors (SLPSM)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	dB	100	**0 db**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00273

Parameter | Name: **C00273 | Moment of inertia motor**			Data type: UNSIGNED_32 Index: 24302_d = $5EEE_h$
Moment of inertia for setpoint feedforward control at servo control (SC) and sensorless vector control (SLVC)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	kg cm^2	6000000.00	**0.00 kg cm^2**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00274

Parameter | Name: **C00274 | SC: Max. change in acceleration**			Data type: UNSIGNED_16 Index: 24301_d = $5EED_h$
Limitation of the acceleration change at servo control (SC) and sensorless control for synchronous motors (SLPSM) • Setting in % of M_Nenn per ms.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	%/ms	400.0	**400.0 %/ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00275

Parameter | Name: **C00275 | Filter setpoint feedforward control**			Data type: UNSIGNED_16 Index: 24300_d = $5EEC_h$
Filter time of setpoint feedforward control at servo control (SC) and sensorless vector control (SLVC) • The setpoint feedforward control requires the entry of the moment of inertia in C00273.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	ms	1000.0	**1.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00276

Parameter | Name: **C00276 | SC: Max. output voltage**			Data type: UNSIGNED_8 Index: 24299_d = $5EEB_h$
Maximum output voltage at servo control (SC) • Regarding the current DC-bus voltage.			
Setting range (min. value | unit | max. value)			**Lenze setting**
80	%	99	**95 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00280

Parameter | Name: **C00280 | SC: Filter time const. DC detection**			Data type: UNSIGNED_16 Index: 24295_d = $5EE7_h$
Filter time constant for DC-bus voltage filtering • The filter time constant is e.g. used for field weakening control at servo control (SC).			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	ms	1000	**25 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00290

Parameter | Name: **C00290 | RCOM error counter**	Data type: UNSIGNED_16 Index: 24285_d = $5EDD_h$
This code is used device-internally and must not be written by the user side!	

C00291

Parameter | Name: **C00291 | Error type RCOM**	Data type: UNSIGNED_8 Index: 24284_d = $5EDC_h$
This code is used device-internally and must not be written by the user side!	

C00295

Parameter | Name: **C00295 | savezycle memory modul**	Data type: INTEGER_32 Index: 24280_d = $5ED8_h$
This code is used device-internally and must not be written by the user side!	

C00296

Parameter | Name: **C00296 | ICOM error number**	Data type: UNSIGNED_16 Index: 24279_d = $5ED7_h$
This code is used device-internally and must not be written by the user side!	

C00297

Parameter | Name: **C00297 | Counter Receive Error Isr**	Data type: UNSIGNED_8 Index: 24278_d = $5ED6_h$
This code is used device-internally and must not be written by the user side!	

C00301

Parameter | Name: **C00301 | DebugAccess**	Data type: UNSIGNED_16 Index: 24274_d = $5ED2_h$
This code is used device-internally and must not be written by the user side!	

C00302

Parameter | Name: **C00302 | Internal Commands**	Data type: UNSIGNED_8 Index: 24273_d = $5ED1_h$
This code is used device-internally and must not be written by the user side!	

C00304

Parameter | Name: **C00304 | Password1**	Data type: UNSIGNED_32 Index: 24271_d = $5ECF_h$
This code is used device-internally and must not be written by the user side!	

C00305

Parameter | Name: **C00305 | Password2**	Data type: UNSIGNED_32 Index: 24270_d = $5ECE_h$
This code is used device-internally and must not be written by the user side!	

C00306

Parameter | Name: **C00306 | Debug address**	Data type: UNSIGNED_32 Index: 24269_d = $5ECD_h$
This code is used device-internally and must not be written by the user side!	

C00307

Parameter | Name: **C00307 | Debug value**	Data type: UNSIGNED_16 Index: 24268_d = $5ECC_h$
This code is used device-internally and must not be written by the user side!	

C00308

Parameter | Name: **C00308 | PartitionOffset**	Data type: UNSIGNED_16 Index: 24267_d = $5ECB_h$
This code is used device-internally and must not be written by the user side!	

C00309

Parameter | Name: **C00309 | PartitionSel**	Data type: UNSIGNED_8 Index: 24266_d = $5ECA_h$
This code is used device-internally and must not be written by the user side!	

C00310

Parameter | Name: **C00310 | PartitionValue**	Data type: UNSIGNED_16 Index: 24265_d = $5EC9_h$
This code is used device-internally and must not be written by the user side!	

C00311

Parameter | Name: C00311 | Runtime measurement	Data type: UNSIGNED_32 Index: 24264_d = $5EC8_h$
This code is used device-internally and must not be written by the user side!	

C00312

Parameter | Name: C00312 | System runtimes			Data type: UNSIGNED_32 Index: 24263_d = $5EC7_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.000	µs	1310.700	
Subcodes	**Lenze setting**		**Info**
C00312/1	0.000 µs		System runtime reserve
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00313

Parameter | Name: C00313 | LS_DataAccess: Activation	Data type: UNSIGNED_8 Index: 24262_d = $5EC6_h$
This code is used device-internally and must not be written by the user side!	

C00314

Parameter | Name: C00314 | LS_DataAccess: Address access	Data type: UNSIGNED_32 Index: 24261_d = $5EC5_h$
This code is used device-internally and must not be written by the user side!	

C00315

Parameter | Name: C00315 | SystemFail-Adr	Data type: UNSIGNED_32 Index: 24260_d = $5EC4_h$
This code is used device-internally and must not be written by the user side!	

C00316

Parameter | Name: C00316 | SystemFail-Info	Data type: UNSIGNED_16 Index: 24259_d = $5EC3_h$
This code is used device-internally and must not be written by the user side!	

C00317

Parameter | Name: C00317 | WatchdogTimeMax	Data type: UNSIGNED_16 Index: 24258_d = $5EC2_h$
This code is used device-internally and must not be written by the user side!	

C00320

Parameter | Name: C00320 | Debug information	Data type: UNSIGNED_32 Index: 24255_d = $5EBF_h$
This code is used device-internally and must not be written by the user side!	

C00321

Parameter | Name: **C00321 | Main program runtime**			Data type: UNSIGNED_16 Index: 24254_d = $5EBE_h$
Display of the current and the maximum runtime of the main program in the inverter			
Setting range (min. value | unit | max. value)			
0	ms	65535	
Subcodes	**Lenze setting**		**Info**
C00321/1	0 ms		Curr. runtime of main program
C00321/2	0 ms		Max. runtime of main program
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00322

Parameter | Name: **C00322 | Transmission mode CAN TxPDOs**			Data type: UNSIGNED_8 Index: 24253_d = $5EBD_h$

TPDO transmission type according to DS301 V4.02
• The following transmission modes are supported:
  • 0: Synchronous and acyclic
  • 1 ... 240: Synchronous and cyclic
  • 252: Synchronous - RTR only
  • 253: Asynchronous - RTR only
  • 254: Asynchronous - manufacturer-specific
  • 255: Asynchronous - device-profile specific
• The basic setting for all PDOs is "Asynchronous - manufacturer-specific" (254).
• Illustration of the CANopen objects I-1800/2 ... I-1803/2 (see DS301 V4.02).
▶ "CAN on board" system bus			

Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C00322/1	254		Transmission mode CAN1 OUT
C00322/2	254		Transmission mode CAN2 OUT
C00322/3	254		Transmission mode CAN3 OUT
C00322/4	254		Transmission mode CAN4 OUT • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00323

Parameter | Name: C00323 | Transmission mode CAN Rx PDOs			Data type: UNSIGNED_8 Index: 24252_d = $5EBC_h$

RPDO transmission type according to DS301 V4.02
• In the case of the RPDO serves as monitoring setting in the case of sync-controlled PDOs.
• The following transmission modes are supported:
  • 0: Synchronous and acyclic
  • 1 ... 240: Synchronous and cyclic
  • 252: Synchronous - RTR only
  • 253: Asynchronous - RTR only
  • 254: Asynchronous - manufacturer-specific
  • 255: Asynchronous - device-profile specific
• The basic setting for all PDOs is "Asynchronous - manufacturer-specific" (254).
• Illustration of the CANopen objects I-1400/2 ... I-1403/2 (see DS301 V4.02).
▶ "CAN on board" system bus			

Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C00323/1	254		Transmission mode CAN1 IN
C00323/2	254		Transmission mode CAN2 IN
C00323/3	254		Transmission mode CAN3 IN
C00323/4	254		Transmission mode CAN4 IN • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00324

Parameter | Name: C00324 | CAN transmission blocking time			Data type: UNSIGNED_16 Index: 24251_d = $5EBB_h$
Blocking time for the transmission of the emergency telegram and the process data **Note:** If the "Asynchronous - manufacturer-specific/device profile-specific" transmisson type is set, the transmission cycle timer is reset to 0 if event-controlled transmission has been triggered. Example: Cycle time (C00356/x) = 500 ms, blocking time = 100 ms, data change sporadically: • In the case of a sporadic data change < 500 ms, due to the blocking time set, transmission takes place every 100 ms (event-controlled transmission) as quickly as possible. • In the case of a sporadic data change > 500 ms, due to the cycle time set, transmission takes place every 500 ms (cyclic transmission). ▶ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			
0	ms	6500	
Subcodes	**Lenze setting**		**Info**
C00324/1	0 ms		CAN emergency blocking time
C00324/2	0 ms		CAN1_OUT blocking time
C00324/3	0 ms		CAN2_OUT blocking time
C00324/4	0 ms		CAN3_OUT blocking time
C00324/5	0 ms		CAN4_OUT blocking time • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00338

Parameter | Name: **C00338 | L_Arithmetik_1: Function**	Data type: UNSIGNED_8 Index: 24237_d = $5EAD_h$
The L_Arithmetik_1 FB: Selection of internal arithmetics	

Selection list(Lenze setting printed in bold)	
0	**Out = In1**
1	Out = In1 + In2
2	nOut_a = nIn1_a - nIn2_a
3	Out = (In1 * In2) / 100%
4	nOut_a = (nIn1_a * 1%) / |nIn2_a|
5	nOut_a = (nIn1_a * 100%) / (100% - nIn2_a)

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1

C00339

Parameter | Name: **C00339 | L_Arithmetik_2: Function**	Data type: UNSIGNED_8 Index: 24236_d = $5EAC_h$
The L_Arithmetik_2 FB: Selection of internal arithmetics	

Selection list(Lenze setting printed in bold)	
0	**nOut_a = nIn1_a**
1	nOut_a = nIn1_a + nIn2_a
2	nOut_a = nIn1_a - nIn2_a
3	nOut_a = (nIn1_a * nIn2_a) / 100%
4	nOut_a = (nIn1_a * 1%) / |nIn2_a|
5	nOut_a = (nIn1_a * 100%) / (100% - nIn2_a)

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1

C00341

Parameter | Name: **C00341 | CAN management - error configuration**			Data type: UNSIGNED_16 Index: 24234_d = $5EAA_h$
Selection of the events for which the *bFail* error output of the LS_CANMAnagement SB must be set to TRUE.			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0000** (decimal: 0)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	BusOff_MsgErr		
Bit 1 ☐	Warning		
Bit 2 ☐	NodeStopped		
Bit 3 ☐	HeartBeatEvent		
Bit 4 ☐	CAN1_In_Überw.		
Bit 5 ☐	CAN2_In_Überw.		
Bit 6 ☐	CAN3_In_Überw.		
Bit 7 ☐	CAN4_In_Überw.		From version 15.00.00
Bit 8 ☐	Reserved		
Bit 9 ☐	Reserved		
Bit 10 ☐	Reserved		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00342

Parameter | Name: **C00342 | CAN decoupling PDOInOut**			Data type: UNSIGNED_16 Index: 24233_d = $5EA9_h$
Configuration defining the events that lead to a decoupling of the process data words. ▸ Configuring exception handling of the CAN PDOs			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	BusOff_MsgErr		
Bit 1	Warning		
Bit 2	NodeStopped		
Bit 3	HeartBeatEvent		
Bit 4	CAN1_In_Überw.		
Bit 5	CAN2_In_Überw.		
Bit 6	CAN3_In_Überw.		
Bit 7	CAN4_In_Überw.		From version 15.00.00
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Trouble		
Bit 15	Fault		
Subcodes	**Lenze setting**		**Info**
C00342/1	0x0000		CAN decoupling PDO_In from the bus
C00342/2	0x0000		CAN decoupling PDO_Out from the appl.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00343

Parameter | Name: **C00343 | LP_CanIn decoupling value**		Data type: UNSIGNED_16 Index: 24232_d = $5EA8_h$
Definition of the value the process data words are to have in the decoupled state. ▶ Configuring exception handling of the CAN PDOs		
Setting range (min. value | unit | max. value)		
0		65535
Subcodes	**Lenze setting**	**Info**
C00343/1	0	LP_CanIn1:wCtrl DiscVal
C00343/2	0	LP_CanIn1:wIn2 DiscVal
C00343/3	0	LP_CanIn1:wIn3 DiscVal
C00343/4	0	LP_CanIn1:wIn4 DiscVal
C00343/5	0	LP_CanIn2:wIn1 DiscVal
C00343/6	0	LP_CanIn2:wIn2 DiscVal
C00343/7	0	LP_CanIn2:wIn3 DiscVal
C00343/8	0	LP_CanIn2:wIn4 DiscVal
C00343/9	0	LP_CanIn3:wIn1 DiscVal
C00343/10	0	LP_CanIn3:wIn2 DiscVal
C00343/11	0	LP_CanIn3:wIn3 DiscVal
C00343/12	0	LP_CanIn3:wIn4 DiscVal
C00343/13	0	LP_CanIn4:wIn1 DiscVal • From version 15.00.00
C00343/14	0	LP_CanIn4:wIn2 DiscVal • From version 15.00.00
C00343/15	0	LP_CanIn4:wIn3 DiscVal • From version 15.00.00
C00343/16	0	LP_CanIn4:wIn4 DiscVal • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00344

Parameter | Name: **C00344 | LP_CanOut decoupling value**			Data type: UNSIGNED_16 Index: 24231_d = $5EA7_h$
Definition of the value the process data words are to have in the decoupled state. ▶ Configuring exception handling of the CAN PDOs			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C00344/1	0		LP_CanOut1:wState DiscVal
C00344/2	0		LP_CanOut1:wOut2 DiscVal
C00344/3	0		LP_CanOut1:wOut3 DiscVal
C00344/4	0		LP_CanOut1:wOut4 DiscVal
C00344/5	0		LP_CanOut2:wOut1 DiscVal
C00344/6	0		LP_CanOut2:wOut2 DiscVal
C00344/7	0		LP_CanOut2:wOut3 DiscVal
C00344/8	0		LP_CanOut2:wOut4 DiscVal
C00344/9	0		LP_CanOut3:wOut1 DiscVal
C00344/10	0		LP_CanOut3:wOut2 DiscVal
C00344/11	0		LP_CanOut3:wOut3 DiscVal
C00344/12	0		LP_CanOut3:wOut4 DiscVal
C00344/13	0		LP_CanOut4:wOut1 DiscVal • From version 15.00.00
C00344/14	0		LP_CanOut4:wOut2 DiscVal • From version 15.00.00
C00344/15	0		LP_CanOut4:wOut3 DiscVal • From version 15.00.00
C00344/16	0		LP_CanOut4:wOut4 DiscVal • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00345

Parameter | Name: **C00345 | CAN error status**		Data type: UNSIGNED_8 Index: 24230_d = $5EA6_h$
▶ "CAN on board" system bus		
Selection list (read only)		
0	No Error	
1	Warning ErrActive	
2	Warning ErrPassive	
3	Bus off	
4	Reserved	
5	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00347

Parameter | Name: **C00347 | CAN status HeartBeat producer**		Data type: UNSIGNED_8 Index: 24228_d = $5EA4_h$
		▶ Heartbeat protocol
Selection list		
0	Boot-up	
4	Stopped	
5	Operational	
127	Pre-Operat.	
250	Failed	
255	NoResponse	
Subcodes		**Info**
C00347/1		Status node 1 ... 15
C00347/...		
C00347/15		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00349

Parameter | Name: **C00349 | CAN setting - DIP switch**			Data type: UNSIGNED_16 Index: 24226_d = $5EA2_h$
DIP switch setting during last mains power-on			▶ "CAN on board" system bus
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Node address 1		
Bit 1	Node address 2		
Bit 2	Node address 4		
Bit 3	Node address 8		
Bit 4	Node address 16		
Bit 5	Node address 32		
Bit 6	Node address 64		
Bit 7	Baud rate 1		
Bit 8	Baud rate 2		
Bit 9	Baud rate 4		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	DIP switch at 24V-ON accepted		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00350

Parameter | Name: **C00350 | CAN node address**			Data type: UNSIGNED_8 Index: 24225_d = $5EA1_h$
Setting of the node address via parameters • The node address can only be parameterised if the node address "0" is set via the DIP switches. • A change in the node address will not be effective until a CAN Reset Node is performed. ▶ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		127	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00351

Parameter | Name: **C00351 | CAN baud rate**		Data type: UNSIGNED_8 Index: 24224_d = $5EA0_h$
Setting of the baud rate via parameters • The baud rate can only be parameterised if the baud rate "0" is set via the DIP switches. • A change in the baud rate will not be effective until a CAN Reset Node is performed. ▶ "CAN on board" system bus		
Selection list(Lenze setting printed in bold)		
0	**500 kbps**	
1	250 kbps	
2	125 kbps	
3	50 kbps	
4	1000 kbps	
5	20 kbps	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00352

Parameter | Name: **C00352 | CAN slave/master**		Data type: UNSIGNED_8 Index: 24223_d = $5E9F_h$
The drive starts as CAN master after mains switching if a value of "1" has been entered and saved here. ▶ "CAN on board" system bus		
Selection list(Lenze setting printed in bold)		
0	**slave**	
1	master	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00353

Parameter | Name: **C00353 | CAN IN/OUT COBID source**		Data type: UNSIGNED_8 Index: 24222_d = $5E9E_h$
Identifier assignment procedure for the CANx In/Out process data ▸ System bus "CAN on board": Identifier of the process data objects		
Selection list		**Info**
0	COBID = C0350 + LenzeBaseID	COBID = node address + LenzeBaseID
1	COBID = C0350 + CANBaseID	COBID = node address + CANBaseID (C00354/x)
2	COBID = C0354/x	COBID = direct setting from C00354/x
Subcodes	**Lenze setting**	**Info**
C00353/1	1: COBID = C0350 + CANBaseID	COBID source CAN1_IN/OUT
C00353/2	1: COBID = C0350 + CANBaseID	COBID source CAN2_IN/OUT
C00353/3	1: COBID = C0350 + CANBaseID	COBID source CAN3_IN/OUT
C00353/4	1: COBID = C0350 + CANBaseID	COBID source CAN4_IN/OUT • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00354

Parameter | Name: **C00354 | COBID**			Data type: UNSIGNED_32 Index: 24221_d = $5E9D_h$
Setting of the default COBID according to CANopen • A change in the COBID will not be effective until a CAN reset node is performed. ▸ System bus "CAN on board": Identifier of the process data objects			
Setting range (min. hex value | max. hex value)			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	COBID Bit0		• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: PDO invalid (is not transmitted)
...	...		
Bit 31	PDO invalid		
Subcodes	**Lenze setting**		**Info**
C00354/1	0x00000201		COBID CAN1_IN
C00354/2	0x00000181		COBID CAN1_OUT
C00354/3	0x00000301		COBID CAN2_IN
C00354/4	0x00000281		COBID CAN2_OUT
C00354/5	0x00000401		COBID CAN3_IN
C00354/6	0x00000381		COBID CAN3_OUT
C00354/7	0x00000501		COBID CAN4_IN • From version 15.00.00
C00354/8	0x00000481		COBID CAN4_OUT • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT			

C00355

Parameter | Name: **C00355 | Active COBID**			Data type: UNSIGNED_16 Index: 24220_d = $5E9C_h$
Display of the COBID of the PDOs that is active in the CAN stack ▸ System bus "CAN on board": Identifier of the process data objects			
Display range (min. value | unit | max. value)			
0		2047	
Subcodes			**Info**
C00355/1			Active COBID CAN1_IN
C00355/2			Active COBID CAN1_OUT
C00355/3			Active COBID CAN2_IN
C00355/4			Active COBID CAN2_OUT
C00355/5			Active COBID CAN3_IN
C00355/6			Active COBID CAN3_OUT
C00355/7			Active COBID CAN4_IN • From version 15.00.00
C00355/8			Active COBID CAN4_OUT • From version 15.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00356

Parameter | Name: **C00356 | CAN time settings**			Data type: UNSIGNED_16 Index: 24219_d = $5E9B_h$
Different time settings for the CAN interface ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			
0	ms	65000	
Subcodes	**Lenze setting**		**Info**
C00356/1	3000 ms		CAN delay boot-up - operat. • Delay time for NMT state change from "Boot-up" to "Operational".
C00356/2	0 ms		CAN2_OUT cycle time
C00356/3	0 ms		CAN3_OUT cycle time
C00356/4	0 ms		CANx_OUT operat. - 1.transmit • When the "Operational" NMT status is reached, the delay time set here is started. After the delay time has been elapsed, the PDOs CAN2_OUT and CAN3_OUT are sent for the first time.
C00356/5	0 ms		CAN1_OUT cycle time
C00356/6	0 ms		CAN4_OUT cycle time • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00357

Parameter | Name: **C00357 | CAN monitoring times**			Data type: UNSIGNED_16 Index: 24218_d = $5E9A_h$
Mapping of the RPDO event time (see DS301 V4.02) • If a value unequal to "0" is entered, the RPDO is not expected before the set time has expired. • If the RPDO is not received within the expected time, the response set in C00593/1...4 will be triggered. ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			
0	ms	65000	
Subcodes	**Lenze setting**		**Info**
C00357/1	3000 ms		CAN1_IN monitoring time
C00357/2	3000 ms		CAN2_IN monitoring time
C00357/3	3000 ms		CAN3_IN monitoring time
C00357/4	3000 ms		CAN4_IN monitoring time • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00358

Parameter | Name: **C00358 | CANx_OUT data length**			Data type: UNSIGNED_8 Index: 24217_d = $5E99_h$
Setting of the data length for TX PDOs ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			
1		8	
Subcodes	**Lenze setting**		**Info**
C00358/1	8		CAN1_OUT data length
C00358/2	8		CAN2_OUT data length
C00358/3	8		CAN3_OUT data length
C00358/4	8		CAN4_OUT data length • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00359

Parameter | Name: **C00359 | CAN status**		Data type: UNSIGNED_8 Index: 24216_d = $5E98_h$
▸ "CAN on board" system bus		
Selection list (read only)		
0	Operational	
1	Pre-Operat.	
2	Reserved	
3	Reserved	
4	BootUp	
5	Stopped	
6	Reserved	
7	Reset	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00360

Parameter | Name: **C00360 | CAN telegram counter**			Data type: UNSIGNED_16 Index: 24215_d = $5E97_h$
			▶ "CAN on board" system bus
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C00360/1			All PDO/SDO sent
C00360/2			All PDO/SDO received
C00360/3			Telegram counter CAN1_OUT
C00360/4			Telegram counter CAN2_OUT
C00360/5			Telegram counter CAN3_OUT
C00360/6			Telegram counter SDO1 OUT
C00360/7			Telegram counter SDO2 OUT
C00360/8			Telegram counter CAN1_IN
C00360/9			Telegram counter CAN2_IN
C00360/10			Telegram counter CAN3_IN
C00360/11			Telegram counter SDO1 IN
C00360/12			Telegram counter SDO2 IN
C00360/13			Telegram counter CAN4_OUT • From version 15.00.00
C00360/14			Telegram counter CAN4_IN • From version 15.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00364

Parameter | Name: **C00364 | CAN MessageError**		Data type: UNSIGNED_8 Index: 24211_d = $5E93_h$
		▶ "CAN on board" system bus
Selection list (read only)		
0	No Error	
1	StuffError	
2	FormError	
3	AckError	
4	Bit1Error	
5	Bit0Error	
6	CRCError	
7	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00366

Parameter | Name: **C00366 | Number of CAN SDO channels**	Data type: UNSIGNED_8 Index: 24209_d = $5E91_h$
Selection of the number of active parameter data channels • In the Lenze setting, only the parameter data channel 1 is activated according to CANopen. • In order to activate both parameter data channels, set the selection "2 SDO Lenze". ▸ "CAN on board" system bus	
Selection list(Lenze setting printed in bold)	
0	**1 SDO CANOpen**
1	2 SDO Lenze
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1	

C00367

Parameter | Name: **C00367 | CAN SYNC Rx identifier**			Data type: UNSIGNED_16 Index: 24208_d = $5E90_h$
Identifier by means of which the sync slave is to receive sync telegrams. • Mapping of the CANopen object I-1005 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0080** (decimal: 128)
Value is bit-coded: (☑ = bit set)			
Bit 0 ☐	COBID Bit0		
Bit 1 ☐	COBID Bit1		
Bit 2 ☐	COBID Bit2		
Bit 3 ☐	COBID Bit3		
Bit 4 ☐	COBID Bit4		
Bit 5 ☐	COBID Bit5		
Bit 6 ☐	COBID Bit6		
Bit 7 ☑	COBID Bit7		
Bit 8 ☐	COBID Bit8		
Bit 9 ☐	COBID Bit9		
Bit 10 ☐	COBID Bit10		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT			

C00368

Parameter | Name: **C00368 | CAN SYNC Tx identifier**			Data type: UNSIGNED_16 Index: 24207_d = $5E8F_h$
Identifier by means of which the sync master is to transmit sync telegrams. • Mapping of the CANopen object I-1005 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0080** (decimal: 128)
Value is bit-coded: (☑ = bit set)			
Bit 0 ☐	COBID Bit0		
Bit 1 ☐	COBID Bit1		
Bit 2 ☐	COBID Bit2		
Bit 3 ☐	COBID Bit3		
Bit 4 ☐	COBID Bit4		
Bit 5 ☐	COBID Bit5		
Bit 6 ☐	COBID Bit6		
Bit 7 ☑	COBID Bit7		
Bit 8 ☐	COBID Bit8		
Bit 9 ☐	COBID Bit9		
Bit 10 ☐	COBID Bit10		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Sync-transmit off		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT			

C00369

Parameter | Name: **C00369 | CAN sync transmission cycle time**			Data type: UNSIGNED_16 Index: 24206_d = $5E8E_h$
Cycle during which the sync master is to transmit sync telegrams. • If "0 ms" is set (Lenze setting), no sync telegrams are generated. • Mapping of the CANopen object I-1006 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00370

Parameter | Name: **C00370 | SyncTxRxTimes**			Data type: INTEGER_16 Index: 24205$_d$ = 5E8D$_h$
			"CAN on board" system bus
Display range (min. value | unit | max. value)			
-1310	µs	1310	
Subcodes			**Info**
C00370/1			CAN Sync instant of transmission
C00370/2			Sync instant of reception
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00371

Parameter | Name: **C00371 | CAN ErrorCode**			Data type: UNSIGNED_16 Index: 24204$_d$ = 5E8C$_h$
From version 13.00.00			"CAN on board" system bus
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C00371/1			CAN ErrorCode
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00372

Parameter | Name: **C00372 | CAN_Tx_Rx_Error**			Data type: UNSIGNED_8 Index: 24203$_d$ = 5E8B$_h$
			"CAN on board" system bus
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C00372/1			CAN Tx_Error
C00372/2			CAN Rx_Error
C00372/3			CAN Tx_Overflow
C00372/4			CAN Rx_Overflow
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00381

Parameter | Name: **C00381 | CAN Heartbeat producer time**			Data type: UNSIGNED_16 Index: 24194$_d$ = 5E82$_h$
Time interval for the transmission of the heartbeat telegram to the consumer(s). • The heartbeat telegram is sent automatically as soon as a time > 0 ms is set. • Mapping of the CANopen object I-1017 (see DS301 V4.02).			Heartbeat protocol
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C00385

Parameter | Name: **C00385 | CAN node addr. HeartBeat producer**			Data type: UNSIGNED_8 Index: 24190_d = $5E7E_h$
The subcodes represent the nodes to be monitored by heartbeat.			▸ Heartbeat protocol
Setting range (min. value | unit | max. value)			
0		127	
Subcodes	**Lenze setting**		**Info**
C00385/1	0		CAN node address HeartBeat producer 1 ... 15
C00385/...			
C00385/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00386

Parameter | Name: **C00386 | CAN HeartBeat ConsumerTime**			Data type: UNSIGNED_16 Index: 24189_d = $5E7D_h$
The subcodes represent the nodes to be monitored by heartbeat.			▸ Heartbeat protocol
Setting range (min. value | unit | max. value)			
0	ms	60000	
Subcodes	**Lenze setting**		**Info**
C00386/1	0 ms		ConsumerTime HeartBeat producer 1 ... 15
C00386/...			
C00386/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00387

Parameter | Name: **C00387 | CAN-GatewayAddr**			Data type: UNSIGNED_8 Index: 24188_d = $5E7C_h$
From version 12.00.00			▸ CAN gateway
Setting range (min. value | unit | max. value)			
0		127	
Subcodes	**Lenze setting**		**Info**
C00387/1	0		CAN_Gateway: Addr.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00400

Parameter | Name: **C00400 | LS_PulseGenerator**		Data type: UNSIGNED_16 Index: 24175_d = $5E6F_h$
Time setting of the pulse to be output by the SB LS_PulseGenerator **Note:** The real length of the low or high level at the *bSquareWave* output is always 1 ms higher than set here in subcode 1 or 2. If "0 ms" is set, a level with a length of 1 ms is created length, for instance, and if "1000 ms" is set, a level with a length of 1001 ms is created.		
Setting range (min. value | unit | max. value)		
0	ms	60000
Subcodes	**Lenze setting**	**Info**
C00400/1	1000 ms	Length of the low level (break) + 1 ms
C00400/2	1000 ms	Length of the high level + 1 ms
C00400/3	100 ms	Delay time for status signal *bFirstCycleDone* • The *bFirstCycleDone* status signal is set to TRUE when the first task cycle is complete and the time set here has expired.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00401

Parameter | Name: **C00401 | CANxInOut: Inversion**		Data type: UNSIGNED_16 Index: 24174_d = $5E6E_h$
This parameter serves to invert the control/status bits of the CAN port blocks. ▶ CAN port block		
Setting range (min. hex value | max. hex value)		
0x0000		0xFFFF
Value is bit-coded:		**Info**
Bit 0	Active	Bit set = bit is inverted
...	...	
Bit 15	Active	
Subcodes	**Lenze setting**	**Info**
C00401/1	0x0000	Inversion of LP_CanIn1.bCtrl1_B0...15
C00401/2	0x0000	Inversion of LP_CanOut1.bState1_B0...15
C00401/3	0x0000	Inversion of LP_CanIn2.bIn1_B0...15
C00401/4	0x0000	Inversion of LP_CanOut2.bOut1_B0...15
C00401/5	0x0000	Inversion of LP_CanIn3.bIn1_B0...15
C00401/6	0x0000	Inversion of LP_CanOut3.bOut1_B0...15
C00401/7	0x0000	Inversion of LP_CanIn4.bIn1_B0...15 • From version 15.00.00
C00401/8	0x0000	Inversion of LP_CanOut4.bOut1_B0...15 • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT		

C00407

Parameter | Name: **C00407 | LP_CanIn Mapping**			Data type: UNSIGNED_32 Index: 24168_d = $5E68_h$
From version 12.00.00 Mapping for port blocks LP_CanIn1...4 • These settings are ORed with the mapping settings for the single words in C00409/x. ▸ CAN port block			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**		**Info**
C00407/1	0		LP_CanIn1:dwIn12 MapVal
C00407/2	0		LP_CanIn1:dwIn34 MapVal
C00407/3	0		LP_CanIn2:dwIn12 MapVal
C00407/4	0		LP_CanIn2:dwIn34 MapVal
C00407/5	0		LP_CanIn3:dwIn12 MapVal
C00407/6	0		LP_CanIn3:dwIn34 MapVal
C00407/7	0		LP_CanIn4:dwIn12 MapVal • From version 15.00.00
C00407/8	0		LP_CanIn4:dwIn34 MapVal • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00408

Parameter | Name: **C00408 | LP_CanIn mapping selection**		Data type: UNSIGNED_8 Index: 24167_d = $5E67_h$
Selection of the mapping source for port blocks LP_CanIn1...4 ▸ CAN port block		
Selection list		**Info**
0	CanIn	CanIn
1	Par.C409|C407	Mapping configured in C00409
Subcodes	**Lenze setting**	**Info**
C00408/1	0: CanIn	Mapping selection LP_CanIn1
C00408/2	0: CanIn	Mapping selection LP_CanIn2
C00408/3	0: CanIn	Mapping selection LP_CanIn3
C00408/4	0: CanIn	Mapping selection LP_CanIn4 • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00409

Parameter | Name: **C00409 | LP_CanIn Mapping**			Data type: UNSIGNED_16 Index: 24166_d = $5E66_h$
Mapping for port blocks LP_CanIn1...4 • These settings are ORed with the mapping settings for the double words in C00407/x. ▸ CAN port block			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C00409/1	0		LP_CanIn1:wCtrl MapVal
C00409/2	0		LP_CanIn1:wIn2 MapVal
C00409/3	0		LP_CanIn1:wIn3 MapVal
C00409/4	0		LP_CanIn1:wIn4 MapVal
C00409/5	0		LP_CanIn2:wIn1 MapVal
C00409/6	0		LP_CanIn2:wIn2 MapVal
C00409/7	0		LP_CanIn2:wIn3 MapVal
C00409/8	0		LP_CanIn2:wIn4 MapVal
C00409/9	0		LP_CanIn3:wIn1MapVal
C00409/10	0		LP_CanIn3:wIn2 MapVal
C00409/11	0		LP_CanIn3:wIn3 MapVal
C00409/12	0		LP_CanIn3:wIn4 MapVal
C00409/13	0		LP_CanIn4:wIn1 MapVal • From version 15.00.00
C00409/14	0		LP_CanIn4:wIn2 MapVal • From version 15.00.00
C00409/15	0		LP_CanIn4:wIn3 MapVal • From version 15.00.00
C00409/16	0		LP_CanIn4:wIn4 MapVal • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00410

Parameter | Name: **C00410 | L_SignalMonitor_a: Signal sources**		Data type: UNSIGNED_16 Index: 24165_d = $5E65_h$
The L_SignalMonitor_a FB: Selection of the signal sources		
Selection list		
See selection list - analog signals		
Subcodes	**Lenze setting**	**Info**
C00410/1	0: Not connected	Signal source for output *nOut1_a*
C00410/2	0: Not connected	Signal source for output *nOut2_a*
C00410/3	0: Not connected	Signal source for output *nOut3_a*
C00410/4	0: Not connected	Signal source for output *nOut4_a*
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00411

Parameter | Name: **C00411 | L_SignalMonitor_b: Signal sources**		Data type: UNSIGNED_16 Index: 24164_d = $5E64_h$
The L_SignalMonitor_b FB: Selection of the signal sources		
Selection list		
See selection list - digital signals		
Subcodes	**Lenze setting**	**Info**
C00411/1	0: Not connected	Signal source for output *bOut1*
C00411/2	0: Not connected	Signal source for output *bOut2*
C00411/3	0: Not connected	Signal source for output *bOut3*
C00411/4	0: Not connected	Signal source for output *bOut4*
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00412

Parameter | Name: **C00412 | L_SignalMonitor_b: Inversion**			Data type: UNSIGNED_8 Index: 24163_d = $5E63_h$
The L_SignalMonitor_b FB: Inversion of the binary outputs			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x00** (decimal: 0)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	bOut1 inverted		Bit set = inversion active
Bit 1 ☐	bOut2 inverted		
Bit 2 ☐	bOut3 inverted		
Bit 3 ☐	bOut4 inverted		
Bit 4 ☐	Reserved		
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00413

Parameter | Name: **C00413 | L_SignalMonitor_a: Offs./gain**			Data type: INTEGER_16 Index: 24162_d = $5E62_h$
The L_SignalMonitor_a FB: Gain and offset of the analog signals			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00413/1	0.00 %		Offset for output *nOut1_a*
C00413/2	100.00 %		Gain for output *nOut1_a*
C00413/3	0.00 %		Offset for output *nOut2_a*
C00413/4	100.00 %		Gain for output *nOut2_a*
C00413/5	0.00 %		Offset for output *nOut3_a*
C00413/6	100.00 %		Gain for output *nOut3_a*
C00413/7	0.00 %		Offset for output *nOut4_a*
C00413/8	100.00 %		Gain for output *nOut4_a*
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00417

Parameter | Name: **C00417 | Deactivate resolver error comp.**		Data type: UNSIGNED_8 Index: 24158_d = $5E5E_h$
		▸ Encoder/feedback system: Resolver
Selection list(Lenze setting printed in bold)		
0	**Resolver error comp. act.**	
1	Resolver error comp. deact.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00420

Parameter | Name: **C00420 | Number of encoder increments**			Data type: UNSIGNED_16 Index: 24155_d = $5E5B_h$
Indication of the encoder constant ▸ Encoder/feedback system: Multi-Encoder			
Setting range (min. value | unit | max. value)			
1	Incr./rev.	32767	
Subcodes	**Lenze setting**		**Info**
C00420/1	128 incr./rev.		FreqIn12: Encoder increment
C00420/2	128 incr./rev.		FreqIn67: Encoder increment
C00420/3	512 incr./rev.		LS_MultiEncoder: Encoder increment
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00421

Parameter | Name: **C00421 | LS_MultiEncoder: Supply voltage**			Data type: UNSIGNED_16 Index: 24154_d = $5E5A_h$
Stop! Before connecting an encoder, make sure that the encoder supply voltage is set correctly! If the set supply voltage exceeds the permissible supply voltage of the connected encoder, the encoder may be destroyed! ▸ Encoder/feedback system: Multi-Encoder			
Setting range (min. value | unit | max. value)			**Lenze setting**
5.0	V	12.0	**5.0 V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00422

Parameter | Name: **C00422 | LS_MultiEncoder: Encoder type**		Data type: UNSIGNED_8 Index: 24153_d = $5E59_h$
		▶ Encoder/feedback system: Multi-Encoder
Selection list(Lenze setting printed in bold)		**Info**
0	**Incremental encoder (TTL)**	
1	Sin/cos encoder	
2	Absolute value encoder (Hiperface)	
3	Absolute value encoder (external)	
4	Absolute value encoder (SSI)	
5	LF In/Out	From version 12.00.00 ▶ Digital frequency coupling
6	Position encoder (Hiperface)	
7	Position encoder (external)	
8	Position encoder (SSI)	
10	Incremental encoder (TTL) with LF Out	
11	Sine/cos encoder with LF Out	
12	Absolute value encoder Hiperface with LF Out	
16	Position encoder Hiperface with LF Out	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00423

Parameter | Name: **C00423 | DOx: Delay times**			Data type: UNSIGNED_16 Index: 24152_d = $5E58_h$
Delay times for the digital output terminals ▶ Digital output terminals			
Setting range (min. value | unit | max. value)			
0.000	s	65.000	
Subcodes	**Lenze setting**		**Info**
C00423/1	0.000 s		Relay ON delay
C00423/2	0.000 s		Relay OFF delay
C00423/3	0.000 s		DO1 ON delay
C00423/4	0.000 s		DO1 OFF delay
C00423/5	0.000 s		DO2 ON delay
C00423/6	0.000 s		DO2 OFF delay
C00423/7	0.000 s		DO3 ON delay
C00423/8	0.000 s		DO3 OFF delay
C00423/9	0.000 s		DO "High Current" ON delay
C00423/10	0.000 s		DO "High Current" OFF delay
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00424

Parameter | Name: **C00424 | Pulse form TTL encoder**		Data type: UNSIGNED_8 Index: 24151_d = $5E57_h$
		▶ Encoder/feedback system: Multi-Encoder
Selection list(Lenze setting printed in bold)		
0	**4x evaluation A/B**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00425

Parameter | Name: **C00425 | Encoder scanning time**		Data type: UNSIGNED_8 Index: 24150_d = $5E56_h$
Encoder sample time for the digital input terminals when configured as frequency inputs ▶ Using DI1(6) and DI2(7) as frequency inputs		
Selection list		
0	1 ms	
1	2 ms	
2	5 ms	
3	10 ms	
4	20 ms	
5	50 ms	
6	100 ms	
7	200 ms	
8	500 ms	
9	1000 ms	
Subcodes	**Lenze setting**	**Info**
C00425/1	3: 10 ms	FreqIn12: Encoder scanning time • Only active with edge-counting procedure (C00496 = 3).
C00425/2	3: 10 ms	FreqIn67: Encoder scanning time
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00426

Parameter | Name: **C00426 | SSI encoder: Data bits**			Data type: UNSIGNED_8 Index: 24149_d = $5E55_h$
Example: Setting for an SSI encode with the coding "8192x4096": • 8192 = 2^{13} = 13 bits Multiturn ➞ C00426/4 = 13 • 4096 = 2^{12} = 12 bits Singleturn ➞ C00426/2 = 12 • This results in a data word length of 25 bits ➞ C00426/1 = 25 • Start bit Singleturn = bit 0 ➞ C00426/3 = 0 • Start bit Multiturn = bit 12 ➞ C00426/5 = 12 ▸ Encoder/feedback system: SSI encoder			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Information**
C00426/1	25		SSI encoder: Data word length
C00426/2	13		SSI encoder: Bits SingleTurn
C00426/3	0		SSI encoder: Start bit Singleturn
C00426/4	12		SSI encoder: Bits Multiturn
C00426/5	13		SSI encoder: Start bit Multiturn
C00426/6	0		SSI encoder: Status bit 1 • From version 02.00.00
C00426/7	0		SSI encoder: Shift of raw value • From version 02.00.00
C00426/8	0		SSI encoder: Status bit 2 • Ab Version 14.00.00
C00426/9	0		SSI encoder: Status bit 3 • Ab Version 14.00.00
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00427

Parameter | Name: **C00427 | SSI encoder: Bit rate**		Data type: UNSIGNED_8 Index: 24148_d = $5E54_h$
▸ Encoder/feedback system: SSI encoder		
Selection list(Lenze setting printed in bold)		
1	**100 kbps**	
2	200 kbps	
3	300 kbps	
4	400 kbps	
5	500 kbps	
6	750 kbps	
7	1000 kbit/s	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00428

Parameter | Name: **C00428 | SSI encoder: Coding**			Data type: UNSIGNED_8 Index: 24147_d = $5E53_h$
			Encoder/feedback system: SSI encoder
Selection list (Lenze setting printed in bold)		**Info**	
0	**Binary code**		
1	Gray code		
2	Position Gray	From version 14.00.00, SSI encoders are supported which transmit the position data with Gray coding and the status bits in binary format.	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00430

Parameter | Name: **C00430 | LS_MultiEncoder: Max. initialization time**			Data type: UNSIGNED_16 Index: 24145_d = $5E51_h$
From version 12.00.00 Encoder/feedback system: sin/cos absolute value encoder with HIPERFACE® protocol			
Setting range (min. value | unit | max. value)			
500	ms	20000	
Subcodes	**Lenze setting**		**Information**
C00430/1	1500 ms		Maximum time required by the encoder to respond to the read request of the inverter.
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00431

Parameter | Name: **C00431 | LS_MultiEncoder: bFail selection**			Data type: UNSIGNED_16 Index: 24144_d = $5E50_h$
From version 13.00.00 Bit coded setting which events cause the *bFail* output to be set.			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Encoder open circuit		
Bit 1	KTY open circuit		
Bit 2	Position beyond C1112/2..3		
Bit 3	Angular drift error		
Bit 4	Communication error		
Bit 5	Hiperface position invalid		Ab Version 14.00.00
Bit 6	Linear data overflow SSI		Ab Version 14.00.00
Bit 7	Max. encoder speed		Ab Version 14.00.00
Bit 8	Hiperface init active		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Error - encoder gain		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Information**
C00431/1	0x001D		LS_MultiEncoder: bFail selection
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C00432

Parameter | Name: **C00432 | LS_Resolver: bFail selection**			Data type: UNSIGNED_16 Index: 24143_d = $5E4F_h$
From version 13.00.00 Bit coded setting which events cause the *bFail* output to be set.			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Resolver wire breakage		
Bit 1	KTY open circuit		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Information**
C00432/1	0x0001		LS_Resolver: bFail selection
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C00434

Parameter | Name: **C00434 | OxU/I: Gain**			Data type: INTEGER_16 Index: 24141_d = $5E4D_h$
Gain of the analog outputs ▸ Analog terminals			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00434/1	100.00 %		O1U: Gain
C00434/2	100.00 %		O2U: Gain
C00434/3	100.00 %		O1I: Gain
C00434/4	100.00 %		O2I: Gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00435

Parameter | Name: **C00435 | OxU/I: Offset**			Data type: INTEGER_16 Index: 24140_d = $5E4C_h$
Offset of the analog outputs ▶ Analog terminals			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00435/1	0.00 %		O1U: Offset
C00435/2	0.00 %		O2U: Offset
C00435/3	0.00 %		O1I: Offset
C00435/4	0.00 %		O2I: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00436

Parameter | Name: **C00436 | OxU: Voltage**			Data type: INTEGER_16 Index: 24139_d = $5E4B_h$
Display of the voltage at the analog outputs ▶ Analog terminals			
Display range (min. value | unit | max. value)			
0.00	V	10.00	
Subcodes			**Info**
C00436/1			O1U: Voltage
C00436/2			O2U: Voltage
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00437

Parameter | Name: **C00437 | OxI: Current**			Data type: INTEGER_32 Index: 24138_d = $5E4A_h$
Display of the current at the analog outputs ▶ Analog terminals			
Display range (min. value | unit | max. value)			
0.000	mA	20.000	
Subcodes			**Info**
C00437/1			O1I: Current
C00437/2			O2I: Current
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00439

Parameter | Name: **C00439 | OxU/I: Input value**			Data type: INTEGER_16 Index: 24136_d = $5E48_h$
Display of the input values for the analog outputs ▸ Analog terminals			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00439/1			O1U: Input value • 100 % ≡ 10 V
C00439/2			O2U: Input value • 100 % ≡ 10 V
C00439/3			O1I: Input value • 100 % ≡ 20 mA
C00439/4			O2I: Input value • 100 % ≡ 20 mA
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00440

Parameter | Name: **C00440 | LS_AnalogIn1: PT1 time constant**			Data type: UNSIGNED_16 Index: 24135_d = $5E47_h$
PT1 time constant (S-ramp time) for the analog inputs ▸ Analog terminals			
Setting range (min. value | unit | max. value)			
0	ms	1000	
Subcodes	**Lenze setting**		**Info**
C00440/1	10 ms		PT1 rounding AnalogIn1
C00440/2	10 ms		PT1 rounding AnalogIn2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00441

Parameter | Name: **C00441 | Decoupling AnalogOut**			Data type: UNSIGNED_16 Index: 24134_d = $5E46_h$
Configuration defining the events that lead to a decoupling of the analog output terminals. ▸ Configuring exception handling of the output terminals			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0000** (decimal: 0)
Value is bit-coded: (☑ = bit set)			
Bit 0 ☐	SafeTorqueOff		
Bit 1 ☐	ReadyToSwitchOn		
Bit 2 ☐	SwitchedOn		
Bit 3 ☐	Reserved		
Bit 4 ☐	Trouble		
Bit 5 ☐	Fault		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
Bit 8 ☐	Reserved		
Bit 9 ☐	Fail CAN_Management		
Bit 10 ☐	Reserved		
Bit 11 ☐	Energy saving mode		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00442

Parameter | Name: **C00442 | AOutx: Decoupling value**			Data type: INTEGER_16 Index: 24133_d = $5E45_h$
Definition of the value the analog output terminals are to have in the decoupled state. ▸ Configuring exception handling of the output terminals			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C00442/1	0.00 %		AOut1_U: Decoupling value
C00442/2	0.00 %		AOut2_U: Decoupling value
C00442/3	0.00 %		AOut1_I: Decoupling value
C00442/4	0.00 %		AOut2_I: Decoupling value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00443

Parameter | Name: **C00443 | DIx: Level**			Data type: UNSIGNED_16 Index: 24132_d = $5E44_h$
Bit coded display of the level of the digital inputs ▸ Digital input terminals			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	DI1		Bit set = HIGH level
Bit 1	DI2		
Bit 2	DI3		
Bit 3	DI4		
Bit 4	DI5		
Bit 5	DI6		
Bit 6	DI7		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	CINH		
Subcodes			**Info**
C00443/1			DIx: Terminal level
C00443/2			DIx: Output level
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00444

Parameter | Name: **C00444 | DOx: Level**			Data type: UNSIGNED_16 Index: 24131_d = $5E43_h$
Bit coded display of the level of the digital outputs ▸ Digital output terminals			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Relay		Bit set = HIGH level
Bit 1	DO1		
Bit 2	DO2		
Bit 3	DO3		
Bit 4	High current		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes			**Info**
C00444/1			DOx: Input level
C00444/2			DOx: Terminal level
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00445

Parameter | Name: **C00445 | FreqInxx_nOut_v**			Data type: INTEGER_16 Index: 24130_d = $5E42_h$
Display of the frequency input signals which are fed into the application. ▸ Using DI1(6) and DI2(7) as frequency inputs			
Display range (min. value | unit | max. value)			
-32767	Incr/ms	32767	
Subcodes			**Info**
C00445/1			FreqIn12_nOut_v
C00445/2			FreqIn67_nOut_v
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00446

Parameter | Name: C00446 | FreqInxx_nOut_a			Data type: INTEGER_16 Index: 24129_d = $5E41_h$
Display of the frequency input signals which are fed into the application. ▶ Using DI1(6) and DI2(7) as frequency inputs			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00446/1			FreqIn12_nOut_a
C00446/2			FreqIn67_nOut_a
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00447

Parameter | Name: C00447 | DigOut decoupling			Data type: UNSIGNED_16 Index: 24128_d = $5E40_h$
Configuration defining the events that lead to a decoupling of the digital output terminals. ▶ Configuring exception handling of the output terminals			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0000** (decimal: 0)
Value is bit-coded: (☑ = bit set)			
Bit 0 ☐	SafeTorqueOff		
Bit 1 ☐	ReadyToSwitchOn		
Bit 2 ☐	SwitchedOn		
Bit 3 ☐	Reserved		
Bit 4 ☐	Trouble		
Bit 5 ☐	Fault		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
Bit 8 ☐	Reserved		
Bit 9 ☐	Fail CAN_Management		
Bit 10 ☐	Reserved		
Bit 11 ☐	Energy saving mode		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00448

Parameter | Name: **C00448 | DigOut decoupling value**			Data type: UNSIGNED_16 Index: 24127_d = $5E3F_h$
Definition of the value the digital output terminals are to have in the decoupled state. • Bit set = HIGH level ▸ Configuring exception handling of the output terminals			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0000** (decimal: 0)
Value is bit-coded: (☑ = bit set)			
Bit 0 ☐	Relay_ON		
Bit 1 ☐	DigOut1_ON		
Bit 2 ☐	DigOut2_ON		
Bit 3 ☐	DigOut3_ON		
Bit 4 ☐	HighCurrent_ON		
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
Bit 8 ☐	Reserved		
Bit 9 ☐	Reserved		
Bit 10 ☐	Reserved		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00449

Parameter | Name: **C00449 | FreqInxx_dnOut_p**			Data type: INTEGER_32 Index: 24126_d = $5E3E_h$
▸ Output of the encoder position of the DI1/DI2 frequency input			
Display range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes			**Info**
C00449/1			FreqIn12_dnOut_p
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00450

Parameter | Name: **C00450 | HTL encoder input frequency**			Data type: UNSIGNED_32 Index: 24125_d = $5E3D_h$
Display range (min. value | unit | max. value)			
0.000	kHz	2147483.647	
Subcodes			**Info**
C00450/1			FreqIn12: input frequency
C00450/2			FreqIn67: input frequency
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00451

Parameter | Name: **C00451 | LS_MultiEncoder: bFail information**			Data type: UNSIGNED_16 Index: 24124_d = $5E3C_h$
Ab Version 14.00.00			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Encoder open circuit		
Bit 1	KTY open circuit		
Bit 2	Position beyond C1112/2..3		
Bit 3	Angular drift error		
Bit 4	Communication error		
Bit 5	Hiperface position invalid		
Bit 6	Linear data overflow SSI		
Bit 7	Max. encoder speed		
Bit 8	Hiperface init active		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Error - encoder gain		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes			**Information**
C00451/1			LS_MultiEncoder: bFail information
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☑ MOT			

C00452

Parameter | Name: **C00452 | LS_Resolver: bFail information**			Data type: UNSIGNED_16 Index: 24123_d = $5E3B_h$
Ab Version 14.00.00			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Resolver wire breakage		
Bit 1	KTY open circuit		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes			**Information**
C00452/1			LS_Resolver: bFail information
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☑ MOT			

C00453

Parameter | Name: **C00453 | Keypad: Default manual jog**			Data type: UNSIGNED_32 Index: 24122_d = $5E3A_h$
From version 14.00.00			
Setting range (min. value | unit | max. value)			
0.000		16000.000	
Subcodes	**Lenze setting**		**Info**
C00453/1	729.001		Keypad: Speed setpoint
C00453/2	56.002		Keypad: Display bar
C00453/3	51.000		Keypad: Def. parameter
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00455

Parameter | Name: **C00455 | FB_call table**	Data type: UNSIGNED_16 Index: 24120_d = $5E38_h$
This code is used device-internally and must not be written by the user side!	

C00456

Parameter | Name: **C00456 | Editor level**	Data type: UNSIGNED_8 Index: 24119_d = $5E37_h$
This code is used device-internally and must not be written by the user side!	

C00458

Parameter | Name: **C00458 | SYS_call table**	Data type: UNSIGNED_16 Index: 24117_d = $5E35_h$
This code is used device-internally and must not be written by the user side!	

C00459

Parameter | Name: **C00459 | SYS_Output table**	Data type: UNSIGNED_16 Index: 24116_d = $5E34_h$
This code is used device-internally and must not be written by the user side!	

C00460

Parameter | Name: **C00460 | Reserved**	Data type: UNSIGNED_8 Index: 24115_d = $5E33_h$
This code is used device-internally and must not be written by the user side!	

C00461

Parameter | Name: **C00461 | Remote: Acceleration/deceleration time**			Data type: UNSIGNED_32 Index: 24114_d = $5E32_h$
			▶ PC manual control
Setting range (min. value | unit | max. value)			
0.000	s	999.999	
Subcodes	**Lenze setting**		**Info**
C00461/1	2.000 s		Remote: Acceleration/deceleration time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00462

Parameter | Name: **C00462 | Remote: Control**			Data type: UNSIGNED_16 Index: 24113_d = $5E31_h$
			▶ PC manual control
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C00462/1	0		Remote: Control mode
C00462/2	0		Remote: Monitoring counter
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00463

Parameter | Name: **C00463 | Remote: MCK control**		Data type: UNSIGNED_32 Index: 24112_d = $5E30_h$
This parameter serves to control the functions of the Motion Control Kernel for PC manual control.		
Setting range (min. hex value | max. hex value)		
0x00000000	0xFFFFFFFF	
Value is bit-coded:		
Bit 0	OpMode_Bit0	
Bit 1	OpMode_Bit1	
Bit 2	OpMode_Bit2	
Bit 3	OpMode_Bit3	
Bit 4	ManJogPos	
Bit 5	ManJogNeg	
Bit 6	ManExecute2ndSpeed	
Bit 7	ReleaseLimitSwitch	
Bit 8	HomStartStop	
Bit 9	HomSetPos	
Bit 10	HomResetPos	
Bit 11	EnableSpeedOverride	
Bit 12	EnableAccOverride	
Bit 13	EnableSRampOverride	
Bit 14	PosTeachSetPos	
Bit 15	PosTeachActPos	
Bit 16	PosExecute	
Bit 17	PosFinishTarget	
Bit 18	PosDisableFollowProfile	
Bit 19	PosStop	
Bit 20	PosModeBit0	
Bit 21	PosModeBit1	
Bit 22	PosModeBit2	
Bit 23	PosModeBit3	
Bit 24	ProfileNo_Bit0	
Bit 25	ProfileNo_Bit1	
Bit 26	ProfileNo_Bit2	
Bit 27	ProfileNo_Bit3	
Bit 28	ProfileNo_Bit4	
Bit 29	ProfileNo_Bit5	
Bit 30	ProfileNo_Bit6	
Bit 31	ProfileNo_Bit7	
Subcodes	**Lenze setting**	**Info**
C00463/1	0x00000000	Remote: MCK control
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT		

C00464

Parameter | Name: **C00464 | Remote: Monitoring timeout**			Data type: UNSIGNED_16 Index: 24111_d = $5E2F_h$
			▶ PC manual control
Setting range (min. value | unit | max. value)			
200	ms	5000	
Subcodes	**Lenze setting**		**Info**
C00464/1	2000 ms		Remote: Monitoring timeout
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00465

Parameter | Name: **C00465 | Keypad: Timeout welcome screen**		Data type: INTEGER_32 Index: 24110_d = $5E2E_h$
Time setting for the automatic change of the keypad display to the welcome screen		
Selection list(Lenze setting printed in bold)		
0	**Never show welcome screen**	
5	5 min	
15	15 min	
30	30 min	
60	60 min	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00466

Parameter | Name: **C00466 | Keypad: Default parameter**			Data type: INTEGER_32 Index: 24109_d = $5E2D_h$
Setting of the default parameter for the keypad			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**51**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00467

Parameter | Name: **C00467 | Keypad: Default welcome screen**		Data type: INTEGER_32 Index: 24108_d = $5E2C_h$
Selection of the welcome screen for the keypad		
Selection list(Lenze setting printed in bold)		
0	Main menu	
1	**Parameter list**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00468

Parameter | Name: **C00468 | Service code**	Data type: INTEGER_32 Index: 24107_d = $5E2B_h$
This code is used device-internally and must not be written by the user side!	

C00469

Parameter | Name: **C00469 | Keypad: Fct. STOP key**		Data type: INTEGER_32 Index: 24106_d = $5E2A_h$
Selection of the function for the STOP key on the keypad		
Selection list(Lenze setting printed in bold)		**Info**
0	No function	STOP key does not have any function
1	**Inhibit inverter**	STOP key sets controller inhibit in the drive
2	Activate quick stop	STOP key sets quick stop in the drive
4	Inhibit inverter and reset error	From version 14.00.00 STOP key sets controller inhibit in the drive. An error reset is carried out at the same time.
5	Activate quick stop and reset error	From version 14.00.00 STOP key sets quick stop in the drive. An error reset is carried out at the same time.
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00470

Parameter | Name: **C00470 | LS_ParFree_b**		Data type: UNSIGNED_8 Index: 24105_d = $5E29_h$
The LS_ParFree_b SB: Setting of the signal level to be output		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C00470/1	0: FALSE	Signal level for output *bPar1 ... bPar32*
C00470/...		
C00470/32		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00471

Parameter | Name: **C00471 | LS_ParFree**			Data type: UNSIGNED_16 Index: 24104_d = $5E28_h$
The LS_ParFree SB: Setting of the words to be output			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Bit0		
...	...		
Bit 15	Bit15		
Subcodes	**Lenze setting**		**Info**
C00471/1	0x0000		Value for output *wPar1 ... wPar32*
C00471/...			
C00471/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00472

Parameter | Name: **C00472 | LS_ParFree_a**			Data type: INTEGER_16 Index: 24103_d = $5E27_h$
The LS_ParFree_a SB: Setting of the analog signals to be output			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00472/1	0.00 %		Value for output *nPar1_a*
C00472/2	0.00 %		Value for output *nPar2_a*
C00472/3	100.00 %		Value for output *nPar3_a*
C00472/4	100.00 %		Value for output *nPar4_a*
C00472/5	0.00 %		Value for output *nPar5_a*
C00472/6	0.00 %		Value for output *nPar6_a*
C00472/7	0.00 %		Value for output *nPar7_a*
C00472/8	0.00 %		Value for output *nPar8_a*
C00472/9	0.00 %		Value for output *nPar9_a*
C00472/10	0.00 %		Value for output *nPar10_a*
C00472/11	0.00 %		Value for output *nPar11_a*
C00472/12	0.00 %		Value for output *nPar12_a*
C00472/13	0.00 %		Value for output *nPar13_a*
C00472/14	0.00 %		Value for output *nPar14_a*
C00472/15	0.00 %		Value for output *nPar15_a*
C00472/16	0.00 %		Value for output *nPar16_a*
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00473

Parameter | Name: **C00473 | LS_ParFree_v**			Data type: INTEGER_16 Index: 24102_d = $5E26_h$
The LS_ParFree_v SB: Setting of the speed signals to be output			
Setting range (min. value | unit | max. value)			
-32767	Incr./ms	32767	
Subcodes	**Lenze setting**		**Info**
C00473/1	0 incr./ms		Values for output *nPar1_v* ... *nPar8_v*
C00473/...			
C00473/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00474

Parameter | Name: **C00474 | LS_ParFree_p**			Data type: INTEGER_32 Index: 24101d = 5E25h
SB LS_ParFree_p: Setting of the position signals to be output			
Setting range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes	**Lenze setting**		**Info**
C00474/1	0 incr.		Values for output *dnPar1_p* ... *dnPar8_p*
C00474/...			
C00474/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00475

Parameter | Name: **C00475 | LS_ParFreeUnit_1_2**			Data type: INTEGER_32 Index: 24100d = 5E24h
From version 02.00.00 SB LS_ParFreeUnit / LS_ParFreeUnit_2: Setting of the position signals to be output			
Setting range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes	**Lenze setting**		**Info**
C00475/1	0.0000 units		Values for output *dnC475_1* ... *dnC475_32*
C00475/...			
C00475/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C00476

Parameter | Name: **C00476 | LS_ParFree_a_2**			Data type: INTEGER_16 Index: 24099_d = $5E23_h$
From version 02.00.00 SB LS_ParFree_a_2: Setting of the analog signals to be output			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00476/1	0.00 %		Value for output *nC476_1_a*
C00476/2	0.00 %		Value for output *nC476_2_a*
C00476/3	0.00 %		Value for output *nC476_3_a*
C00476/4	0.00 %		Value for output *nC476_4_a*
C00476/5	0.00 %		Value for output *nC476_5_a*
C00476/6	0.00 %		Value for output *nC476_6_a*
C00476/7	0.00 %		Value for output *nC476_7_a*
C00476/8	0.00 %		Value for output *nC476_8_a*
C00476/9	0.00 %		Value for output *nC476_9_a*
C00476/10	0.00 %		Value for output *nC476_10_a*
C00476/11	0.00 %		Value for output *nC476_11_a*
C00476/12	0.00 %		Value for output *nC476_12_a*
C00476/13	0.00 %		Value for output *nC476_13_a*
C00476/14	0.00 %		Value for output *nC476_14_a*
C00476/15	0.00 %		Value for output *nC476_15_a*
C00476/16	0.00 %		Value for output *nC476_16_a*
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00477

Parameter | Name: **C00477 | LS_ParFree_2**			Data type: UNSIGNED_16 Index: 24098_d = $5E22_h$
From version 02.00.00 SB LS_ParFree_2: Setting of the words to be output			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Bit0		
...	...		
Bit 15	Bit15		
Subcodes	**Lenze setting**		**Info**
C00477/1	0x0000		Value for output *wC477_1 ... wC477_32*
C00477/...			
C00477/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00478

Parameter | Name: **C00478 | LS_ParFree_v_2**			Data type: INTEGER_16 Index: 24097_d = $5E21_h$
From version 02.00.00 SB LS_ParFree_v_2: Setting of the speed signals to be output			
Setting range (min. value | unit | max. value)			
-32767	Incr./ms	32767	
Subcodes	**Lenze setting**		**Info**
C00478/1	0 incr./ms		Values for output *nC478_1_v ... nC478_8_v*
C00478/...			
C00478/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00479

Parameter | Name: **C00479 | LS_ParFree32**			Data type: INTEGER_32 Index: 24096_d = $5E20_h$
From version 02.00.00 SB LS_ParFree32: Setting of the 32-bit values to be output			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C00479/1	0		Values for output *dnC479_1 ... dnC479_8*
C00479/...			
C00479/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00480

Parameter | Name: **C00480 | LS_DisFree_b**			Data type: UNSIGNED_16 Index: 24095_d = $5E1F_h$
The LS_DisFree_b SB: Display of the input values			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	bDis1		Signal level input *bDis1 ... bDis16*
...	...		
Bit 15	bDis16		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00481

Parameter | Name: **C00481 | LS_DisFree**			Data type: UNSIGNED_16 Index: 24094_d = $5E1E_h$
The LS_DisFree SB: Display of the input values			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
...	...		
Bit 15	Active		
Subcodes			**Info**
C00481/1			Input values *wDis1* ... *wDis8*
C00481/...			
C00481/8			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00482

Parameter | Name: **C00482 | LS_DisFree_a**			Data type: INTEGER_16 Index: 24093_d = $5E1D_h$
The LS_DisFree_a SB: Display of the input values			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00482/1			Input values *nDis1_a* ... *nDis8_a*
C00482/...			
C00482/8			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00483

Parameter | Name: **C00483 | LS_DisFree_p**			Data type: INTEGER_32 Index: 24092_d = $5E1C_h$
SB LS_DisFree_p: Display of the input values			
Display range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes			**Info**
C00483/1			Input values *dnDis1_p* ... *dnDis8_p*
C00483/...			
C00483/8			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00484

Parameter | Name: **C00484 | Application units: Offset**			Data type: INTEGER_16 Index: 24091_d = $5E1B_h$
The LS_DisFree_a SB: Offset for display of the input variables in application unit ▸ Display of internal process factors in application units			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00484/1	0.00 %		Application unit 1: Offset
C00484/2	0.00 %		Application unit 2: Offset
C00484/3	0.00 %		Application unit 3: Offset
C00484/4	0.00 %		Application unit 4: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00485

Parameter | Name: **C00485 | Application units: Display factor**			Data type: INTEGER_32 Index: 24090_d = $5E1A_h$
The LS_DisFree_a SB: Display factor for display of the input variables in application unit ▸ Display of internal process factors in application units			
Setting range (min. value | unit | max. value)			
-65536.0000		65536.0000	
Subcodes	**Lenze setting**		**Info**
C00485/1	1.0000		Application unit 1: Display factor
C00485/2	1.0000		Application unit 2: Display factor
C00485/3	1.0000		Application unit 3: Display factor
C00485/4	1.0000		Application unit 4: Display factor
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C00486

Parameter | Name: **C00486 | Application units: Text**		Data type: VISIBLE_STRING Index: 24089_d = $5E19_h$
The LS_DisFree_a SB: Text for the display of the input variables in application unit ▸ Display of internal process factors in application units		
Subcodes	**Lenze setting**	**Info**
C00486/1		Application unit 1: Text
C00486/2		Application unit 2: Text
C00486/3		Application unit 3: Text
C00486/4		Application unit 4: Text
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ COM ☐ MOT Character length: 7		

C00487

Parameter | Name: **C00487 - Application units**			Data type: INTEGER_32 Index: 24088_d = $5E18_h$
SB LS_DisFree_a: Display of the input values in a configurable application unit ▸ Display of internal process factors in application units			
Display range (min. value | unit | max. value)			
-21474836.47	units	21474836.47	
Subcodes			**Info**
C00487/1			Application units 1
C00487/2			Application units 2
C00487/3			Application units 3
C00487/4			Application units 4
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00488

Parameter | Name: **C00488 | L_JogCtrlEdgeDetect**		Data type: UNSIGNED_8 Index: 24087_d = $5E17_h$
The L_JogCtrlExtension_1 FB: Signal methodology • Selection whether the corresponding function is to be activated by edge or level.		
Selection list		
0	Level	
1	Edge	
Subcodes	**Lenze setting**	**Info**
C00488/1	0: Level	InputSens.SlowDown1 • Selection of edge or level for starting slow-down function 1
C00488/2	0: Level	InputSens.Stop1 • Selection of edge or level for stop function 1
C00488/3	0: Level	InputSens.SlowDown2 • Selection of edge or level for starting slow-down function 2
C00488/4	0: Level	InputSens.Stop2 • Selection of edge or level for stop function 2
C00488/5	0: Level	InputSens.SlowDown3 • Selection of edge or level for starting slow-down function 3
C00488/6	0: Level	InputSens.Stop3 • Selection of edge or level for stop function 3
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00490

Parameter | Name: **C00490 | Position encoder selection**			Data type: UNSIGNED_8 Index: 24085$_d$ = 5E15$_h$
Selection of the feedback system for the generation of the actual position on the load ▸ Encoder/feedback system			
Selection list(Lenze setting printed in bold)		**Info**	
0	**No sensor: nSpeedSetValue_a**	No encoder available on the load. The position signal *dnMotorPosAct_p* is derived from the speed setpoint *nSpeedSetValue_a*.	
1	Sensor signal FreqIn12	Position encoder signal is fed via digital inputs DI1 and DI2	
2	Encoder signal FreqIn67	Position encoder signal is fed via digital inputs DI6 and DI7	
3	Multi encoder	Position encoder signal is supplied via multi-encoder interface	
4	Resolver	Position encoder signal is supplied via resolver interface	
5	Encoder signal FreqIn1267	From version 15.00.00 Position encoder signal is fed via digital inputs DI1/DI2 and DI6/DI7 (connection of 4-track HTL encoders)	
10	Encoder selection C495 or nVoltageAngleAct_a	From version 12.00.00 The position signal *dnMotorPosAct_p* is either calculated from the set speed feedback (for C00495 > 0) or from the speed signal *nMotorSpeedAct_v* (for C00495 = 0).	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00491

Parameter | Name: **C00491 | Hiperface messages**			Data type: UNSIGNED_8 Index: 24084$_d$ = 5E14$_h$
Ab Version 14.00.00			
Display range (min. value | unit | max. value)			
0		255	
Subcodes		**Information**	
C00491/1		Hiperface: Status 1	
C00491/2		Hiperface: Status 2	
C00491/3		Hiperface: Status 3	
C00491/4		Hiperface: Status 4	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00492

Parameter | Name: **C00492 | Hiperface: Detected TypCode**			Data type: UNSIGNED_8 Index: 24083$_d$ = 5E13$_h$
From version 02.00.00 ▸ Encoder/feedback system: sin/cos absolute value encoder with HIPERFACE® protocol			
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00493

Parameter | Name: **C00493 | Hiperface: TypCode**			Data type: UNSIGNED_8 Index: 24082_d = $5E12_h$
From version 02.00.00			
			▸ Encoder/feedback system: sin/cos absolute value encoder with HIPERFACE® protocol
Setting range (min. value | unit | max. value)			**Lenze setting**
0		255	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00494

Parameter | Name: **C00494 | Hiperface: Resolutions**			Data type: UNSIGNED_32 Index: 24081_d = $5E11_h$
From version 02.00.00			
			▸ Encoder/feedback system: sin/cos absolute value encoder with HIPERFACE® protocol
Setting range (min. value | unit | max. value)			
0		65536	
Subcodes	**Lenze setting**		**Information**
C00494/1	0		Hiperface: Number of revolutions
C00494/2	0		Hiperface: Steps per revolution
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00495

Parameter | Name: **C00495 | Speed sensor selection**		Data type: UNSIGNED_8 Index: 24080_d = $5E10_h$
Selection of the feedback system for the actual speed for motor control and display		
		▸ Encoder/feedback system
Selection list(Lenze setting printed in bold)		**Info**
0	**No sensor**	No sensor available for the actual speed detection
1	Sensor signal FreqIn12	Speed sensor signal is fed via the digital DI1 and DI2 inputs
2	Encoder signal FreqIn67	Speed encoder signal is fed via digital inputs DI6 and DI7
3	Multi encoder	Speed encoder signal is supplied via multi-encoder interface
4	Resolver	Speed encoder signal is supplied via resolver interface
5	Encoder signal FreqIn1267	From version 15.00.00 Speed encoder signal is fed via digital inputs DI1/DI2 and DI6/DI7 (connection of 4-track HTL encoders)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00496

Parameter | Name: C00496 | Encoder evaluation method DigIn12			Data type: UNSIGNED_8 Index: 24079_d = $5E0F_h$
			▸ Encoder/feedback system
Selection list (Lenze setting printed in bold)			**Info**
0	High-resolution encoders		High-precision procedure for high-resolution encoders (>=512 increments)
1	Low-resolution encoder (StateLine)		High-precision procedure for low-resolution encoders (<=128 increments)
2	**Comb. encoder procedure**		Combination of the first two procedures as a function of the speed (recommended procedure)
3	Edge-counting procedure		Simple edge counting procedure with adjustable scanning time (C00425)
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00497

Parameter | Name: C00497 | Filter time constant			Data type: UNSIGNED_16 Index: 24078_d = $5E0E_h$
			▸ Encoder/feedback system
Setting range (min. value | unit | max. value)			
0.0	ms	500.0	
Subcodes	**Lenze setting**		**Info**
C00497/1	1.0 ms		FreqIn12: Encoder filter time
C00497/2	1.0 ms		FreqIn67: Encoder filter time
C00497/3	1.0 ms		LS_MultiEncoder: Encoder filter time
C00497/4	2.0 ms		LS_Resolver: Encoder filter time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00498

Parameter | Name: **C00498 | Open-circuit monitoring**		Data type: UNSIGNED_8 Index: 24077_d = $5E0D_h$
Configuration of open-circuit monitoring for encoder/feedback system • We recommend using the Lenze setting. • The settings 1 ... 3 make sense if the respective encoder is not to be used as speed encoder or position encoder but within the scope of the application. Moreover, a directed disconnection of the monitoring is possible. ▸ Encoder/feedback system		
Selection list		**Info**
0	Speed and position encoder	Open-circuit monitoring is active for the speed encoder selected in C00495 and the position encoder selected in C00490. • Depending on whether a speed encoder or position encoder has been selected, the monitoring mode for the resolver and/or encoder is activated. • If no speed encoder and position encoder have been selected, open-circuit monitoring is deactivated.
1	Resolver only	Open-circuit monitoring is only active for the resolver, independent on the selection of the speed encoder and position encoder. • Open-circuit monitoring for the encoder is deactivated.
2	Encoder only	Open-circuit monitoring is only active for the encoder, independent on the selection of the speed encoder and position encoder. • Open-circuit monitoring for the resolver is deactivated.
3	Resolver and encoder	Open-circuit monitoring is active for the resolver and encoder, independent on the selection of the speed encoder and position encoder.
Subcodes	**Lenze setting**	**Information**
C00498/1	0: Speed encoder and position encoder	Open-circuit monitoring
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00499

Parameter | Name: **C00499 | Hiperface: Settings**			Data type: UNSIGNED_16 Index: 24076_d = $5E0C_h$
Ab Version 14.00.00			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Manual entry of encoder data		
Bit 1	write defined position		
Bit 2	Also write value directly in case of PSM		
Bit 3	PLI compensation		
Bit 4	Delay open circuit monit.		From version 15.00.00
Bit 5	High resolution deactivated		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Information**
C00499/1	0x0000		Hiperface: Advanced settings
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C00505

Parameter | Name: **C00505 | Password data**		Data type: VISIBLE_STRING Index: 24070_d = $5E06_h$
		▸ Device access protection
Subcodes	**Lenze setting**	**Info**
C00505/1		MasterPin
C00505/2		Binding ID
C00505/3		Password
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ COM ☐ MOT Character length: 16		

C00506

Parameter | Name: **C00506 | PW protection internal config**			Data type: UNSIGNED_16 Index: 24069_d = $5E05_h$
From version 15.00.00 Configuration of the password protection for device-internal parameter access ▶ Individual password protection for single communication channels			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Only access to user menu		The menus in the keypad are reduced. Only the user menu can be used to change parameters.
Bit 1	Parameter write protection		Via system blocks (LS_ParReadWrite1-6 and LS_WriteParamList), write/read parameters cannot be written.
Bit 2	Parameter read protection		Via system blocks (LS_ParReadWrite1-6 and LS_WriteParamList), write/read parameters cannot be read.
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C00506/1	0x0007		PW protection internal config
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00507

Parameter | Name: **C00507 | Current password protection**			Data type: UNSIGNED_16 Index: 24068_d = $5E04_h$
Display of the currently active device access protection (password protection and device personalisation) **Note:** As the password protection can be configured individually for each single communication channel from version 15.00.00 bit 1 and bit 2 indicate the active protection always with regard to the communication channel used. ▸ Device access protection			
Display area **(min. hex value | max. hex value)**			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Only access to user menu		1 ≡The menus in the keypad are reduced. Only the user menu can be used to change parameters. ▸ Password protection
Bit 1	Parameter write protection		1 ≡ The currently used communication channel cannot be used to write into write/read parameters. ▸ Password protection
Bit 2	Parameter read protection		1 ≡ The currently used communication channel cannot be used to read write/read parameters. ▸ Password protection
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Memory module binding on		1 ≡ Device personalisation is active. All write/read actions between inverter and memory module are only executed if both components have the same binding ID. ▸ Device personalisation
Subcodes			**Info**
C00507/1			Password protection - all communication channels
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00508

Parameter | Name: **C00508 | PW protection config X6(DIAG)**			Data type: UNSIGNED_16 Index: 24067_d = $5E03_h$
From version 15.00.00 Configuration of the password protection for parameter access via diagnostic interface X6 ("DIAG") ▸ Individual password protection for single communication channels			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Reserved		
Bit 1	Parameter write protection		The diagnostics interface X6 cannot be used to write into write/read parameters.
Bit 2	Parameter read protection		The diagnostics interface X6 cannot be used to read write/read parameters.
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C00508/1	0x0006		PW protection config X6(DIAG)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00509

Parameter | Name: **C00509 | PW protection config X1(CAN)**			Data type: UNSIGNED_16 Index: 24066_d = $5E02_h$
From version 15.00.00 Configuration of the password protection for parameter access via CANopen interface X1 ("CAN on board") ▸ Individual password protection for single communication channels			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:		**Info**	
Bit 0	Reserved		
Bit 1	Parameter write protection	The CANopen interface X1 cannot be used to write into write/read parameters.	
Bit 2	Parameter read protection	The CANopen interface X1 cannot be used to read write/read parameters.	
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**	**Info**	
C00509/1	0x0006	PW protection config X1 (CAN)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00510

Parameter | Name: **C00510 | PW protection config MCI**			Data type: UNSIGNED_16 Index: 24065_d = $5E01_h$
From version 15.00.00 Configuration of the password protection for parameter access via fieldbus interface ("MCI") ▶ Individual password protection for single communication channels			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Reserved		
Bit 1	Parameter write protection		The fieldbus interface (MCI) cannot be used to write into write/read parameters.
Bit 2	Parameter read protection		The fieldbus interface (MCI) cannot be used to read write/read parameters.
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C00510/1	0x0006		PW protection config. MCI
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00516

Parameter | Name: **C00516 | Checksums**	Data type: UNSIGNED_32 Index: 24059_d = $5DFB_h$
This code is used device-internally and must not be written by the user side!	

C00517

Parameter | Name: **C00517 | User menu**			Data type: INTEGER_32 Index: 24058_d = $5DFA_h$
When a system is installed, parameters must be changed time and again until the system runs satisfactorily. The user menu of a device serves to create a selection of frequently used parameters to be able to access and change these parameters quickly. • Format: <code number>,<subcode number> • If "0.000" is set, no entry will be displayed in the user menu.			
Setting range (min. value | unit | max. value)			
0.000		16000.000	
Subcodes	**Lenze setting**		**Info**
C00517/1	51.000		C00051: Display of actual speed value
C00517/2	53.000		C00053: Display of DC-bus voltage
C00517/3	54.000		C00054: Display of motor current
C00517/4	61.000		C00061: Display of heatsink temperature
C00517/5	137.000		C00137: Display of device status
C00517/6	166.003		C00166/3: Display of current error message
C00517/7	0.000		User menu: Entry 7
C00517/8	11.000		C00011: Reference speed
C00517/9	39.001		C00039/1: Fixed setpoint 1
C00517/10	39.002		C00039/2: Fixed setpoint 2
C00517/11	12.000		C00012: Accel. time - main setpoint
C00517/12	13.000		C00013: Decel. time - main setpoint
C00517/13	15.000		C00015: V/f base frequency
C00517/14	16.000		C00016: Vmin boost
C00517/15	22.000		C00022: Imax in motor mode
C00517/16	120.000		C00120: Setting of motor overload (I^2xt)
C00517/17	87.000		C00087: Rated motor speed
C00517/18	99.000		C00099: Display of firmware version
C00517/19	200.000		C00200: Display of firmware product type
C00517/20	0.000		User menu: Entry 20
C00517/21	0.000		User menu: Entry 21
C00517/22	0.000		User menu: Entry 22
C00517/23	0.000		User menu: Entry 23
C00517/24	105.000		C00105: Decel. time - quick stop
C00517/25	173.000		C00173: Mains voltage
C00517/26	0.000		User menu: Entry 26
C00517/27	0.000		User menu: Entry 27
C00517/28	0.000		User menu: Entry 28
C00517/29	0.000		User menu: Entry 29
C00517/30	0.000		User menu: Entry 30
C00517/31	0.000		User menu: Entry 31
C00517/32	0.000		User menu: Entry 32
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00540

Parameter | Name: **C00540 | LS_DFOut: Function**		Data type: UNSIGNED_8 Index: 24035_d = $5DE3_h$
From version 12.00.00 Selection of the signal source for the digital frequency output (multi-encoder interface X8)		
		Digital frequency coupling
Selection list(Lenze setting printed in bold)		
1	Off	
2	**DFIn**	
3	MotorSpeed	
4	LoadSpeed	
5	Resolver	
6	DigIn 1/2	
7	FB input	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00545

Parameter | Name: **C00545 | LS_DFOut: Angular offset**			Data type: INTEGER_32 Index: 24030_d = $5DDE_h$
From version 12.00.00			
			Digital frequency coupling
Setting range (min. value | unit | max. value)			
0	Incr.	65535	
Subcodes	**Lenze setting**		**Information**
C00545/1	0 incr.		LS_DFOut: Angular offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00560

Parameter | Name: **C00560 | Fan switching status**		Data type: UNSIGNED_8 Index: 24015_d = $5DCF_h$
Display of the function status of the device fans		
Selection list		
0	Off	
1	On	
2	No fan	
Subcodes		**Info**
C00560/1		Switching status - internal fan
C00560/2		Switching status - heatsink fan
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00561

Parameter | Name: **C00561 | Failure indication**		Data type: UNSIGNED_8 Index: 24014$_d$ = 5DCE$_h$
Failure display of device fans and motor phases		
Selection list		
0	No error	
1	Error	
Subcodes		**Info**
C00561/1		Internal fan
C00561/2		Heatsink fan
C00561/3		Motor phase U • From version 02.00.00
C00561/4		Motor phase V • From version 02.00.00
C00561/5		Motor phase W • From version 02.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00563

Parameter | Name: **C00563 | Current monitoring: Delay time**			Data type: UNSIGNED_32 Index: 24012$_d$ = 5DCC$_h$
From version 16.00.00			▸ Current monitoring overload
Setting range (min. value | unit | max. value)			
0.000	s	999.900	
Subcodes	**Lenze setting**		**Info**
C00563/1	3.000 s		Current monitoring: Delay time overload
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00565

Parameter | Name: **C00565 | Resp. to mains phase failure**		Data type: UNSIGNED_8 Index: 24010$_d$ = 5DCA$_h$
Response to the failure of mains phases		
Selection list (Lenze setting printed in bold)		
0	No Reaction	
1	Fault	
5	**Warning**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00566

Parameter | Name: **C00566 | Resp. to fan failure**		Data type: UNSIGNED_8 Index: 24009_d = $5DC9_h$
Response to the detection of a fan failure		
Selection list(Lenze setting printed in bold)		
0	No Reaction	
1	Fault	
5	**Warning**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00567

Parameter | Name: **C00567 | Resp. to speed controller limited**		Data type: UNSIGNED_8 Index: 24008_d = $5DC8_h$
Response if speed controller output is limited (*bLimSpeedCtrlOut* = TRUE)		
Selection list(Lenze setting printed in bold)		
0	**No Reaction**	
1	Fault	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00569

Parameter | Name: **C00569 | Resp. to peak current**		Data type: UNSIGNED_8 Index: 24006_d = $5DC6_h$
Configuration of monitoring of the motor control (group 1)		
Selection list		
0	No Reaction	
1	Fault	
5	Warning	
Subcodes	**Lenze setting**	**Info**
C00569/1	0: No Reaction	Response for overcurrent detection and clamp operation
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00570

Parameter | Name: **C00570 | Resp. to controller limitations**		Data type: UNSIGNED_8 Index: 24005_d = $5DC5_h$
Configuration of monitoring of the motor control (group 2)		
Selection list		
0	No Reaction	
1	Fault	
5	Warning	
Subcodes	**Lenze setting**	**Info**
C00570/1	0: No Reaction	Response if direct-axis current controller is limited • e.g. at servo control (SC)
C00570/2	0: No Reaction	Response if cross current controller is limited • e.g. at servo control (SC)
C00570/3	0: No Reaction	Response if torque setpoint is limited • Limitation of the speed controller output, the differential setpoint precontrol, and of the additive torque at (SC) servo control and (SLVC) sensorless vector control.
C00570/4	0: No Reaction	Response if field controller is limited
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00571

Parameter | Name: **C00571 | Resp. to wrong controller setting**		Data type: UNSIGNED_8 Index: 24004_d = $5DC4_h$
From version 14.00.00		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00571/1	6: Information	Resp. to: Motor control does not match motor data
C00571/2	1: Fault	Resp. to a non-set speed controller
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00572

Parameter | Name: **C00572 | Brake resistor overload threshold**			Data type: UNSIGNED_8 Index: 24003_d = $5DC3_h$
Adjustable threshold for monitoring the brake resistor utilisation • The response for reaching the threshold can be selected in C00574.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00574

Parameter | Name: **C00574 | Resp. to brake resist. overtemp.**		Data type: UNSIGNED_8 Index: 24001_d = $5DC1_h$
Response which is triggered if the threshold set in C00572 for monitoring brake resistor utilisation is reached.		
Selection list(Lenze setting printed in bold)		
0	**No Reaction**	
1	Fault	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00575

Parameter | Name: **C00575 | Resp. to logbook information**	Data type: UNSIGNED_8 Index: 24000_d = $5DC0_h$
This code is used device-internally and must not be written by the user side!	

C00576

Parameter | Name: **C00576 | SC: Optimisation of field feedforward control**			Data type: UNSIGNED_16 Index: 23999_d = $5DBF_h$
Point of action of the direct-axis current setpoint precontrol for an early reduction of the field current. In this way, the acceleration behaviour can be improved in the field weakening range. • The entry [%] refers to the slip of the asynchronous motor.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	600	**200 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00577

Parameter | Name: **C00577 | SC: Vp field weakening controller**			Data type: UNSIGNED_16 Index: 23998_d = $5DBE_h$
Proportional gain of the field weakening controller • When "0" is set, the P component of the controller is deactivated. • The recommended setting is in the range between 0 and 0.0020			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000		2.0000	**0.0010**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C00578

Parameter | Name: **C00578 | SC: Tn field weakening controller**			Data type: UNSIGNED_16 Index: 23997_d = $5DBD_h$
Time constant of the field weakening controller • The recommended setting is in the range between 10 and 30 ms			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.1	ms	6200.0	**20.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C00579

Parameter | Name: **C00579 | Resp. to max. speed/output freq. reached**		Data type: UNSIGNED_8 Index: 23996_d = $5DBC_h$
Response when the max. speed limit (C00909) or output frequency limit (C00910) has been reached.		
Selection list(Lenze setting printed in bold)		
0	**No Reaction**	
1	Fault	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00580

Parameter | Name: **C00580 | Resp. to operating system error**		Data type: UNSIGNED_8 Index: 23995_d = $5DBB_h$
From version 02.00.00 Response if the required computing time of the application exceeds the available computing time.		
Selection list		
0	No Reaction	
1	Fault	
5	Warning	
Subcodes	**Lenze setting**	**Info**
C00580/1	0: No Reaction	Resp. to runtime exceedance
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00581

Parameter | Name: **C00581 | Resp. to LS_SetError_x**		Data type: UNSIGNED_8 Index: 23994_d = $5DBA_h$
Selection of the error responses for application error messages • An application error message is tripped by a FALSE/TRUE edge at the binary inputs *bSetError1...4*.		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00581/1	0: No Reaction	LS_SetError_1: Resp. to bSetError1
C00581/2	0: No Reaction	LS_SetError_1: Resp. to bSetError2
C00581/3	0: No Reaction	LS_SetError_1: Resp. to bSetError3
C00581/4	0: No Reaction	LS_SetError_1: Resp. to bSetError4
C00581/5	0: No Reaction	LS_SetError_2: Resp. to bSetError1
C00581/6	0: No Reaction	LS_SetError_2: Resp. to bSetError2
C00581/7	0: No Reaction	LS_SetError_2: Resp. to bSetError3
C00581/8	0: No Reaction	LS_SetError_2: Resp. to bSetError4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00582

Parameter | Name: **C00582 | Resp. to heatsink temp. > shutdown temp. -5°C**		Data type: UNSIGNED_8 Index: 23993_d = $5DB9_h$
Response if the heatsink temperature has reached the switch-off temperature threshold.		
Selection list(Lenze setting printed in bold)		
0	**No Reaction**	
1	Fault	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00583

Parameter | Name: **C00583 | Resp. to motor temperature KTY**		Data type: UNSIGNED_8 Index: 23992_d = $5DB8_h$
Response to motor overtemperature • The motor temperature is measured via the resolver or encoder cable. ▸ Encoder/feedback system: Motor temperature monitoring (KTY)		
Selection list		
0	No Reaction	
1	Fault	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Information**
C00583/1	1: Fault	Resp. to motor overtemp. KTY resolver
C00583/2	1: Fault	Resp. to motor overtemp. KTY MultiEncoder
C00583/3	5: Warning	Resp. to motor temp. > C00121 resolver
C00583/4	5: Warning	Resp. to motor temp. > C00121 encoder
C00583/5	1: Fault	Resp. to temp. sensor error KTY resolver
C00583/6	1: Fault	Resp. to temp. sensor error KTY MultiEncoder
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00584

Parameter | Name: **C00584 | Resp. to current monitoring**		Data type: UNSIGNED_8 Index: 23991_d = $5DB7_h$
From version 16.00.00 Response in the event of current overload Current monitoring overload		
Selection list		**Info**
0	No Reaction	
1	Fault	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00584/1	0: No Reaction	Resp. to current monitoring overload
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00585

Parameter | Name: **C00585 | Resp. to motor overtemp. PTC**		Data type: UNSIGNED_8 Index: 23990_d = $5DB6_h$
Response to motor overtemperature • The motor temperature is measured by means of a PTC thermistor at terminal X106.		
Selection list(Lenze setting printed in bold)		
0	No Reaction	
1	**Fault**	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00586

Parameter | Name: **C00586 | Resp. to open circuit HTL encoder**		Data type: UNSIGNED_8 Index: 23989_d = $5DB5_h$
Response to encoder feedback system failure or encoder feedback system track failure due to open circuit		
Selection list(Lenze setting printed in bold)		
0	No Reaction	
1	**Fault**	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00588

Parameter | Name: **C00588 | Resp. to max. speed at switching freq.**		Data type: UNSIGNED_8 Index: 23987_d = $5DB3_h$
Response if the maximum speed for the set inverter switching frequency is reached (C00018)		
Selection list(Lenze setting printed in bold)		
0	**No Reaction**	
1	Fault	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00590

Parameter | Name: **C00590 | Resp. to switching frequency red.**		Data type: UNSIGNED_8 Index: 23985_d = $5DB1_h$
Response to reduction of the inverter switching frequency (C00018)		
Selection list(Lenze setting printed in bold)		
0	**No Reaction**	
1	Fault	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00591

Parameter | Name: **C00591 | Resp. to axis bus error**		Data type: UNSIGNED_8 Index: 23984_d = $5DB0_h$
From version 12.00.00 Configuration of axis bus monitoring ▶ Axis bus		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Information**
C00591/1	0: No Reaction	Resp. to axis bus data error ▶ Data transfer axis bus
C00591/2	0: No Reaction	Resp. to axis bus IO error ▶ IO axis bus
C00591/3	0: No Reaction	Resp. to axis bus IO request
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00592

Parameter | Name: **C00592 | Resp. to CAN bus connection**		Data type: UNSIGNED_8 Index: 23983_d = $5DAF_h$
Configuration of monitoring of the CAN interface (group 1) "CAN on board" system bus		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00592/1	0: No Reaction	Response to incorrect telegram for CAN communication
C00592/2	0: No Reaction	Response to "BusOff" (bus system switched off)
C00592/3	0: No Reaction	Response to warnings of the CAN controller
C00592/4	0: No Reaction	Response to communication stop of a CAN bus node
C00592/5	0: No Reaction	Response to an event in the case of monitoring via heartbeat protocol
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00593

Parameter | Name: **C00593 | Resp. to CANx_IN monitoring**		Data type: UNSIGNED_8 Index: 23982_d = $5DAE_h$
Configuration of monitoring of the CAN interface (group 2) "CAN on board" system bus		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00593/1	0: No Reaction	Response if the monitoring time set in C00357/1 for the reception of the PDO CAN1_IN is exceeded.
C00593/2	0: No Reaction	Response if the monitoring time set in C00357/2 for the reception of the PDO CAN2_IN is exceeded.
C00593/3	0: No Reaction	Response if the monitoring time set in C00357/3 for the reception of the PDO CAN3_IN is exceeded.
C00593/4	0: No Reaction	Response if the monitoring time set in C00357/4 for the reception of the PDO CAN4_IN is exceeded. • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C00594

Parameter | Name: **C00594 | Resp. to control word error**		Data type: UNSIGNED_8 Index: 23981_d = $5DAD_h$
Configuration of device control monitoring		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
5	Warning	
Subcodes	**Lenze setting**	**Info**
C00594/1	1: Fault	Response if error bit 14 in the CAN control word is set.
C00594/2	1: Fault	Response if error bit 14 in the MCI control word is set.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00595

Parameter | Name: **C00595 | MCK: Resp. to MCK error**		Data type: UNSIGNED_8 Index: 23980_d = $5DAC_h$
Configuration of monitoring of the Motion Control Kernel ▸ Basic drive functions		
Selection list		
0	No Reaction	
1	Fault	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00595/1	3: TroubleQuickStop	Response if the input *bLimitSwitchPos* for travel range monitoring is set to FALSE (fail-safe) by the positive hardware limit switch .
C00595/2	3: TroubleQuickStop	Response if the input *bLimitSwitchNeg* for travel range monitoring is set to FALSE (fail-safe) by the negative hardware limit switch .
C00595/3	3: TroubleQuickStop	Response for detection that the position is beyond the positive software limit position (C01229/1).
C00595/4	3: TroubleQuickStop	Response for detection that the position is beyond the negative software limit position (C01229/2).
C00595/5	5: Warning	Response if following error limit 1 is exceeded (C01215/1)
C00595/6	5: Warning	Response if following error limit 2 is exceeded (C01215/2)
C00595/7	3: TroubleQuickStop	Response if the maximum travel distance (display in C01213/1) is exceeded
C00595/8	4: WarningLocked	Response to start procedures with reference condition when the reference is not set
C00595/9	4: WarningLocked	Response to a non-supported positioning mode
C00595/10	4: WarningLocked	Response to implausible profile set data
C00595/11	5: Warning	Response to the selection of an invalid operating mode of the MCK
C00595/12	4: WarningLocked	Response to indicating an invalid profile data set
C00595/13	5: Warning	Response to an error of the FB L_MckCtrlInterface_1
C00595/14	4: WarningLocked	Response to a profile start with a target position outside the software limit positions (C01229/1 and C01229/2).
C00595/15	1: Fault	Response to activated connection monitoring in case of PC manual control
C00595/16	6: Information	MCK: direction conflict_Ccw
C00595/17	6: Information	MCK: direction conflict_Cw
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00597

Parameter | Name: **C00597 | Resp. to motor phase failure**		Data type: UNSIGNED_8 Index: 23978_d = $5DAA_h$
Response to motor phase failure • If a phase current does not exceed the threshold set in C00599 for more than one period, the response set here will be triggered.		
Selection list(Lenze setting printed in bold)		
0	**No Reaction**	
1	Fault	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00598

Parameter | Name: **C00598 | Resp. to open circuit AINx**		Data type: UNSIGNED_8 Index: 23977_d = $5DA9_h$
Configuration of monitoring of the analog inputs ▸ Analog terminals		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
5	Warning	
Subcodes	**Lenze setting**	**Info**
C00598/1	3: TroubleQuickStop	Response to open circuit at AIN1 if configured as 4 ... 20 mA current loop
C00598/2	3: TroubleQuickStop	Response to open circuit at AIN2 when being configured as 4 ... 20 mA-current loop
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00599

Parameter | Name: **C00599 | Motor phase failure threshold**			Data type: INTEGER_16 Index: 23976_d = $5DA8_h$
Threshold for motor phase failure monitoring • 100 % ≡ rated inverter current (C00098) • If a phase current does not exceed the threshold set here for more than one period, the response to motor phase failure set in C00597 will be triggered.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**5.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00600

Parameter | Name: **C00600 | Resp. to DC bus voltage**		Data type: UNSIGNED_8 Index: 23975_d = $5DA7_h$
Configuration of monitoring of the motor control (group 3)		
Selection list		
1	Fault	
2	Trouble	
Subcodes	**Lenze setting**	**Info**
C00600/1	2: Trouble	Response to DC bus undervoltage
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00601

Parameter | Name: **C00601 | Del. resp. to fault: DC bus overvoltage**			Data type: UNSIGNED_16 Index: 23974_d = $5DA6_h$
Error response delay times			
Setting range **(min. value | unit | max. value)**			
0.000	s	65.000	
Subcodes	**Lenze setting**		**Info**
C00601/1	2.000 s		Delay time for triggering the "DC-bus overvoltage" error • If a DC-bus overvoltage occurs, an error will not be triggered until the set delay time has elapsed.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1000			

C00602

Parameter | Name: **C00602 | Resp. to earth fault**		Data type: UNSIGNED_8 Index: 23973_d = $5DA5_h$
Response to earth fault in the motor phase(s)		
Selection list **(Lenze setting printed in bold)**		
0	No Reaction	
1	**Fault**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00603

Parameter | Name: **C00603 | Resp. to feedback**		Data type: UNSIGNED_8 Index: 23972_d = $5DA4_h$
Response of different monitoring modes for the encoder/feedback system ▸ Encoder/feedback system		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Information**
C00603/1	1: Fault	Resp. to MultiEncoder open circuit
C00603/2	1: Fault	Resp. to resolver open circuit
C00603/3	0: No Reaction	Up to and including version 02.xx.xx: Resp. to encoder communication error From version 12.00.00: Resp. to encoder angular drift monitoring
C00603/4	1: Fault	Up to and including version 02.xx.xx: Resp. to encoder angular drift monitoring From version 12.00.00: Resp. to encoder communication error
C00603/5	6: Information	Resp. to Hiperface status • Ab Version 14.00.00
C00603/6	6: Information	Resp. to invalid Hiperface position • Ab Version 14.00.00
C00603/7	6: Information	Resp. to SinCos inaccuracy • Ab Version 14.00.00
C00603/8	0: No Reaction	Resp. to Hiperface sin/cos deviation • From version 15.00.00
C00603/9	1: Fault	Resp. to encoder supply
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00604

Parameter | Name: **C00604 | Resp. to device overload (Ixt)**		Data type: UNSIGNED_8 Index: 23971_d = $5DA3_h$
Response if the adjustable device utilisation threshold (C00123) is reached. • The current device utilisation is displayed in C00064.		
Selection list (Lenze setting printed in bold)		
0	No Reaction	
1	Fault	
5	**Warning**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00605

Parameter | Name: C00605 | Resp. to feedback		Data type: UNSIGNED_8 Index: 23970_d = $5DA2_h$
From version 15.00.00		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00605/1	1: Fault	Resp. to open circuit HTL 4-fold
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00606

Parameter | Name: C00606 | Resp. to motor overload (I²xt)	Data type: UNSIGNED_8 Index: 23969_d = $5DA1_h$
Response when the motor load displayed in C00066 reaches the value "100.00 %". ▶ Motor overload monitoring (I2xt)	
Selection list(Lenze setting printed in bold)	
0	No Reaction
1	Fault
5	**Warning**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1	

C00607

Parameter | Name: C00607 | Resp. to max freq. feedb. DIG12/67	Data type: UNSIGNED_8 Index: 23968_d = $5DA0_h$
Response when the maximum input frequency has been reached via the digital inputs.	
Selection list(Lenze setting printed in bold)	
0	No Reaction
1	**Fault**
2	Trouble
3	TroubleQuickStop
4	WarningLocked
5	Warning
6	Information
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1	

C00608

Parameter | Name: **C00608 | Resp. to maximum torque**	Data type: UNSIGNED_8 Index: 23967_d = $5D9F_h$
Response if the maximum torque (C00057) is reached.	
Selection list(Lenze setting printed in bold)	
0	**No Reaction**
1	Fault
5	Warning
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1	

C00609

Parameter | Name: **C00609 | Resp. to maximum current**	Data type: UNSIGNED_8 Index: 23966_d = $5D9E_h$
Response if the maximum current (C00022, C00023) is reached.	
Selection list(Lenze setting printed in bold)	
0	**No Reaction**
1	Fault
5	Warning
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1	

C00610

Parameter | Name: **C00610 | 16-bit connection table**	Data type: UNSIGNED_16 Index: 23965_d = $5D9D_h$
This code is used device-internally and must not be written by the user side!	

C00611

Parameter | Name: **C00611 | Bool connection table**	Data type: UNSIGNED_16 Index: 23964_d = $5D9C_h$
This code is used device-internally and must not be written by the user side!	

C00612

Parameter | Name: **C00612 | 32-bit connection table**	Data type: UNSIGNED_16 Index: 23963_d = $5D9B_h$
This code is used device-internally and must not be written by the user side!	

C00613

Parameter | Name: **C00613 | 16-bit connection table AdditionalFBsHL**	Data type: UNSIGNED_16 Index: 23962_d = $5D9A_h$
This code is used device-internally and must not be written by the user side!	

C00615

Parameter | Name: **C00615 | Bool connection table AdditionalFBsHL**	Data type: UNSIGNED_16 Index: 23960_d = $5D98_h$
This code is used device-internally and must not be written by the user side!	

C00617

Parameter | Name: **C00617 | 32-bit connection table AdditionalFBsHL**	Data type: UNSIGNED_16 Index: 23958_d = $5D96_h$
This code is used device-internally and must not be written by the user side!	

C00620

Parameter | Name: **C00620 | System connection list: 16-bit**		Data type: UNSIGNED_16 Index: 23955_d = $5D93_h$
Connection parameters: 16-bit inputs • Selection of the 16 bit output signals to be connected to the 16 bit input signals • The selection list contains all 16 bit output signals which can be assigned to the 16 bit inputs displayed by the subcodes. • Non-listed subcodes are "reserved".		
Selection list		
See selection list - analog signals		
Subcodes	**Lenze setting**	**Info**
C00620/1	1003: LA_NCtrl: nMotorSpeedAct_a	LS_AnalogOutput: nOut1_a (V) Analog output 1: Voltage
C00620/2	0: Not connected	LP_CanOut1: wState CAN1 output: Status word
C00620/3	0: Not connected	LP_CanOut1: wOut2 CAN1 output: Data word 2
C00620/4	0: Not connected	LP_CanOut1: wOut3 CAN1 output: Data word 3
C00620/5	0: Not connected	LP_CanOut1: wOut4 CAN1 output: Data word 4
C00620/6	0: Not connected	LP_CanOut2: wOut1 CAN2 output: Data word 1
C00620/7	0: Not connected	LP_CanOut2: wOut2 CAN2 output: Data word 2
C00620/8	0: Not connected	LP_CanOut2: wOut3 CAN2 output: Data word 3
C00620/9	0: Not connected	LP_CanOut2: wOut4 CAN2 output: Data word 4
C00620/10	0: Not connected	LP_CanOut3: wOut1 CAN3 output: Data word 1
C00620/11	0: Not connected	LP_CanOut3: wOut2 CAN3 output: Data word 2
C00620/12	0: Not connected	LP_CanOut3: wOut3 CAN3 output: Data word 3
C00620/13	0: Not connected	LP_CanOut3: wOut4 CAN3 output: Data word 4
C00620/14	0: Not connected	LS_DisFree_a: nDis1_a Display of analog signal 1
C00620/15	0: Not connected	LS_DisFree_a: nDis2_a Display of analog signal 2
C00620/16	0: Not connected	LS_DisFree_a: nDis3_a Display of analog signal 3
C00620/17	0: Not connected	LS_DisFree_a: nDis4_a Display of analog signal 4
C00620/18	0: Not connected	LS_DisFree: wDis1 Display of 16-bit signal 1

Parameter | Name: **C00620 | System connection list: 16-bit**		Data type: UNSIGNED_16 Index: 23955_d = $5D93_h$
C00620/19	0: Not connected	LS_DisFree: wDis2 Display of 16-bit signal 2
C00620/20	0: Not connected	LS_DisFree: wDis3 Display of 16-bit signal 3
C00620/21	0: Not connected	LS_DisFree: wDis4 Display of 16-bit signal 4
C00620/22	0: Not connected	LP_MciOut: wState MCI output: Status word
C00620/23	0: Not connected	LP_MciOut: wOut2 MCI output: Data word 2
C00620/24	0: Not connected	LP_MciOut: wOut3 MCI output: Data word 3
C00620/25	0: Not connected	LP_MciOut: wOut4 MCI output: Data word 4
C00620/26	0: Not connected	LP_MciOut: wOut5 MCI output: Data word 5
C00620/27	0: Not connected	LP_MciOut: wOut6 MCI output: Data word 6
C00620/28	0: Not connected	LP_MciOut: wOut7 MCI output: Data word 7
C00620/29	0: Not connected	LP_MciOut: wOut8 MCI output: Data word 8
C00620/30	0: Not connected	LP_MciOut: wOut9 MCI output: Data word 9
C00620/31	0: Not connected	LP_MciOut: wOut10 MCI output: Data word 10
C00620/32	0: Not connected	LP_MciOut: wOut11 MCI output: Data word 11
C00620/33	0: Not connected	LP_MciOut: wOut12 MCI output: Data word 12
C00620/34	0: Not connected	LP_MciOut: wOut13 MCI output: Data word 13
C00620/35	0: Not connected	LP_MciOut: wOut14 MCI output: Data word 14
C00620/36	0: Not connected	LP_MciOut: wOut15 MCI output: Data word 15
C00620/37	0: Not connected	LP_MciOut: wOut16 MCI output: Data word 16
C00620/38	0: Not connected	LS_AnalogOutput: nOut2_a (V) Analog output 2: Voltage
C00620/39	0: Not connected	LS_AnalogOutput: nOut1_a (I) Analog output 1: Current
C00620/40	0: Not connected	LS_AnalogOutput: nOut2_a (I) Analog output 2: Current
C00620/41	0: Not connected	LS_DisFree_a: nDis5_a Display of analog signal 5
C00620/42	0: Not connected	LS_DisFree_a: nDis6_a Display of analog signal 6
C00620/43	0: Not connected	LS_DisFree_a: nDis7_a Display of analog signal 7

Parameter | Name: **C00620 | System connection list: 16-bit**		Data type: UNSIGNED_16 Index: 23955_d = $5D93_h$
C00620/44	0: Not connected	LS_DisFree_a: nDis8_a Display of analog signal 8
C00620/45	0: Not connected	LS_DisFree: wDis5 Display of 16-bit signal 5
C00620/46	0: Not connected	LS_DisFree: wDis6 Display of 16-bit signal 6
C00620/47	0: Not connected	LS_DisFree: wDis7 Display of 16-bit signal 7
C00620/48	0: Not connected	LS_DisFree: wDis8 Display of 16-bit signal 8
C00620/49	0: Not connected	LS_ParReadWrite_1: wParIndex Read/write request 1: Code
C00620/50	0: Not connected	LS_ParReadWrite_1: wParSubindex Read/write request 1: Subcode
C00620/51	0: Not connected	LS_ParReadWrite_1: wInHWord Read/write request 1: Value (high word)
C00620/52	0: Not connected	LS_ParReadWrite_1: wInLWord Read/write request 1: Value (low word)
C00620/53	0: Not connected	LS_ParReadWrite_2: wParIndex Read/write request 2: Code
C00620/54	0: Not connected	LS_ParReadWrite_2: wParSubindex Read/write request 2: Subcode
C00620/55	0: Not connected	LS_ParReadWrite_2: wInHWord Read/write request 2: Value (high word)
C00620/56	0: Not connected	LS_ParReadWrite_2: wInLWord Read/write request 2: Value (low word)
C00620/57	0: Not connected	LS_ParReadWrite_3: wParIndex Read/write request 3: Code
C00620/58	0: Not connected	LS_ParReadWrite_3: wParSubindex Read/write request 3: Subcode
C00620/59	0: Not connected	LS_ParReadWrite_3: wInHWord Read/write request 3: Value (high word)
C00620/60	0: Not connected	LS_ParReadWrite_3: wInLWord Read/write request 3: Value (low word)
C00620/61	0: Not connected	LS_ParReadWrite_4: wParIndex Read/write request 4: Code
C00620/62	0: Not connected	LS_ParReadWrite_4: wParSubindex Read/write request 4: Subcode
C00620/63	0: Not connected	LS_ParReadWrite_4: wInHWord Read/write request 4: Value (high word)
C00620/64	0: Not connected	LS_ParReadWrite_4: wInLWord Read/write request 4: Value (low word)
C00620/65	0: Not connected	LS_ParReadWrite_5: wParIndex Read/write request 5: Code
C00620/66	0: Not connected	LS_ParReadWrite_5: wParSubindex Read/write request 5: Subcode
C00620/67	0: Not connected	LS_ParReadWrite_5: wInHWord Read/write request 5: Value (high word)
C00620/68	0: Not connected	LS_ParReadWrite_5: wInLWord Read/write request 5: Value (low word)

Parameter | Name: **C00620 | System connection list: 16-bit**		Data type: UNSIGNED_16 Index: 23955_d = $5D93_h$
C00620/69	0: Not connected	LS_ParReadWrite_6: wParIndex Read/write request 6: Code
C00620/70	0: Not connected	LS_ParReadWrite_6: wParSubindex Read/write request 6: Subcode
C00620/71	0: Not connected	LS_ParReadWrite_6: wInHWord Read/write request 6: Value (high word)
C00620/72	0: Not connected	LS_ParReadWrite_6: wInLWord Read/write request 6: Value (low word)
C00620/89	0: Not connected	LS_RetainData: wIn1 Retain data: 16-bit input value 1
C00620/90	0: Not connected	LS_RetainData: wIn2 Retain data: 16-bit input value 2
C00620/91	0: Not connected	LS_RetainData: wIn3 Retain data: 16-bit input value 3
C00620/92	0: Not connected	LS_RetainData: wIn4 Retain data: 16-bit input value 4
C00620/93	0: Not connected	LS_AxisBusOut: wLine1 Axis bus output: Line data (word 1)
C00620/94	0: Not connected	LS_AxisBusOut: wLine2 Axis bus output: Line data (word 2)
C00620/95	0: Not connected	LS_AxisBusOut: wLine3 Axis bus output: Line data (word 3)
C00620/96	0: Not connected	LS_AxisBusOut: wCas1 Axis bus output: Cascaded data (word 1)
C00620/97	0: Not connected	LS_AxisBusOut: wCas2 Axis bus output: Cascaded data (word 2)
C00620/98	0: Not connected	LS_AxisBusOut: wCas3 Axis bus output: Cascaded data (word 3)
C00620/99	0: Not connected	LS_AxisBusOut: wCas4 Axis bus output: Cascaded data (word 4)
C00620/100	0: Not connected	LS_AxisBusAux: wAuxOut1 Axis bus output: @sel data (word 1)
C00620/101	0: Not connected	LS_AxisBusAux: wAuxOut2 Axis bus output: @sel data (word 2)
C00620/102	0: Not connected	LS_AxisBusAux: wAuxOut3 Axis bus output: @sel data (word 3)
C00620/103	0: Not connected	LS_AxisBusAux: wAuxOut4 Axis bus output: @sel data (word 4)
C00620/104	0: Not connected	LS_AxisBusAux: wSelectSlave Axis bus address of slave
C00620/105	0: Not connected	LS_DFOut: nOut_v Digital frequency output X8: Speed in [increments/ms]
C00620/106	0: Not connected	LS_MultiEncoder: wActPosExternalHW
C00620/107	0: Not connected	LS_MultiEncoder: wActPosExternalLW
C00620/108	0: Not connected	LP_CanOut4: wOut1
C00620/109	0: Not connected	LP_CanOut4: wOut2
C00620/110	0: Not connected	LP_CanOut4: wOut3
C00620/111	0: Not connected	LP_CanOut4: wOut4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00621

Parameter | Name: **C00621 | System connection list: Bool**		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
Connection parameters: Binary inputs • Selection of the binary output signals to be connected to the binary input signals • The selection list contains all binary output signals which can be assigned to the binary inputs mapped by the subcodes. • Non-listed subcodes are "reserved".		
Selection list		
See selection list - digital signals		
Subcodes	**Lenze setting**	**Info**
C00621/1	1001: LA_NCtrl: bDriveFail	LS_DigitalOutput: bRelay Digital relay output: Input signal
C00621/2	1000: LA_NCtrl: bDriveReady	LS_DigitalOutput: bOut1 Digital output 1: Input signal
C00621/3	0: Not connected	LS_DigitalInput: bCountIn1_Reset Digital input 1: Reset counter
C00621/4	0: Not connected	LS_DigitalInput: bCountIn1_LoadStartValue Digital input 1: Load counter starting value
C00621/5	0: Not connected	LP_CanOut1: bState_B0 CAN1 output: Status word bit 0
C00621/6	0: Not connected	LP_CanOut1: bState_B1 CAN1 output: Status word bit 1
C00621/7	0: Not connected	LP_CanOut1: bState_B2 CAN1 output: Status word bit 2
C00621/8	0: Not connected	LP_CanOut1: bState_B3 CAN1 output: Status word bit 3
C00621/9	0: Not connected	LP_CanOut1: bState_B4 CAN1 output: Status word bit 4
C00621/10	0: Not connected	LP_CanOut1: bState_B5 CAN1 output: Status word bit 5
C00621/11	0: Not connected	LP_CanOut1: bState_B6 CAN1 output: Status word bit 6
C00621/12	0: Not connected	LP_CanOut1: bState_B7 CAN1 output: Status word bit 7
C00621/13	0: Not connected	LP_CanOut1: bState_B8 CAN1 output: Status word bit 8
C00621/14	0: Not connected	LP_CanOut1: bState_B9 CAN1 output: Status word bit 9
C00621/15	0: Not connected	LP_CanOut1: bState_B10 CAN1 output: Status word bit 10
C00621/16	0: Not connected	LP_CanOut1: bState_B11 CAN1 output: Status word bit 11
C00621/17	0: Not connected	LP_CanOut1: bState_B12 CAN1 output: Status word bit 12
C00621/18	0: Not connected	LP_CanOut1: bState_B13 CAN1 output: Status word bit 13
C00621/19	0: Not connected	LP_CanOut1: bState_B14 CAN1 output: Status word bit 14
C00621/20	0: Not connected	LP_CanOut1: bState_B15 CAN1 output: Status word bit 15

Parameter | Name: **C00621 | System connection list: Bool**		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
C00621/21	0: Not connected	LS_DisFree_b: bDis1 Display of digital signal 1
C00621/22	0: Not connected	LS_DisFree_b: bDis2 Display of digital signal 2
C00621/23	0: Not connected	LS_DisFree_b: bDis3 Display of digital signal 3
C00621/24	0: Not connected	LS_DisFree_b: bDis4 Display of digital signal 4
C00621/25	0: Not connected	LS_DisFree_b: bDis5 Display of digital signal 5
C00621/26	0: Not connected	LS_DisFree_b: bDis6 Display of digital signal 6
C00621/27	0: Not connected	LS_DisFree_b: bDis7 Display of digital signal 7
C00621/28	0: Not connected	LS_DisFree_b: bDis8 Display of digital signal 8
C00621/29	0: Not connected	LP_CanOut2: bOut1_B0 CAN2 output: Data word 1- bit 0
C00621/30	0: Not connected	LP_CanOut2: bOut1_B1 CAN2 output: Data word 1- bit 1
C00621/31	0: Not connected	LP_CanOut2: bOut1_B2 CAN2 output: Data word 1- bit 2
C00621/32	0: Not connected	LP_CanOut2: bOut1_B3 CAN2 output: Data word 1- bit 3
C00621/33	0: Not connected	LP_CanOut2: bOut1_B4 CAN2 output: Data word 1- bit 4
C00621/34	0: Not connected	LP_CanOut2: bOut1_B5 CAN2 output: Data word 1- bit 5
C00621/35	0: Not connected	LP_CanOut2: bOut1_B6 CAN2 output: Data word 1- bit 6
C00621/36	0: Not connected	LP_CanOut2: bOut1_B7 CAN2 output: Data word 1- bit 7
C00621/37	0: Not connected	LP_CanOut2: bOut1_B8 CAN2 output: Data word 1- bit 8
C00621/38	0: Not connected	LP_CanOut2: bOut1_B9 CAN2 output: Data word 1- bit 9
C00621/39	0: Not connected	LP_CanOut2: bOut1_B10 CAN2 output: Data word 1- bit 10
C00621/40	0: Not connected	LP_CanOut2: bOut1_B11 CAN2 output: Data word 1- bit 11
C00621/41	0: Not connected	LP_CanOut2: bOut1_B12 CAN2 output: Data word 1- bit 12
C00621/42	0: Not connected	LP_CanOut2: bOut1_B13 CAN2 output: Data word 1- bit 13
C00621/43	0: Not connected	LP_CanOut2: bOut1_B14 CAN2 output: Data word 1- bit 14
C00621/44	0: Not connected	LP_CanOut2: bOut1_B15 CAN2 output: Data word 1- bit 15
C00621/45	0: Not connected	LP_CanOut3: bOut1_B0 CAN3 output: Data word 1- bit 0

Parameter | Name: **C00621 | System connection list: Bool**		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
C00621/46	0: Not connected	LP_CanOut3: bOut1_B1 CAN3 output: Data word 1- bit 1
C00621/47	0: Not connected	LP_CanOut3: bOut1_B2 CAN3 output: Data word 1- bit 2
C00621/48	0: Not connected	LP_CanOut3: bOut1_B3 CAN3 output: Data word 1- bit 3
C00621/49	0: Not connected	LP_CanOut3: bOut1_B4 CAN3 output: Data word 1- bit 4
C00621/50	0: Not connected	LP_CanOut3: bOut1_B5 CAN3 output: Data word 1- bit 5
C00621/51	0: Not connected	LP_CanOut3: bOut1_B6 CAN3 output: Data word 1- bit 6
C00621/52	0: Not connected	LP_CanOut3: bOut1_B7 CAN3 output: Data word 1- bit 7
C00621/53	0: Not connected	LP_CanOut3: bOut1_B8 CAN3 output: Data word 1- bit 8
C00621/54	0: Not connected	LP_CanOut3: bOut1_B9 CAN3 output: Data word 1- bit 9
C00621/55	0: Not connected	LP_CanOut3: bOut1_B10 CAN3 output: Data word 1- bit 10
C00621/56	0: Not connected	LP_CanOut3: bOut1_B11 CAN3 output: Data word 1- bit 11
C00621/57	0: Not connected	LP_CanOut3: bOut1_B12 CAN3 output: Data word 1- bit 12
C00621/58	0: Not connected	LP_CanOut3: bOut1_B13 CAN3 output: Data word 1- bit 13
C00621/59	0: Not connected	LP_CanOut3: bOut1_B14 CAN3 output: Data word 1- bit 14
C00621/60	0: Not connected	LP_CanOut3: bOut1_B15 CAN3 output: Data word 1- bit 15
C00621/61	0: Not connected	LP_MciOut: bState_B0 MCI output: Status word - bit 0
C00621/62	0: Not connected	LP_MciOut: bState_B1 MCI output: Status word - bit 1
C00621/63	0: Not connected	LP_MciOut: bState_B2 MCI output: Status word - bit 2
C00621/64	0: Not connected	LP_MciOut: bState_B3 MCI output: Status word - bit 3
C00621/65	0: Not connected	LP_MciOut: bState_B4 MCI output: Status word - bit 4
C00621/66	0: Not connected	LP_MciOut: bState_B5 MCI output: Status word - bit 5
C00621/67	0: Not connected	LP_MciOut: bState_B6 MCI output: Status word - bit 6
C00621/68	0: Not connected	LP_MciOut: bState_B7 MCI output: Status word - bit 7
C00621/69	0: Not connected	LP_MciOut: bState_B8 MCI output: Status word - bit 8
C00621/70	0: Not connected	LP_MciOut: bState_B9 MCI output: Status word - bit 9

Parameter | Name: C00621 | System connection list: Bool		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
C00621/71	0: Not connected	LP_MciOut: bState_B10 MCI output: Status word - bit 10
C00621/72	0: Not connected	LP_MciOut: bState_B11 MCI output: Status word - bit 11
C00621/73	0: Not connected	LP_MciOut: bState_B12 MCI output: Status word - bit 12
C00621/74	0: Not connected	LP_MciOut: bState_B13 MCI output: Status word - bit 13
C00621/75	0: Not connected	LP_MciOut: bState_B14 MCI output: Status word - bit 14
C00621/76	0: Not connected	LP_MciOut: bState_B15 MCI output: Status word - bit 15
C00621/77	0: Not connected	LP_MciOut: bOut2_B0 MCI output: Data word 2 - bit 0
C00621/78	0: Not connected	LP_MciOut: bOut2_B1 MCI output: Data word 2 - bit 1
C00621/79	0: Not connected	LP_MciOut: bOut2_B2 MCI output: Data word 2 - bit 2
C00621/80	0: Not connected	LP_MciOut: bOut2_B3 MCI output: Data word 2 - bit 3
C00621/81	0: Not connected	LP_MciOut: bOut2_B4 MCI output: Data word 2 - bit 4
C00621/82	0: Not connected	LP_MciOut: bOut2_B5 MCI output: Data word 2 - bit 5
C00621/83	0: Not connected	LP_MciOut: bOut2_B6 MCI output: Data word 2 - bit 6
C00621/84	0: Not connected	LP_MciOut: bOut2_B7 MCI output: Data word 2 - bit 7
C00621/85	0: Not connected	LP_MciOut: bOut2_B8 MCI output: Data word 2 - bit 8
C00621/86	0: Not connected	LP_MciOut: bOut2_B9 MCI output: Data word 2 - bit 9
C00621/87	0: Not connected	LP_MciOut: bOut2_B10 MCI output: Data word 2 - bit 10
C00621/88	0: Not connected	LP_MciOut: bOut2_B11 MCI output: Data word 2 - bit 11
C00621/89	0: Not connected	LP_MciOut: bOut2_B12 MCI output: Data word 2 - bit 12
C00621/90	0: Not connected	LP_MciOut: bOut2_B13 MCI output: Data word 2 - bit 13
C00621/91	0: Not connected	LP_MciOut: bOut2_B14 MCI output: Data word 2 - bit 14
C00621/92	0: Not connected	LP_MciOut: bOut2_B15 MCI output: Data word 2 - bit 15
C00621/93	0: Not connected	LS_SetError_1: bSetError1 Input for tripping "US01: User error 1"
C00621/94	0: Not connected	LS_SetError_1: bSetError2 Input for tripping "US02: User error 2"
C00621/95	0: Not connected	LS_SetError_1: bSetError3 Input for tripping "US03: User error 3"

Parameter | Name: **C00621 | System connection list: Bool**		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
C00621/96	0: Not connected	LS_SetError_1: bSetError4 Input for tripping "US04: User error 4"
C00621/97	0: Not connected	LS_DigitalInput: bCountIn6_Reset Digital input 6: Reset counter
C00621/98	0: Not connected	LS_DigitalInput: bCountIn6_LoadStartValue Digital input 6: Load counter starting value
C00621/99	0: Not connected	LS_DigitalOutput: bOut2 Digital output 2: Input signal
C00621/100	0: Not connected	LS_DigitalOutput: bOut3 Digital output 3: Input signal
C00621/101	0: Not connected	LS_DigitalOutput: bOut HighCurrent Digital output (HC) for brake control: Input signal
C00621/102	0: Not connected	LS_DisFree_b: bDis9 Display of digital signal 9
C00621/103	0: Not connected	LS_DisFree_b: bDis10 Display of digital signal 10
C00621/104	0: Not connected	LS_DisFree_b: bDis11 Display of digital signal 11
C00621/105	0: Not connected	LS_DisFree_b: bDis12 Display of digital signal 12
C00621/106	0: Not connected	LS_DisFree_b: bDis13 Display of digital signal 13
C00621/107	0: Not connected	LS_DisFree_b: bDis14 Display of digital signal 14
C00621/108	0: Not connected	LS_DisFree_b: bDis15 Display of digital signal 15
C00621/109	0: Not connected	LS_DisFree_b: bDis16 Display of digital signal 16
C00621/111	0: Not connected	LS_ParReadWrite_1: bExecute Read/write request 1: Trigger
C00621/112	0: Not connected	LS_ParReadWrite_1: bReadWrite Read/write request 1: Selection of reading/writing
C00621/113	0: Not connected	LS_ParReadWrite_2: bExecute Read/write request 2: Trigger
C00621/114	0: Not connected	LS_ParReadWrite_2: bReadWrite Read/write request 2: Selection of reading/writing
C00621/115	0: Not connected	LS_ParReadWrite_3: bExecute Read/write request 3: Trigger
C00621/116	0: Not connected	LS_ParReadWrite_3: bReadWrite Read/write request 3: Selection of reading/writing
C00621/117	0: Not connected	LS_ParReadWrite_4: bExecute Read/write request 4: Trigger
C00621/118	0: Not connected	LS_ParReadWrite_4: bReadWrite Read/write request 4: Selection of reading/writing
C00621/119	0: Not connected	LS_ParReadWrite_5: bExecute Read/write request 5: Trigger
C00621/120	0: Not connected	LS_ParReadWrite_5: bReadWrite Read/write request 5: Selection of reading/writing
C00621/121	0: Not connected	LS_ParReadWrite_6: bExecute Read/write request 6: Initiate

Parameter | Name: C00621 | System connection list: Bool		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
C00621/122	0: Not connected	LS_ParReadWrite_6: bReadWrite Read/write request 6: Select reading/writing
C00621/123	0: Not connected	LS_WriteParamList: bExecute Writing the parameter list: Activate
C00621/124	0: Not connected	LS_WriteParamList: bSelectWriteValue_1 Writing to parameter list: Selection of value set - 1
C00621/125	0: Not connected	LS_WriteParamList: bSelectWriteValue_2 Writing to parameter list: Select value set - 2
C00621/126	0: Not connected	LS_CANManagement: bResetNode Reset CAN node
C00621/127	0: Not connected	LS_CANManagement: bReInitCAN Reinitialise CAN interface
C00621/128	0: Not connected	LS_DigitalInput: bPosIn12_Load Frequency input DI1/DI2: Load angle integrator with starting value and reset status signal
C00621/147	0: Not connected	LS_TouchProbe: bDisableTPDigIn3 TP input DI3: Disable TP function dynamically
C00621/148	0: Not connected	LS_TouchProbe: bDisableTPDigIn4 TP input DI4: Disable TP function dynamically
C00621/149	0: Not connected	LS_TouchProbe: bDisableTPDigIn5 TP input DI5: Disable TP function dynamically
C00621/150	0: Not connected	LS_TouchProbe: bDisableTPDigIn6 TP input DI6: Disable TP function dynamically
C00621/151	0: Not connected	LS_TouchProbe: bDisableTPDigIn7 TP input DI7: Disable TP function dynamically
C00621/152	0: Not connected	LS_TouchProbe: bDisableTPDigIn3_Rising TP input DI3: Disable recognition of rising edges dynamically
C00621/153	0: Not connected	LS_TouchProbe: bDisableTPDigIn3_Falling TP input DI3: Disable recognition of falling edges dynamically
C00621/154	0: Not connected	LS_TouchProbe: bDisableTPDigIn4_Rising TP input DI4: Disable recognition of rising edges dynamically
C00621/155	0: Not connected	LS_TouchProbe: bDisableTPDigIn4_Falling TP input DI4: Disable recognition of falling edges dynamically
C00621/156	0: Not connected	LS_TouchProbe: bDisableTPDigIn5_Rising TP input DI5: Disable recognition of rising edges dynamically
C00621/157	0: Not connected	LS_TouchProbe: bDisableTPDigIn5_Falling TP input DI5: Disable recognition of falling edges dynamically
C00621/158	0: Not connected	LS_TouchProbe: bDisableTPDigIn3Window TP input DI3: Enable acceptance window
C00621/159	0: Not connected	LS_TouchProbe: bDisableTPDigIn4Window TP input DI4: Enable acceptance window
C00621/160	0: Not connected	LS_TouchProbe: bDisableTPDigIn5Window TP input DI5: Enable acceptance window
C00621/161	0: Not connected	LS_AxisBusIO: bSetFail Set IO axis bus into the "error" status / IO data exchange
C00621/162	0: Not connected	LS_AxisBusIO: bResetFail IO axis bus: Reset "error" status

Parameter | Name: **C00621 | System connection list: Bool**		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
C00621/164	0: Not connected	LS_MultiEncoder: bSetRef
C00621/165	0: Not connected	LS_RetainData: bSetRetain_1 Retain data: Save input values (selection 1) in retain memory
C00621/166	0: Not connected	LS_RetainData: bSetRetain_2 Retain data: Save input values (selection 2) in retain memory
C00621/167	0: Not connected	LS_RetainData: bSetRetain_3 Retain data: Save input values (selection 3) in retain memory
C00621/168	0: Not connected	LS_RetainData: bLoadParams Retain data: Set selected outputs & retain values to parameter values
C00621/169	0: Not connected	LS_RetainData: bIn1 Retain data: Binary input value 1
C00621/170	0: Not connected	LS_RetainData: bIn2 Retain data: Binary input value 2
C00621/171	0: Not connected	LS_RetainData: bIn3 Retain data: Binary input value 3
C00621/172	0: Not connected	LS_RetainData: bIn4 Retain data: Binary input value 4
C00621/173	0: Not connected	LS_AxisBusAux: bReadWrite Axis bus: Selection of reading/writing
C00621/174	0: Not connected	LS_AxisBusAux: bExecute Axis bus: Data acceptance
C00621/175	0: Not connected	LS_AxisBusAux: bStop Axis bus: Stop data transfer
C00621/176	0: Not connected	LS_DigitalOutput: bUserLED Switch on USER-LED on the inverter front panel
C00621/177	0: Not connected	LS_TouchProbe: bDisableTPEncoderWindow
C00621/178	0: Not connected	LS_TouchProbe: bDisableTPResolverWindow
C00621/179	0: Not connected	LS_TouchProbe: bDisableTPEncoder
C00621/180	0: Not connected	LS_TouchProbe: bDisableTPResolver
C00621/181	0: Not connected	LF_DFOut: bSynRdy
C00621/184	0: Not connected	LP_CanOut4: bOut1_B0
C00621/185	0: Not connected	LP_CanOut4: bOut1_B1
C00621/186	0: Not connected	LP_CanOut4: bOut1_B2
C00621/187	0: Not connected	LP_CanOut4: bOut1_B3
C00621/188	0: Not connected	LP_CanOut4: bOut1_B4
C00621/189	0: Not connected	LP_CanOut4: bOut1_B5
C00621/190	0: Not connected	LP_CanOut4: bOut1_B6
C00621/191	0: Not connected	LP_CanOut4: bOut1_B7
C00621/192	0: Not connected	LP_CanOut4: bOut1_B8
C00621/193	0: Not connected	LP_CanOut4: bOut1_B9
C00621/194	0: Not connected	LP_CanOut4: bOut1_B10
C00621/195	0: Not connected	LP_CanOut4: bOut1_B11

Parameter | Name: **C00621 | System connection list: Bool**		Data type: UNSIGNED_16 Index: 23954_d = $5D92_h$
C00621/196	0: Not connected	LP_CanOut4: bOut1_B12
C00621/197	0: Not connected	LP_CanOut4: bOut1_B13
C00621/198	0: Not connected	LP_CanOut4: bOut1_B14
C00621/199	0: Not connected	LP_CanOut4: bOut1_B15
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00622

Parameter | Name: **C00622 | System connection list: Angle**		Data type: UNSIGNED_16 Index: 23953_d = $5D91_h$
Connection parameters: 32-bit inputs • Selection of the 32-bit output signals for connection with the 32-bit input signals. • The selection list contains all 32-bit output signals which can be assigned to the 32-bit inputs mapped by the subcodes. • Non-listed subcodes are "reserved".		
Selection list		
See selection list - angle signals		
Subcodes	**Lenze setting**	**Info**
C00622/1	0: Not connected	LS_DisFree_p: dnDis1_p Display of 32-bit signal 1
C00622/2	0: Not connected	LS_DisFree_p: dnDis2_p Display of 32-bit signal 2
C00622/3	0: Not connected	LS_DisFree_p: dnDis3_p Display of 32-bit signal 3
C00622/4	0: Not connected	LS_DisFree_p: dnDis4_p Display of 32-bit signal 4
C00622/5	0: Not connected	LS_DisFree_p: dnDis5_p Display of 32-bit signal 5
C00622/6	0: Not connected	LS_DisFree_p: dnDis6_p Display of 32-bit signal 6
C00622/7	0: Not connected	LS_DisFree_p: dnDis7_p Display of 32-bit signal 7
C00622/8	0: Not connected	LS_DisFree_p: dnDis8_p Display of 32-bit signal 8
C00622/9	0: Not connected	LP_CanOut1: dnOut34_p CAN1 output: Data words 3 + 4
C00622/10	0: Not connected	LP_CanOut2: dnOut34_p CAN2 output: Data words 3 + 4
C00622/11	0: Not connected	LP_CanOut3: dnOut34_p CAN3 output: Data words 3 + 4
C00622/12	0: Not connected	LP_MciOut: dnOut34_p MCI output: Data words 3 + 4
C00622/13	0: Not connected	LS_DigitalInput: dnPosIn12_Set_p Frequency input DI1/DI2: Starting value for angle integrator
C00622/16	0: Not connected	LS_RetainData: dnIn1 Retain data: 32-bit input value 1
C00622/17	0: Not connected	LS_RetainData: dnIn2 Retain data: 32-bit input value 2
C00622/18	0: Not connected	LS_RetainData: dnIn3 Retain data: 32-bit input value 3

Parameter | Name: C00622 | System connection list: Angle		Data type: UNSIGNED_16 Index: 23953_d = $5D91_h$
C00622/19	0: Not connected	LS_RetainData: dnIn4 Retain data: 32-bit input value 4
C00622/20	0: Not connected	LS_AxisBusOut: dnLine12 Axis bus output: Line data (word 1 + 2)
C00622/21	0: Not connected	LS_AxisBusOut: dnCas12 Axis bus output: Cascaded data (word 1 + 2)
C00622/22	0: Not connected	LS_AxisBusOut: dnCas34 Axis bus output: Cascaded data (word 3 + 4)
C00622/23	0: Not connected	LS_AxisBusAux: dnAuxOut12 Axis bus output: @sel data (word 1 + 2)
C00622/24	0: Not connected	LS_AxisBusAux: dnAuxOut34 Axis bus output: @sel data (word 3 + 4)
C00622/25	0: Not connected	LP_CanOut4: dnOut34_p CAN4 output: Data words 3 + 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00630

Parameter | Name: C00630 | L_Limit 1-2: Min/Max			Data type: INTEGER_16 Index: 23945_d = $5D89_h$
Setting the limits			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00630/1	-199.99 %		L_Limit_1: Min.Limit
C00630/2	199.99 %		L_Limit_1: Max.Limit
C00630/3	-199.99 %		L_Limit_2: Min.Limit
C00630/4	199.99 %		L_Limit_2: Max.Limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00631

Parameter | Name: C00631 | L_LimitPhi 1-3: Min/Max			Data type: INTEGER_32 Index: 23944_d = $5D88_h$
Setting the limits			
Setting range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes	**Lenze setting**		**Info**
C00631/1	-2147483647 incr.		L_LimitPhi_1: Min.Limit
C00631/2	2147483647 incr.		L_LimitPhi_1: Max.Limit
C00631/3	-2147483647 incr.		L_LimitPhi_2: Min.Limit
C00631/4	2147483647 incr.		L_LimitPhi_2: Max.Limit
C00631/5	-2147483647 incr.		L_LimitPhi_3: Min.Limit
C00631/6	2147483647 incr.		L_LimitPhi_3: Max.Limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00632

Parameter | Name: **C00632 | L_NSet_1: Max.SkipFrq.**			Data type: INTEGER_16 Index: 23943_d = $5D87_h$
Maximum limit values for the speed blocking zones • Selection of the maximum limit values for the blocking zones in which the speed must not be constant.			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00632/1	0.00 %		L_NSet_1: Blocking speed1 max
C00632/2	0.00 %		L_NSet_1: Blocking speed2 max
C00632/3	0.00 %		L_NSet_1: Blocking speed3 max
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00633

Parameter | Name: **C00633 | L_NSet_1: Min.SkipFrq.**			Data type: INTEGER_16 Index: 23942_d = $5D86_h$
Minimum limit values for the speed blocking zones • Selection of the minimum limit values for the blocking zones in which the speed must not be constant.			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00633/1	0.00 %		L_NSet_1: Blocking speed1 min
C00633/2	0.00 %		L_NSet_1: Blocking speed2 min
C00633/3	0.00 %		L_NSet_1: Blocking speed3 min
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00634

Parameter | Name: **C00634 | L_NSet_1: wState**			Data type: UNSIGNED_16 Index: 23941_d = $5D85_h$
The L_NSet_1 FB: Bit coded status display			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	No blocking zone active		1 ≡ No blocking zone set for constant speeds
Bit 1	Blocking zone 1 active		1 ≡ Suppression of constant speed characteristics within the limits of blocking zone 1
Bit 2	Blocking zone 2 active		1 ≡ Suppression of constant speed characteristics within the limits of blocking zone 2
Bit 3	Blocking zone 3 active		1 ≡ Suppression of constant speed characteristics within the limits of blocking zone 3
Bit 4	Jog in blocking zone		1 ≡ A ramp is used to keep the speed setpoint within a speed blocking zone
Bit 5	MaxLimit active		1 ≡ Speed setpoint is at the maximum speed limit
Bit 6	MinLimit active		1 ≡ Speed setpoint is at the minimum speed limit
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00635

Parameter | Name: **C00635 | L_NSet_1: nMaxLimit**			Data type: INTEGER_16 Index: 23940_d = $5D84_h$
The L_NSet_1 FB: Maximum speed setpoint for speed setpoint limitation			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**199.99 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00636

Parameter | Name: **C00636 | L_NSet_1: nMinLimit**			Data type: INTEGER_16 Index: 23939_d = $5D83_h$
The L_NSet_1 FB: Minimum speed setpoint for speed setpoint limitation			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**-199.99 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00637

Parameter | Name: **C00637 | L_NSet_1: Output blocking zones**			Data type: INTEGER_16 Index: 23938$_d$ = 5D82$_h$
The L_NSet_1 FB: Speed setpoint is displayed after being processed by blocking zone function			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00638

Parameter | Name: **C00638 | L_NSet_1: Output ramp rounding**			Data type: INTEGER_16 Index: 23937$_d$ = 5D81$_h$
The L_NSet_1 FB: Speed setpoint is displayed after being processed by PT1 filter function			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00639

Parameter | Name: **C00639 | L_NSet_1: Output additional value**			Data type: INTEGER_16 Index: 23936$_d$ = 5D80$_h$
The L_NSet_1 FB: Additional speed setpoint is displayed after being processed by the ramp generator			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00640

Parameter | Name: **C00640 | L_NSet_1: nNOut_a**			Data type: INTEGER_16 Index: 23935$_d$ = 5D7F$_h$
The L_NSet_1 FB: Display of the generated main speed setpoint at the output *nNOut_a*			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00643

Parameter | Name: **C00643 | Resp. to PLI monitoring**		Data type: UNSIGNED_8 Index: 23932$_d$ = 5D7C$_h$
		▸ Pole position identification
Selection list		
0	No Reaction	
1	Fault	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C00643/1	1: Fault	Response to PLI monitoring
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00644

Parameter | Name: C00644 | PLI traversing direction		Data type: UNSIGNED_8 Index: 23931_d = $5D7B_h$
		▸ Pole position identification
Selection list		
0	right rotating field	
1	left rotating field	
Subcodes	**Lenze setting**	**Information**
C00644/1	0: right rotating field	PLI 360° traversing direction
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00645

Parameter | Name: C00645 | PLI max. permissible deflection			Data type: INTEGER_16 Index: 23930_d = $5D7A_h$
			▸ Pole position identification
Setting range (min. value | unit | max. value)			
-6.0	°	90.0	
Subcodes	**Lenze setting**		**Information**
C00645/1	0.0 °		Reserved
C00645/2	0.0 °		PLI 360° max. error tolerance
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00646

Parameter | Name: C00646 | PLI current amplitude			Data type: UNSIGNED_16 Index: 23929_d = $5D79_h$
			▸ Pole position identification
Setting range (min. value | unit | max. value)			
1	%	1000	
Subcodes	**Lenze setting**		**Information**
C00646/1	1 %		Reserved
C00646/2	100 %		PLI 360° current amplitude
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00647

Parameter | Name: C00647 | PLI ramp time			Data type: UNSIGNED_16 Index: 23928_d = $5D78_h$
			▸ Pole position identification
Setting range (min. value | unit | max. value)			
5		1000	
Subcodes	**Lenze setting**		**Information**
C00647/1	5		Reserved
C00647/2	100		PLI 360° ramp time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00650

Parameter | Name: **C00650 | L_Arithmetik 3-5: Function**		Data type: UNSIGNED_8 Index: 23925_d = $5D75_h$
Selection of the internal arithmetics		
Selection list		
0	nOut_a = nIn1_a	
1	nOut_a = nIn1_a + nIn2_a	
2	nOut_a = nIn1_a - nIn2_a	
3	nOut_a = (nIn1_a * nIn2_a) / 100%	
4	nOut_a = (nIn1_a * 1%) / |nIn2_a|	
5	nOut_a = (nIn1_a * 100%) / (100% - nIn2_a)	
21	nOut_a = nIn1_a + nIn2_a w/o limit	
22	nOut_a = nIn1_a - nIn2_a w/o limit	
Subcodes	**Lenze setting**	**Info**
C00650/1	0: nOut_a = nIn1_a	L_Arithmetik_3: Function
C00650/2	0: nOut_a = nIn1_a	L_Arithmetik_4: Function
C00650/3	0: nOut_a = nIn1_a	L_Arithmetik_5: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00653

Parameter | Name: **C00653 | Sensibility - Setpoint feedforward control**			Data type: UNSIGNED_8 Index: 23922_d = $5D72_h$
From version 12.00.00 Selection of the sensitivity of the differential setpoint feedforward control • Depending on the selection, the number of indicated higher-order bits is evaluated. **Note:** The most significant bit determines the sign of the value, the remaining bits determine the numerical value.			
Selection list		**Info**	
0	Inactive		
1	15 bits	Bit 0 ... bit 14 are evaluated	
2	14 Bit	Bit 0 ... bit 13 are evaluated	
3	13 bits	Bit 0 ... bit 12 are evaluated	
4	12 bits	Bit 0 ... bit 11 are evaluated	
5	11 Bit	Bit 0 ... bit 10 are evaluated	
6	10 Bit	Bit 0 ... bit 9 are evaluated	
7	9 Bit	Bit 0 ... bit 8 are evaluated	
Subcodes	**Lenze setting**	**Info**	
C00653/1	0: Inactive	Sensibility - Setpoint feedforward control	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00654

Parameter | Name: **C00654 | Source of diff. setpoint feedforward control**			Data type: UNSIGNED_8 Index: 23921_d = $5D71_h$
From version 12.00.00 Selection of the process signal for the setpoint feedforward control			
Selection list		**Info**	
0	nSpeedSetValue_a		
1	nSpeedSetValueInertia_a	The new process signal *nSpeedSetValueInertia_a* at SB LS_MotorInterface can be used to define any input value (e.g. position or process controller setpoint) for torque feedforward control.	
2	nSpeedSetValue_a exact		
3	nSpeedSetValue_a V14.0		
Subcodes	**Lenze setting**	**Info**	
C00654/1	0: nSpeedSetValue_a	Source of diff. setpoint feedforward control	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00658

Parameter | Name: **C00658 | PRBS amplitude**	Data type: INTEGER_16 Index: 23917_d = $5D6D_h$
This code is used device-internally and must not be written by the user side!	

C00659

Parameter | Name: **C00659 | PRBS duration**	Data type: UNSIGNED_16 Index: 23916_d = $5D6C_h$
This code is used device-internally and must not be written by the user side!	

C00660

Parameter | Name: **C00660 | L_FixSet_a_1: Analog values**			Data type: INTEGER_16 Index: 23915_d = $5D6B_h$
FB L_FixSet_a_1: Setting of the fixed values			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00660/1	0.00 %		Fixed value 0 ... 15
C00660/...			
C00660/16			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00661

Parameter | Name: **C00661 | L_FixSet_w_1: Fixed values**			Data type: UNSIGNED_16 Index: 23914_d = $5D6A_h$
FB L_FixSet_w_1: Setting of the fixed values			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C00661/1	0		Fixed value 0 ... 15
C00661/...			
C00661/16			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00662

Parameter | Name: **C00662 | L_FixSet_w_2: Fixed values**			Data type: UNSIGNED_16 Index: 23913_d = $5D69_h$
FB L_FixSet_w_2: Setting of the fixed values			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C00662/1	0		Fixed value 0 ... 15
C00662/...			
C00662/16			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00670

Parameter | Name: **C00670 | L_OffsetGainP_1: Gain**			Data type: INTEGER_32 Index: 23905_d = $5D61_h$
The L_OffsetGainP_1 FB: Gain as multiplier of the input signal + offset			
Setting range (min. value | unit | max. value)			**Lenze setting**
-100.0000		100.0000	**1.0000**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C00671

Parameter | Name: **C00671 | L_OffsetGainP_2: Gain**			Data type: INTEGER_32 Index: 23904_d = $5D60_h$
The L_OffsetGainP_2 FB: Gain as multiplier of the input signal + offset			
Setting range (min. value | unit | max. value)			**Lenze setting**
-100.0000		100.0000	**1.0000**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C00672

Parameter | Name: C00672 | L_OffsetGainP_3: Gain			Data type: INTEGER_32 Index: 23903_d = $5D5F_h$
The L_OffsetGainP_3 FB: Gain as multiplier of the input signal + offset			
Setting range (min. value | unit | max. value)			**Lenze setting**
-100.0000		100.0000	**1.0000**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C00673

Parameter | Name: C00673 | L_OffsetGainPhiP 1-2: Offset			Data type: INTEGER_32 Index: 23902_d = $5D5E_h$
Angular offset (is added to the angular input signal)			
Setting range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes	**Lenze setting**		**Info**
C00673/1	0 incr.		L_OffsetGainPhiP_1: Offset
C00673/2	0 incr.		L_OffsetGainPhiP_2: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00674

Parameter | Name: C00674 | L_OffsetGainPhiP 1-2: Gain			Data type: INTEGER_32 Index: 23901_d = $5D5D_h$
Angular gain as multiplier of the input signal + angular offset			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C00674/1	65536		L_OffsetGainPhiP_1: Gain
C00674/2	65536		L_OffsetGainPhiP_2: Gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00677

Parameter | Name: C00677 | L_GainOffsetP 1-3: Parameter			Data type: INTEGER_16 Index: 23898_d = $5D5A_h$
Gain and offset			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00677/1	100.00 %		L_GainOffsetP_1: Gain
C00677/2	0.00 %		L_GainOffsetP_1: Offset
C00677/3	100.00 %		L_GainOffsetP_2: Gain
C00677/4	0.00 %		L_GainOffsetP_2: Offset
C00677/5	100.00 %		L_GainOffsetP_3: Gain
C00677/6	0.00 %		L_GainOffsetP_3: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00678

Parameter | Name: **C00678 | L_GainOffsetPhiP 1-2: Parameter**			Data type: INTEGER_32 Index: 23897_d = $5D59_h$
Gain and offset			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C00678/1	65536		L_GainOffsetPhiP_1: Gain
C00678/2	0		L_GainOffsetPhiP_1: Offset
C00678/3	65536		L_GainOffsetPhiP_2: Gain
C00678/4	0		L_GainOffsetPhiP_2: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00679

Parameter | Name: **C00679 | L_MulDiv_2: Parameter**			Data type: INTEGER_16 Index: 23896_d = $5D58_h$
The L_MulDiv_2 FB: Numerator and denominator			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C00679/1	0		L_MulDiv_2: Numerator
C00679/2	10000		L_MulDiv_2: Denominator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00680

Parameter | Name: **C00680 | L_Compare_1: Fct.**		Data type: UNSIGNED_8 Index: 23895_d = $5D57_h$
The L_Compare_1 FB: Comparison operation • If the statement of the selected comparison operation is true, the binary *bOut* output will be set to TRUE.		
Selection list(Lenze setting printed in bold)		
1	In1 = In2	
2	In1 > In2	
3	In1 < In2	
4	|In1| = |In2|	
5	|In1| > |In2|	
6	**|In1| < |In2|**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00681

Parameter | Name: **C00681 | L_Compare_1: Hysteresis**			Data type: INTEGER_16 Index: 23894_d = $5D56_h$
The L_Compare_1 FB: Hysteresis for the comparison function selected in C00680			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**0.50 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00682

Parameter | Name: **C00682 | L_Compare_1: Window**			Data type: INTEGER_16 Index: 23893_d = $5D55_h$
The L_Compare_1 FB: Window for the comparison function selected in C00680			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**2.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00685

Parameter | Name: **C00685 | L_Compare_2: Fct.**		Data type: UNSIGNED_8 Index: 23890_d = $5D52_h$
The L_Compare_2 FB: Comparison operation • If the statement of the selected comparison operation is true, the binary *bOut* output will be set to TRUE.		
Selection list(Lenze setting printed in bold)		
1	In1 = In2	
2	In1 > In2	
3	In1 < In2	
4	**|In1| = |In2|**	
5	|In1| > |In2|	
6	|In1| < |In2|	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00686

Parameter | Name: **C00686 | L_Compare_2: Hysteresis**			Data type: INTEGER_16 Index: 23889_d = $5D51_h$
The L_Compare_2 FB: Hysteresis for the comparison function selected in C00685			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**0.50 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00687

Parameter | Name: **C00687 | L_Compare_2: Window**			Data type: INTEGER_16 Index: 23888_d = $5D50_h$
The L_Compare_2 FB: Window for the comparison function selected in C00685			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**2.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00690

Parameter | Name: **C00690 | L_Compare_3: Function**		Data type: UNSIGNED_8 Index: 23885_d = $5D4D_h$
The L_Compare_3 FB: Comparison operation • If the statement of the selected comparison operation is true, the binary *bOut* output will be set to TRUE.		
Selection list(Lenze setting printed in bold)		
1	**In1 = In2**	
2	In1 > In2	
3	In1 < In2	
4	|In1| = |In2|	
5	|In1| > |In2|	
6	|In1| < |In2|	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00691

Parameter | Name: **C00691 | L_Compare_3: Hysteresis**			Data type: INTEGER_16 Index: 23884_d = $5D4C_h$
FB L_Compare_3: Hysteresis for the comparison operation selected in C00690			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00692

Parameter | Name: **C00692 | L_Compare_3: Window**			Data type: INTEGER_16 Index: 23883_d = $5D4B_h$
FB L_Compare_3: window for the comparison operation selected in C00690			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00693

Parameter | Name: **C00693 | L_Compare 4-5: Function**		Data type: UNSIGNED_8 Index: 23882_d = $5D4A_h$
Comparison operation • If the statement of the selected comparison operation is true, the binary *bOut* output will be set to TRUE.		
Selection list		
1	In1 = In2	
2	In1 > In2	
3	In1 < In2	
4	|In1| = |In2|	
5	|In1| > |In2|	
6	|In1| < |In2|	
Subcodes	**Lenze setting**	**Info**
C00693/1	1: In1 = In2	L_Compare_4: Function
C00693/2	1: In1 = In2	L_Compare_5: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00694

Parameter | Name: **C00694 | L_Compare 4-5: Hysteresis**			Data type: INTEGER_16 Index: 23881_d = $5D49_h$
Hysteresis for the comparison operation selected in C00693			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C00694/1	0.00 %		L_Compare_4: Hysteresis
C00694/2	0.00 %		L_Compare_5: Hysteresis
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00695

Parameter | Name: **C00695 | L_Compare 4-5: Window**			Data type: INTEGER_16 Index: 23880_d = $5D48_h$
Window for the comparison operation selected in C00693			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C00695/1	0.00 %		L_Compare_4: Window
C00695/2	0.00 %		L_Compare_5: Window
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00696

Parameter | Name: **C00696 | L_OffsetGainP_1: Offset**			Data type: INTEGER_16 Index: 23879_d = $5D47_h$
The L_OffsetGainP_1 FB: Offset (additive to the input signal)			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00697

Parameter | Name: **C00697 | L_OffsetGainP_2: Offset**			Data type: INTEGER_16 Index: 23878_d = $5D46_h$
The L_OffsetGainP_2 FB: Offset (additive to the input signal)			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00698

Parameter | Name: **C00698 | L_OffsetGainP_3: Offset**			Data type: INTEGER_16 Index: 23877_d = $5D45_h$
The L_OffsetGainP_3 FB: Offset (additive to the input signal)			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00699

Parameter | Name: **C00699 | L_MulDiv_1: Parameter**			Data type: INTEGER_16 Index: $23876_d = 5D44_h$
The L_MulDiv_1 FB: Numerator and denominator			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C00699/1	0		L_MulDiv_1: Numerator
C00699/2	10000		L_MulDiv_1: Denominator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00700

Parameter | Name: **C00700 | LA_NCtrl: Analog connection list**		Data type: UNSIGNED_16 Index: $23875_d = 5D43_h$
Connection parameters for "Actuating drive speed" application: 16-bit inputs • Selection of the 16 bit output signals to be connected to the 16 bit input signals • The selection list contains all 16 bit output signals which can be assigned to the 16 bit inputs displayed by the subcodes.		
Selection list		
See selection list - analog signals		
Subcodes	**Lenze setting**	**Info**
C00700/1	20005: LS_ParFix: wDriveCtrl	LA_NCtrl: wCANDriveControl Input for control word from CAN to device control
C00700/2	20005: LS_ParFix: wDriveCtrl	LA_NCtrl: wMCIDriveControl Input for control word from communication interface to device control
C00700/3	20012: LS_ParFree_a: nC472_3_a	LA_NCtrl: nTorqueMotLim_a Input for maximum torque in motor mode
C00700/4	20013: LS_ParFree_a: nC472_4_a	LA_NCtrl: nTorqueGenLim_a Input for maximum torque in generator mode
C00700/5	0: Not connected	LA_NCtrl: nPIDVpAdapt_a Input for adapting the PID controller gain
C00700/6	0: Not connected	LA_NCtrl: nPIDActValue_a Input for actual PID controller value
C00700/7	16000: LS_AnalogInput: nIn1_a	LA_NCtrl: nMainSetValue_a Input for main speed setpoint
C00700/8	0: Not connected	LA_NCtrl: nAuxSetValue_a Input for additional speed setpoint
C00700/9	0: Not connected	LA_NCtrl: nGPAnalogSwitchIn1_a Input for analog switch - analog signal 1
C00700/10	0: Not connected	LA_NCtrl: nGPAnalogSwitchIn2_a Input for analog switch - analog signal 2
C00700/11	0: Not connected	LA_NCtrl: nGPArithmetikIn1_a Input for arithmetic function - analog signal 1
C00700/12	0: Not connected	LA_NCtrl: nGPArithmetikIn2_a Input for arithmetic function - analog signal 2
C00700/13	0: Not connected	LA_NCtrl: nGPMulDivIn_a Input for analog signal for multiplication/division
C00700/14	0: Not connected	LA_NCtrl: nGPCompareIn1_a Input for comparison operation - analog signal 1

Parameter | Name: **C00700 | LA_NCtrl: Analog connection list**		Data type: UNSIGNED_16 Index: 23875_d = $5D43_h$
C00700/15	0: Not connected	LA_NCtrl: nGPCompareIn2_a Input for comparison operation - analog signal 2
C00700/16	0: Not connected	LA_NCtrl: nVoltageAdd_a Input for additive voltage boost
C00700/17	0: Not connected	LA_NCtrl: nPIDInfluence_a Input for influence signal of PID controller correcting variable
C00700/18	0: Not connected	LA_NCtrl: nPIDSetValue_a Input for PID controller setpoint
C00700/19	0: Not connected	LA_NCtrl: nPWMAngleOffset Input for pulse width modulation phase offset
C00700/20	0: Not connected	LA_NCtrl: nBoost_a Input for additional setpoint for motor voltage at speed = 0
C00700/21	0: Not connected	LA_NCtrl: wSMCtrl Interface to the optional safety system
C00700/22	0: Not connected	Reserved
C00700/23	0: Not connected	Reserved
C00700/24	0: Not connected	Reserved
C00700/25	0: Not connected	Reserved
C00700/26	0: Not connected	LA_NCtrl: wFreeIn1 Input for user signal 1
C00700/27	0: Not connected	LA_NCtrl: wFreeIn2 Input for user signal 2
C00700/28	0: Not connected	LA_NCtrl: wFreeIn3 Input for user signal 3
C00700/29	0: Not connected	LA_NCtrl: wFreeIn4 Input for user signal 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00701

Parameter | Name: **C00701 | LA_NCtrl: Digital connection list**		Data type: UNSIGNED_16 Index: 23874_d = $5D42_h$
Connection parameters for "Actuating drive speed" application: Binary inputs • Selection of the binary output signals to be connected to the binary input signals • The selection list contains all binary output signals which can be assigned to the binary inputs mapped by the subcodes.		
Selection list		
See selection list - digital signals		
Subcodes	**Lenze setting**	**Info**
C00701/1	0: Not connected	LA_NCtrl: bCInh Control input for setting controller inhibit
C00701/2	16008: LS_DigitalInput: bCInh	LA_NCtrl: bFailReset Control input for error acknowledgement
C00701/3	0: Not connected	LA_NCtrl: bSetQuickstop Control input for quick stop request
C00701/4	16002: LS_DigitalInput: bIn3	LA_NCtrl: bSetDCBrake Control input for DC-injection braking request

Parameter | Name: **C00701 | LA_NCtrl: Digital connection list**		Data type: UNSIGNED_16 Index: 23874_d = $5D42_h$
C00701/5	0: Not connected	LA_NCtrl: bRFG_Stop Control input for stopping the speed ramp function generator
C00701/6	0: Not connected	LA_NCtrl: bRFG_0 Control input for setting the speed ramp function generator to 0
C00701/7	0: Not connected	Reserved
C00701/8	16003: LS_DigitalInput: bIn4	LA_NCtrl: bSetSpeedCcw Control input for change of direction of rotation
C00701/9	16000: LS_DigitalInput: bIn1	LA_NCtrl: bJogSpeed1 Selection input for fixed setpoints
C00701/10	16001: LS_DigitalInput: bIn2	LA_NCtrl: bJogSpeed2 Selection input for fixed setpoints
C00701/11	0: Not connected	LA_NCtrl: bJogSpeed4 Selection input for fixed setpoints
C00701/12	0: Not connected	LA_NCtrl: bJogSpeed8 Selection input for fixed setpoints
C00701/13	0: Not connected	LA_NCtrl: bJogRamp1 Selection input for additional acceleration/deceleration times
C00701/14	0: Not connected	LA_NCtrl: bJogRamp2 Selection input for additional acceleration/deceleration times
C00701/15	0: Not connected	LA_NCtrl: bJogRamp4 Selection input for additional acceleration/deceleration times
C00701/16	0: Not connected	LA_NCtrl: bJogRamp8 Selection input for additional acceleration/deceleration times
C00701/17	0: Not connected	LA_NCtrl: bMPOTInAct Control input for deactivation of motor potentiometer
C00701/18	0: Not connected	LA_NCtrl: bMPOTUp Control input for motor potentiometer ramp-up
C00701/19	0: Not connected	LA_NCtrl: bMPOTDown Control input for motor potentiometer ramp-down
C00701/20	0: Not connected	LA_NCtrl: bMBRKRelease Control input for manual holding brake release request
C00701/21	0: Not connected	LA_NCtrl: bGPFree1 Input for binary user signal
C00701/22	0: Not connected	LA_NCtrl: bGPFree2 Input for binary user signal
C00701/23	0: Not connected	LA_NCtrl: bGPAnalogSwitchSet Control input for analog-value selector change-over
C00701/24	0: Not connected	LA_NCtrl: bGPDigitalDelayIn Input for digital signal with time delay
C00701/25	0: Not connected	LA_NCtrl: bGPLogicIn1 Input signal 1 for digital logic
C00701/26	0: Not connected	LA_NCtrl: bGPLogicIn2 Input signal 2 for digital logic
C00701/27	0: Not connected	LA_NCtrl: bGPLogicIn3 Input signal 3 for digital logic

Parameter | Name: **C00701 | LA_NCtrl: Digital connection list**		Data type: UNSIGNED_16 Index: 23874_d = $5D42_h$
C00701/28	0: Not connected	LA_NCtrl: bGPDFlipFlopInD Control input for DFlipFlop setting signal
C00701/29	0: Not connected	LA_NCtrl: bGPDFlipFlopInClk Control input for DFlipFlop clock signal
C00701/30	0: Not connected	LA_NCtrl: bGPDFlipFlopInClr Control input for DFlipFlop reset signal
C00701/31	0: Not connected	LA_NCtrl: bMPotEnable Control input for activation of motor potentiometer
C00701/32	0: Not connected	LA_NCtrl: bPIDEnableInfluenceRamp Control input for activation of influence of output correcting variable of PID controller
C00701/33	0: Not connected	LA_NCtrl: bPIDIOff Control input for deactivation of PID controller I component
C00701/34	20000: LS_ParFix: bTrue	LA_NCtrl: bRLQCw Control input for activation of CW direction of rotation of speed setpoint
C00701/35	0: Not connected	LA_NCtrl: bRLQCcw Control input for activation of CCW direction of rotation of speed setpoint
C00701/36	0: Not connected	Reserved
C00701/37	0: Not connected	Reserved
C00701/38	0: Not connected	Reserved
C00701/39	0: Not connected	Reserved
C00701/40	0: Not connected	Reserved
C00701/41	0: Not connected	LA_NCtrl: bFreeIn1 Input for binary user signal 1
C00701/42	0: Not connected	LA_NCtrl: bFreeIn2 Input for binary user signal 2
C00701/43	0: Not connected	LA_NCtrl: bFreeIn3 Input for binary user signal 3
C00701/44	0: Not connected	LA_NCtrl: bFreeIn4 Input for binary user signal 4
C00701/45	0: Not connected	LA_NCtrl: bFreeIn5 Input for binary user signal 5
C00701/46	0: Not connected	LA_NCtrl: bFreeIn6 Input for binary user signal 6
C00701/47	0: Not connected	LA_NCtrl: bFreeIn7 Input for binary user signal 7
C00701/48	0: Not connected	LA_NCtrl: bFreeIn8 Input for binary user signal 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00705

Parameter | Name: **C00705 | LA_NCtrl_Out: Analog signal list**	Data type: UNSIGNED_16 Index: 23870_d = $5D3E_h$
This code is used device-internally and must not be written by the user side!	

C00706

Parameter | Name: C00706 | LA_NCtrl_Out digital signal list	Data type: UNSIGNED_16 Index: 23869_d = $5D3D_h$
This code is used device-internally and must not be written by the user side!	

C00710

Parameter | Name: C00710 | LA_TabPos: Analog connection list		Data type: UNSIGNED_16 Index: 23865_d = $5D39_h$
Connection parameters for "Table positioning" application: 16-bit inputs • Selection of the 16 bit output signals to be connected to the 16 bit input signals • The selection list contains all 16 bit output signals which can be assigned to the 16 bit inputs displayed by the subcodes.		
Selection list		
See selection list - analog signals		
Subcodes	**Lenze setting**	**Info**
C00710/1	0: Not connected	LA_TabPos: wCanDriveControl Input for control word from CAN to device control
C00710/2	0: Not connected	LA_TabPos: wMciDriveControl Input for control word from communication interface to device control
C00710/3	0: Not connected	LA_TabPos: nTorqueMotLim_a Input for maximum torque in motor mode
C00710/4	0: Not connected	LA_TabPos: nTorqueGenLim_a Input for maximum torque in generator mode
C00710/5	0: Not connected	LA_TabPos: nMainSetValue_a Input for main speed setpoint
C00710/6	0: Not connected	LA_TabPos: nAuxSetValue_a Input for additional speed setpoint
C00710/7	0: Not connected	LA_TabPos: wMckCtrl1 Input for MCK control word 1
C00710/8	0: Not connected	LA_TabPos: wMckCtrl2 Input for MCK control word 2
C00710/9	0: Not connected	LA_TabPos: wMckOperationMode Input for selection of MCK operating mode
C00710/10	0: Not connected	LA_TabPos: wPosProfileMode Input for selection of MCK positioning mode in positioning mode
C00710/11	0: Not connected	LA_TabPos: wPosProfileNo Input for MCK positioning profile number in positioning mode
C00710/12	0: Not connected	LA_TabPos: nGPAnalogSwitchIn1_a Input for analog switch - analog signal 1
C00710/13	0: Not connected	LA_TabPos: nGPAnalogSwitchIn2_a Input for analog switch - analog signal 2
C00710/14	0: Not connected	LA_TabPos: nGPArithmetikIn1_a Input for arithmetic function - analog signal 1
C00710/15	0: Not connected	LA_TabPos: nGPArithmetikIn2_a Input for arithmetic function - analog signal 2
C00710/16	0: Not connected	LA_TabPos: nGPMulDivIn_a Input for analog signal for multiplication/division
C00710/17	0: Not connected	LA_TabPos: nGPCompareIn1_a Input for comparison operation - analog signal 1

Parameter | Name: **C00710 | LA_TabPos: Analog connection list**		Data type: UNSIGNED_16 Index: 23865_d = $5D39_h$
C00710/18	0: Not connected	LA_TabPos: nGPCompareIn2_a Input for comparison operation - analog signal 2
C00710/19	0: Not connected	LA_TabPos: wGPCounter1LdVal Input for load value for counter module 1
C00710/20	0: Not connected	LA_TabPos: wGPCounter1CmpVal Input for comparison value for counter module 1
C00710/21	0: Not connected	LA_TabPos: nSpeedOverride_a Input for speed override
C00710/22	0: Not connected	LA_TabPos: nAccOverride_a Input for acceleration override
C00710/23	0: Not connected	LA_TabPos: wFreeIn1 Input for user signal 1
C00710/24	0: Not connected	LA_TabPos: wFreeIn2 Input for user signal 2
C00710/25	0: Not connected	LA_TabPos: wFreeIn3 Input for user signal 3
C00710/26	0: Not connected	LA_TabPos: wFreeIn4 Input for user signal 4
C00710/27	0: Not connected	LA_TabPos: nPosCtrlOutLimit Input for correcting variable limitation of position controller
C00710/28	0: Not connected	LA_TabPos: nPosCtrlPAdapt Input for adapting the position controller gain
C00710/29	0: Not connected	LA_TabPos: wSMCtrl Interface to the optional safety system
C00710/30	0: Not connected	LA_TabPos: wPosProfileUnitsLW Input for selecting the target position in [units], Low-Word
C00710/31	0: Not connected	LA_TabPos: wPosProfileUnitsHW Input for selecting the target position in [units], High-Word
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00711

Parameter | Name: **C00711 | LA_TabPos: Digital connection list**		Data type: UNSIGNED_16 Index: 23864_d = $5D38_h$
Connection parameters for "Table positioning" application: Binary inputs • Selection of the binary output signals to be connected to the binary input signals • The selection list contains all binary output signals which can be assigned to the binary inputs mapped by the subcodes.		
Selection list		
See selection list - digital signals		
Subcodes	**Lenze setting**	**Info**
C00711/1	0: Not connected	LA_TabPos: bCInh Control input for setting controller inhibit
C00711/2	0: Not connected	LA_TabPos: bFailReset Control input for error acknowledgement
C00711/3	0: Not connected	LA_TabPos: bSetQuickstop Control input for quick stop request
C00711/4	0: Not connected	LA_TabPos: bSetSpeedCcw Control input for negation of speed direction

Parameter | Name: **C00711 | LA_TabPos: Digital connection list**		Data type: UNSIGNED_16 Index: 23864_d = $5D38_h$
C00711/5	0: Not connected	LA_TabPos: bJogSpeed1 Control input for fixed speed selection value 1
C00711/6	0: Not connected	LA_TabPos: bJogSpeed2 Control input for fixed speed selection value 2
C00711/7	0: Not connected	LA_TabPos: bMPotEnable Control input for activation of motor potentiometer
C00711/8	0: Not connected	LA_TabPos: bMPotUp Control input for motor potentiometer ramp-up
C00711/9	0: Not connected	LA_TabPos: bMPotDown Control input for motor potentiometer ramp-down
C00711/10	0: Not connected	LA_TabPos: bMBrakeRelease Control input for manual holding brake release request
C00711/11	0: Not connected	LA_TabPos: bPosCtrlOn Control input for activation of position controller
C00711/12	0: Not connected	LA_TabPos: bLimitSwitchPos Input for positive hardware limit switch signal
C00711/13	0: Not connected	LA_TabPos: bLimitSwitchNeg Input for negative hardware limit switch signal
C00711/14	0: Not connected	LA_TabPos: bReleaseLimitSwitch Control input for hardware limit switch retracting request
C00711/15	0: Not connected	LA_TabPos: bManJogPos Control input for manual jog in positive direction request
C00711/16	0: Not connected	LA_TabPos: bManJogNeg Control input for manual jog in negative direction request
C00711/17	0: Not connected	LA_TabPos: bManEnable2ndSpeed Control input for activation of the second manual speed for manual jog
C00711/18	0: Not connected	LA_TabPos: bEnableSpeedOverride Control input for activation of speed override
C00711/19	0: Not connected	LA_TabPos: bEnableAccOverride Control input for activation of acceleration override
C00711/20	0: Not connected	LA_TabPos: bHomeStartStop Control input for start/stop homing
C00711/21	0: Not connected	LA_TabPos: bHomeSetPosition Control input for setting the home position
C00711/22	0: Not connected	LA_TabPos: bHomeResetPosition Control input for reset of "Home position known"
C00711/23	0: Not connected	LA_TabPos: bHomeMark Input for pre-stop signal for homing
C00711/24	0: Not connected	LA_TabPos: bPosSetProfilePosition Control input for accepting the profile position in the profile data set
C00711/25	0: Not connected	LA_TabPos: bPosSetActualPosition Control input for accepting the current position in the profile data set
C00711/26	0: Not connected	LA_TabPos: bPosExecute Control input for positioning start
C00711/27	0: Not connected	LA_TabPos: bPosFinishTarget Control input for completion of positioning to target position

Parameter | Name: **C00711 | LA_TabPos: Digital connection list**		Data type: UNSIGNED_16 Index: 23864_d = $5D38_h$
C00711/28	0: Not connected	LA_TabPos: bPosDisableFollowProfile Control input for suppression of sequence profile linkage
C00711/29	0: Not connected	LA_TabPos: bPosStop Control input for interruption of profile generation by ramp-down procedure
C00711/30	0: Not connected	LA_TabPos: bGPAnalogSwitchSet Control input for analog-value selector change-over
C00711/31	0: Not connected	LA_TabPos: bGPDigitalDelayIn Input for digital signal with time delay
C00711/32	0: Not connected	LA_TabPos: bGPLogicIn1 Input signal 1 for digital logic
C00711/33	0: Not connected	LA_TabPos: bGPLogicIn2 Input signal 2 for digital logic
C00711/34	0: Not connected	LA_TabPos: bGPLogicIn3 Input signal 3 for digital logic
C00711/35	0: Not connected	LA_TabPos: bGPDFlipFlop_InD Control input for DFlipFlop setting signal
C00711/36	0: Not connected	LA_TabPos: bGPDFlipFlop_InClk Control input for DFlipFlop clock signal
C00711/37	0: Not connected	LA_TabPos: bGPDFlipFlop_InClr Control input for DFlipFlop reset signal
C00711/38	0: Not connected	LA_TabPos: bGPCounter1ClkUp Control input for up-counting counter module 1
C00711/39	0: Not connected	LA_TabPos: bGPCounter1ClkDown Control input for down-counting counter module 1
C00711/40	0: Not connected	LA_TabPos: bGPCounter1Load Control input for load value acceptance in counter module 1
C00711/41	0: Not connected	LA_TabPos: bMckOperationMode_1 Control input for MCK operating mode changeover value 1
C00711/42	0: Not connected	LA_TabPos: bMckOperationMode_2 Control input for MCK operating mode changeover value 2
C00711/43	0: Not connected	LA_TabPos: bMckOperationMode_4 Control input for MCK operating mode changeover value 4
C00711/44	0: Not connected	LA_TabPos: bMckOperationMode_8 Control input for MCK operating mode changeover value 8
C00711/45	0: Not connected	LA_TabPos: bPosProfileNo_1 Control input for selection of profile number value 1
C00711/46	0: Not connected	LA_TabPos: bPosProfileNo_2 Control input for selection of profile number value 2
C00711/47	0: Not connected	LA_TabPos: bPosProfileNo_4 Control input for selection of profile number value 4
C00711/48	0: Not connected	LA_TabPos: bPosProfileNo_8 Control input for selection of profile number value 8
C00711/49	0: Not connected	LA_TabPos: bFreeIn1 Input for binary user signal 1
C00711/50	0: Not connected	LA_TabPos: bFreeIn2 Input for binary user signal 2

Parameter | Name: **C00711 | LA_TabPos: Digital connection list**		Data type: UNSIGNED_16 Index: 23864_d = $5D38_h$
C00711/51	0: Not connected	LA_TabPos: bFreeIn3 Input for binary user signal 3
C00711/52	0: Not connected	LA_TabPos: bFreeIn4 Input for binary user signal 4
C00711/53	0: Not connected	LA_TabPos: bFreeIn5 Input for binary user signal 5
C00711/54	0: Not connected	LA_TabPos: bFreeIn6 Input for binary user signal 6
C00711/55	0: Not connected	LA_TabPos: bFreeIn7 Input for binary user signal 7
C00711/56	0: Not connected	LA_TabPos: bFreeIn8 Input for binary user signal 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00712

Parameter | Name: **C00712 | LA_TabPos: phi connection list**		Data type: UNSIGNED_16 Index: 23863_d = $5D37_h$
Connection parameters for "Table positioning" application: 32-bit inputs • Selection of the 32-bit output signals for connection with the 32-bit input signals • The selection list contains all 32-bit output signals which can be assigned to the 32-bit inputs mapped by the subcodes.		
Selection list		
See selection list - angle signals		
Subcodes	**Lenze setting**	**Info**
C00712/1	0: Not connected	LA_TabPos: dnPosProfilePosition Input for selecting the target position in [increments]
C00712/2	0: Not connected	LA_TabPos: dnFreeIn1 Input for 32-bit user signal 1
C00712/3	0: Not connected	LA_TabPos: dnFreeIn2 Input for 32-bit user signal 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00715

Parameter | Name: **C00715 | LA_TabPos_Out: Analog signal list**	Data type: UNSIGNED_16 Index: 23860_d = $5D34_h$
This code is used device-internally and must not be written by the user side!	

C00716

Parameter | Name: **C00716 | LA_TabPos_Out: Digital signal list**	Data type: UNSIGNED_16 Index: 23859_d = $5D33_h$
This code is used device-internally and must not be written by the user side!	

C00717

Parameter | Name: **C00717 | LA_TabPos_Out: phi signal list**	Data type: UNSIGNED_16 Index: 23858_d = $5D32_h$
This code is used device-internally and must not be written by the user side!	

C00720

Parameter | Name: **C00720 | L_DigitalDelay_1: Delay**			Data type: UNSIGNED_32 Index: 23855_d = $5D2F_h$
Switch-on/off delay time			
Setting range (min. value | unit | max. value)			
0.000	s	3600.000	
Subcodes	**Lenze setting**		**Info**
C00720/1	0.000 s		L_DigitalDelay_1: ON delay
C00720/2	0.000 s		L_DigitalDelay_1: OFF delay
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00721

Parameter | Name: **C00721 | L_DigitalDelay 2..3: Delay**			Data type: UNSIGNED_32 Index: 23854_d = $5D2E_h$
Switch-on/off delay time			
Setting range (min. value | unit | max. value)			
0.000	s	3600.000	
Subcodes	**Lenze setting**		**Info**
C00721/1	0.000 s		L_DigitalDelay_2: ON delay
C00721/2	0.000 s		L_DigitalDelay_2: OFF delay
C00721/3	0.000 s		L_DigitalDelay_3: ON delay
C00721/4	0.000 s		L_DigitalDelay_3: OFF delay
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00725

Parameter | Name: **C00725 | Current switching frequency**	Data type: UNSIGNED_8 Index: 23850_d = $5D2A_h$
Display of the current switching frequency • In C00018 you can choose between a drive-optimised setting for good smooth-running characteristics and an inverter loss-optimised setting (min. Pv). • Both possibilities offer fixed and variable switching frequencies. • When a variable switching frequency is selected in C00018, the switching frequency may change as a function of the load and rotational frequency.	

Selection list (read only)	
1	4 kHz var./drive-optimised
2	8 kHz var./drive-optimised
3	16 kHz var./drive-optimised
5	2 kHz constant/drive-optimised
6	4 kHz constant/drive-optimised
7	8 kHz constant/drive-optimised
8	16 kHz constant/drive-optimised
11	4 kHz var./min. Pv
12	8 kHz var./min. Pv
13	16 kHz var./min. Pv
14	Reserved
15	2 kHz constant/min. Pv
16	4 kHz constant/min. Pv
17	8 kHz constant/min. Pv
18	16 kHz constant/min. Pv
21	8 kHz var./drive-opt./4 kHz min
22	16 kHz var./drive-opt./4 kHz min
23	16 kHz var./drive-opt./8 kHz min
31	8 kHz var./min. Pv/4 kHz min
32	16 kHz var./min. Pv/4 kHz min
33	16 kHz var./min. Pv/8 kHz min

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1

C00726

Parameter | Name: **C00726 | Current limit values**	Data type: UNSIGNED_8 Index: 23849_d = $5D29_h$
This code is used device-internally and must not be written by the user side!	

C00727

Parameter | Name: **C00727 | LS_Keypad digital values**			Data type: UNSIGNED_8 Index: 23848_d = $5D28_h$
Execution of control commands for keypad operation			
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Info**
C00727/1	0		1 ≡ request quick stop
C00727/2	0		1 ≡ request DC-injection braking
C00727/3	0		1 ≡ request change of direction of rotation
C00727/4	0		1 ≡ request fixed speed setpoint 1
C00727/5	0		1 ≡ request fixed speed setpoint 2
C00727/6	0		1 ≡ motor potentiometer: request activation
C00727/7	0		1 ≡ motor potentiometer: request pos. acceleration
C00727/8	0		1 ≡ motor potentiometer: request neg. acceleration
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00728

Parameter | Name: **C00728 | Keypad analog values**			Data type: INTEGER_16 Index: 23847_d = $5D27_h$
Selection of different setpoints when operating via keypad			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00728/1	100.00 %		Torque limit in motor mode
C00728/2	100.00 %		Torque limit in generator mode
C00728/3	0.00 %		Setpoint speed • 100% ≡ C00011
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00729

Parameter | Name: **C00729 | Remote: Setpoint selection**			Data type: INTEGER_16 Index: 23846_d = $5D26_h$
Setting range (min. value | unit | max. value)			
-199.99		199.99	
Subcodes	**Lenze setting**		**Info**
C00729/1	0.00		Remote: Setpoint keypad
C00729/2	0.00		Remote: Setpoint PC
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00730

Parameter | Name: **C00730 | Oscilloscope scanning interval**	Data type: UNSIGNED_32 Index: 23845_d = $5D25_h$
This code is used device-internally and must not be written by the user side!	

C00731

Parameter | Name: **C00731 | Oscilloscope recording length**	Data type: UNSIGNED_32 Index: 23844_d = $5D24_h$
This code is used device-internally and must not be written by the user side!	

C00732

Parameter | Name: **C00732 | Oscilloscope command**	Data type: UNSIGNED_8 Index: 23843_d = $5D23_h$
This code is used device-internally and must not be written by the user side!	

C00734

Parameter | Name: **C00734 | Oscilloscope trigger channel selection**	Data type: UNSIGNED_16 Index: 23841_d = $5D21_h$
This code is used device-internally and must not be written by the user side!	

C00735

Parameter | Name: **C00735 | Oscilloscope channel source type**	Data type: UNSIGNED_8 Index: 23840_d = $5D20_h$
This code is used device-internally and must not be written by the user side!	

C00736

Parameter | Name: **C00736 | Oscilloscope data type/data width**	Data type: UNSIGNED_16 Index: 23839_d = $5D1F_h$
This code is used device-internally and must not be written by the user side!	

C00740

Parameter | Name: **C00740 | Oscilloscope offset variables**	Data type: UNSIGNED_32 Index: 23835_d = $5D1B_h$
This code is used device-internally and must not be written by the user side!	

C00741

Parameter | Name: **C00741 | Oscilloscope trigger mode**	Data type: UNSIGNED_8 Index: 23834_d = $5D1A_h$
This code is used device-internally and must not be written by the user side!	

C00742

Parameter | Name: **C00742 | Oscilloscope trigger delay**	Data type: INTEGER_16 Index: 23833_d = $5D19_h$
This code is used device-internally and must not be written by the user side!	

C00743

Parameter | Name: **C00743 | Oscilloscope trigger level**	Data type: UNSIGNED_32 Index: 23832_d = $5D18_h$
This code is used device-internally and must not be written by the user side!	

C00744

Parameter | Name: **C00744 | Oscilloscope trigger mask**	Data type: UNSIGNED_32 Index: 23831_d = $5D17_h$
This code is used device-internally and must not be written by the user side!	

C00746

Parameter | Name: **C00746 | Oscilloscope trigger counter**	Data type: UNSIGNED_32 Index: 23829_d = $5D15_h$
This code is used device-internally and must not be written by the user side!	

C00747

Parameter | Name: **C00747 | Oscilloscope status word**	Data type: UNSIGNED_16 Index: 23828_d = $5D14_h$
This code is used device-internally and must not be written by the user side!	

C00748

Parameter | Name: **C00748 | Oscilloscope no. of measured values**	Data type: UNSIGNED_32 Index: 23827_d = $5D13_h$
This code is used device-internally and must not be written by the user side!	

C00749

Parameter | Name: **C00749 | Oscilloscope recording**	Data type: UNSIGNED_32 Index: 23826_d = $5D12_h$
This code is used device-internally and must not be written by the user side!	

C00750

Parameter | Name: **C00750 | Select. of MCTRL oscillos. channels**	Data type: UNSIGNED_8 Index: 23825_d = $5D11_h$
This code is used device-internally and must not be written by the user side!	

C00751

Parameter | Name: **C00751 | Oscilloscope data memory**	Data type: UNSIGNED_32 Index: 23824_d = $5D10_h$
This code is used device-internally and must not be written by the user side!	

C00753

Parameter | Name: **C00753 | Oscilloscope data memory octet string**	Data type: OCTET_STRING Index: 23822_d = $5D0E_h$
This code is used device-internally and must not be written by the user side!	

C00759

Parameter | Name: **C00759 | Oscilloscope: start after mains power ON**		Data type: UNSIGNED_8 Index: 23816_d = $5D08_h$
Selection list		**Info**
0	No	Oscilloscope does not start automatically after switching on the device.
1	Yes	Oscilloscope starts automatically after switching on the device.

C00760

Parameter | Name: **C00760 | LA_SwitchPos: Analog connection list**		Data type: UNSIGNED_16 Index: 23815_d = $5D07_h$
Connection parameters for "Switch-off positioning" application: 16-bi inputs • Selection of the 16 bit output signals to be connected to the 16 bit input signals • The selection list contains all 16 bit output signals which can be assigned to the 16 bit inputs displayed by the subcodes.		
Selection list		
See selection list - analog signals		
Subcodes	**Lenze setting**	**Info**
C00760/1	0: Not connected	LA_SwitchPos: wCANDriveControl Input for control word from CAN to device control
C00760/2	0: Not connected	LA_SwitchPos: wMCIDriveControl Input for control word from communication interface to device control
C00760/3	0: Not connected	LA_SwitchPos: nVoltageAdd_a Input for additive voltage impression
C00760/4	0: Not connected	LA_SwitchPos: nBoost_a Input for additional setpoint for motor voltage at speed = 0
C00760/5	0: Not connected	LA_SwitchPos: nPWMAngleOffset Input for additional offset for the electrical angle of rotation
C00760/6	0: Not connected	LA_SwitchPos: nTorqueMotLim_a Input for maximum torque in motor mode
C00760/7	0: Not connected	LA_SwitchPos: nTorqueGenLim_a Input for maximum torque in generator mode
C00760/8	0: Not connected	LA_SwitchPos: nMainSetValue_a Input for main speed setpoint
C00760/9	0: Not connected	LA_SwitchPos: nAuxSetValue_a Input for additional speed setpoint
C00760/10	0: Not connected	LA_SwitchPos: nGPAnalogSwitchIn1_a Input for analog switch - analog signal 1
C00760/11	0: Not connected	LA_SwitchPos: nGPAnalogSwitchIn2_a Input for analog switch - analog signal 2
C00760/12	0: Not connected	LA_SwitchPos: nGPArithmetikIn1_a Input for arithmetic function - analog signal 1
C00760/13	0: Not connected	LA_SwitchPos: nGPArithmetikIn2_a Input for arithmetic function - analog signal 2
C00760/14	0: Not connected	LA_SwitchPos: nGPMulDivIn_a Input for analog signal for multiplication/division
C00760/15	0: Not connected	LA_SwitchPos: nGPCompareIn1_a Input for comparison operation - analog signal 1

Parameter | Name: **C00760 | LA_SwitchPos: Analog connection list**		Data type: UNSIGNED_16 Index: 23815_d = $5D07_h$
C00760/16	0: Not connected	LA_SwitchPos: nGPCompareIn2_a Input for comparison operation - analog signal 2
C00760/17	0: Not connected	LA_SwitchPos: wSMCtrl Interface to the optional safety system
C00760/18	0: Not connected	Reserved
C00760/19	0: Not connected	Reserved
C00760/20	0: Not connected	Reserved
C00760/21	0: Not connected	Reserved
C00760/22	0: Not connected	LA_SwitchPos: wFreeIn1 Input for user signal 1
C00760/23	0: Not connected	LA_SwitchPos: wFreeIn2 Input for user signal 2
C00760/24	0: Not connected	LA_SwitchPos: wFreeIn3 Input for user signal 3
C00760/25	0: Not connected	LA_SwitchPos: wFreeIn4 Input for user signal 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00761

Parameter | Name: **C00761 | LA_SwitchPos: Digital connection list**		Data type: UNSIGNED_16 Index: 23814_d = $5D06_h$
Connection parameters for "Switch-off positioning" application: Binary inputs • Selection of the binary output signals to be connected to the binary input signals • The selection list contains all binary output signals which can be assigned to the binary inputs mapped by the subcodes.		
Selection list		
See selection list - digital signals		
Subcodes	**Lenze setting**	**Info**
C00761/1	0: Not connected	LA_SwitchPos: bCInh Control input for setting controller inhibit
C00761/2	0: Not connected	LA_SwitchPos: bFailReset Control input for error acknowledgement
C00761/3	0: Not connected	LA_SwitchPos: bSetQuickstop Control input for quick stop request
C00761/4	0: Not connected	LA_SwitchPos: bSetDCBrake Control input for DC-injection braking request
C00761/5	0: Not connected	LA_SwitchPos: bRFG_Stop Control input for stopping the speed ramp function generator
C00761/6	0: Not connected	LA_SwitchPos: bSetSpeedCcw Control input for change of direction of rotation
C00761/7	0: Not connected	LA_SwitchPos: bRLQCw Control input for activation of CW rotation (fail-safe)
C00761/8	0: Not connected	LA_SwitchPos: bRLQCcw Control input for activation of CCW rotation (fail-safe)
C00761/9	0: Not connected	LA_SwitchPos: bJogCtrlInputSel1 Selection input 1 for binary coded selection of the switch-off position 1 ... 3

Parameter | Name: **C00761 | LA_SwitchPos: Digital connection list**		Data type: UNSIGNED_16 Index: 23814_d = $5D06_h$
C00761/10	0: Not connected	LA_SwitchPos: bJogCtrlInputSel2 Selection input 2 for binary coded selection of the switch-off position 1 ... 3
C00761/11	0: Not connected	LA_SwitchPos: bJogCtrlRfgIn Control input for setpoint generator ramp-down
C00761/12	0: Not connected	LA_SwitchPos: bJogCtrlJog1 Selection input 1 for overriding fixed setpoints (JOG setpoints) for the main setpoint
C00761/13	0: Not connected	LA_SwitchPos: bJogCtrlJog2 Selection input 2 for overriding fixed setpoints (JOG setpoints) for the main setpoint
C00761/14	0: Not connected	LA_SwitchPos: bJogCtrlSlowDown1 Control input for selection of pre-switch off 1
C00761/15	0: Not connected	LA_SwitchPos: bJogCtrlStop1 Control input for stop function 1
C00761/16	0: Not connected	LA_SwitchPos: bJogCtrlSlowDown2 Control input for selection of pre-switch off 2
C00761/17	0: Not connected	LA_SwitchPos: bJogCtrlStop2 Control input for stop function 2
C00761/18	0: Not connected	LA_SwitchPos: bJogCtrlSlowDown3 Control input for selection of pre-switch off 3
C00761/19	0: Not connected	LA_SwitchPos: bJogCtrlStop3 Control input for stop function 3
C00761/20	0: Not connected	LA_SwitchPos: bJogSpeed4 Selection input for fixed setpoints
C00761/21	0: Not connected	LA_SwitchPos: bJogSpeed8 Selection input for fixed setpoints
C00761/22	0: Not connected	LA_SwitchPos: bJogRamp1 Selection input for additional acceleration/deceleration times
C00761/23	0: Not connected	LA_SwitchPos: bJogRamp2 Selection input for additional acceleration/deceleration times
C00761/24	0: Not connected	LA_SwitchPos: bJogRamp4 Selection input for additional acceleration/deceleration times
C00761/25	0: Not connected	LA_SwitchPos: bJogRamp8 Selection input for additional acceleration/deceleration times
C00761/26	0: Not connected	LA_SwitchPos: bMBrkRelease Control input for manual holding brake release request
C00761/27	0: Not connected	LA_SwitchPos: bGPAnalogSwitchSet Control input for analog-value selector change-over
C00761/28	0: Not connected	LA_SwitchPos: bGPDigitalDelayIn Input for digital signal with time delay
C00761/29	0: Not connected	LA_SwitchPos: bGPLogicIn1 Input signal 1 for digital logic
C00761/30	0: Not connected	LA_SwitchPos: bGPLogicIn2 Input signal 2 for digital logic
C00761/31	0: Not connected	LA_SwitchPos: bGPLogicIn3 Input signal 3 for digital logic
C00761/32	0: Not connected	LA_SwitchPos: bGPDFlipFlop_InD Control input for DFlipFlop setting signal

Parameter | Name: **C00761 | LA_SwitchPos: Digital connection list**		Data type: UNSIGNED_16 Index: 23814_d = $5D06_h$
C00761/33	0: Not connected	LA_SwitchPos: bGPDFlipFlop_InClk Control input for DFlipFlop clock signal
C00761/34	0: Not connected	LA_SwitchPos: bGPDFlipFlop_InClr Control input for DFlipFlop reset signal
C00761/35	0: Not connected	Reserved
C00761/36	0: Not connected	Reserved
C00761/37	0: Not connected	Reserved
C00761/38	0: Not connected	Reserved
C00761/39	0: Not connected	Reserved
C00761/40	0: Not connected	LA_SwitchPos: bFreeIn1 Input for binary user signal 1
C00761/41	0: Not connected	LA_SwitchPos: bFreeIn2 Input for binary user signal 2
C00761/42	0: Not connected	LA_SwitchPos: bFreeIn3 Input for binary user signal 3
C00761/43	0: Not connected	LA_SwitchPos: bFreeIn4 Input for binary user signal 4
C00761/44	0: Not connected	LA_SwitchPos: bFreeIn5 Input for binary user signal 5
C00761/45	0: Not connected	LA_SwitchPos: bFreeIn6 Input for binary user signal 6
C00761/46	0: Not connected	LA_SwitchPos: bFreeIn7 Input for binary user signal 7
C00761/47	0: Not connected	LA_SwitchPos: bFreeIn8 Input for binary user signal 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00762

Parameter | Name: **C00762 | LA_SwitchPos: phi connection list**	Data type: UNSIGNED_16 Index: 23813_d = $5D05_h$
This code is used device-internally and must not be written by the user side!	

C00765

Parameter | Name: **C00765 | LA_SwitchPos_Out: Analog signal list**	Data type: UNSIGNED_16 Index: 23810_d = $5D02_h$
This code is used device-internally and must not be written by the user side!	

C00766

Parameter | Name: **C00766 | LA_SwitchPos_Out: Digital signal list**	Data type: UNSIGNED_16 Index: 23809_d = $5D01_h$
This code is used device-internally and must not be written by the user side!	

C00767

Parameter | Name: **C00767 | LA_SwitchPos_Out: phi signal list**	Data type: UNSIGNED_16 Index: 23808_d = $5D00_h$
This code is used device-internally and must not be written by the user side!	

C00800

Parameter | Name: **C00800 | L_MPot_1: Upper limit**			Data type: INTEGER_16 Index: 23775_d = $5CDF_h$
The L_MPot_1 FB: Upper limit of the motor potentiometer function			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**100.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00801

Parameter | Name: **C00801 | L_MPot_1: Lower limit**			Data type: INTEGER_16 Index: 23774_d = $5CDE_h$
The L_MPot_1 FB: Lower limit of the motor potentiometer function			
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**-100.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00802

Parameter | Name: **C00802 | L_MPot_1: Acceleration time**			Data type: UNSIGNED_16 Index: 23773_d = $5CDD_h$
The L_MPot_1 FB: Acceleration time of the motor potentiometer function			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.1	s	6000.0	**10.0 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C00803

Parameter | Name: **C00803 | L_MPot_1: Deceleration time**			Data type: UNSIGNED_16 Index: 23772_d = $5CDC_h$
The L_MPot_1 FB: Deceleration time of the motor potentiometer function			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.1	s	6000.0	**10.0 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C00804

Parameter | Name: **C00804 | L_MPot_1: Inactive fct.**		Data type: UNSIGNED_8 Index: 23771_d = $5CDB_h$
The L_MPot_1 FB: Selection of the response if the motor potentiometer is deactivated via input *bInAct*		
Selection list(Lenze setting printed in bold)		**Info**
0	**Retain value**	Keep output value
1	Deceleration to 0	Deceleration via ramp to 0
2	Deceleration to lower limit	Deceleration via ramp to the lower limit (C00801)
3	Without ramp to 0	Step change to 0
4	Without ramp to lower limit	Jump to lower limit (C00800)
5	Acceleration to upper limit	Acceleration via ramp to upper limit (C00800)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00805

Parameter | Name: **C00805 | L_MPot_1: Init fct.**		Data type: UNSIGNED_8 Index: 23770_d = $5CDA_h$
The L_MPot_1 FB: Selection of the response at device switch-on		
Selection list(Lenze setting printed in bold)		
0	**Load last value**	
1	Load lower limit	
2	Load 0	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00806

Parameter | Name: **C00806 | L_MPot_1: Use**		Data type: UNSIGNED_8 Index: 23769_d = $5CD9_h$
The L_MPot_1 FB: Use of the motor potentiometer		
Selection list(Lenze setting printed in bold)		**Info**
0	**No**	The motor potentiometer is not used. • The analog value applied to the *nIn_a* input is looped through without any changes to the *nOut_a* output.
1	Yes	The motor potentiometer is used. • The analog value applied at the *nIn_a* input is led via the motor potentiometer and provided at the *nOut_a* output.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00807

Parameter | Name: **C00807 | L_NLim_1: Max.SkipFrq.**			Data type: INTEGER_16 Index: 23768_d = $5CD8_h$
FB L_NLim_1: Maximum blocking frequencies			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00807/1	0.00 %		Maximum blocking frequency for zone 1
C00807/2	0.00 %		Maximum blocking frequency for zone 2
C00807/3	0.00 %		Maximum blocking frequency for zone 3
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00808

Parameter | Name: **C00808 | L_NLim_1: Min.SkipFrq.**			Data type: INTEGER_16 Index: 23767_d = $5CD7_h$
FB L_NLim_1: Minimum skip frequencies			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00808/1	0.00 %		Minimum blocking frequency for zone 1
C00808/2	0.00 %		Minimum blocking frequency for zone 2
C00808/3	0.00 %		Minimum blocking frequency for zone 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00809

Parameter | Name: **C00809 | L_NLim_2: Max.SkipFrq.**			Data type: INTEGER_16 Index: 23766_d = $5CD6_h$
FB L_NLim_2: Maximum skip frequencies			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00809/1	0.00 %		Maximum blocking frequency for zone 1
C00809/2	0.00 %		Maximum blocking frequency for zone 2
C00809/3	0.00 %		Maximum blocking frequency for zone 3
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00810

Parameter | Name: **C00810 | L_NLim_2: Min.SkipFrq.**			Data type: INTEGER_16 Index: 23765_d = $5CD5_h$
FB L_NLim_2: Minimum skip frequencies			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00810/1	0.00 %		Minimum blocking frequency for zone 1
C00810/2	0.00 %		Minimum blocking frequency for zone 2
C00810/3	0.00 %		Minimum blocking frequency for zone 3
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00811

Parameter | Name: **C00811 | L_NLim_1: Current output value**			Data type: INTEGER_16 Index: 23764_d = $5CD4_h$
From version 02.00.00			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00811/1			L_NLim_1: Current output value
C00811/2			L_NLim_2: Current output value
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00812

Parameter | Name: **C00812 | L_NLim: Current status**			Data type: UNSIGNED_16 Index: 23763_d = $5CD3_h$
From version 02.00.00			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	No blocking zone active		
Bit 1	Blocking zone 1 active		
Bit 2	Blocking zone 2 active		
Bit 3	Blocking zone 3 active		
Bit 4	Value in blocking zone		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes			**Info**
C00812/1			L_NLim_1: Current status
C00812/2			L_NLim_2: Current status
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00820

Parameter | Name: **C00820 | L_DigitalLogic_1: Function**		Data type: UNSIGNED_8 Index: 23755_d = $5CCB_h$
The L_DigitalLogic_1 FB: Selection of the internal logic function		
Selection list (Lenze setting printed in bold)		**Info**
0	**bOut = 0**	Constant value "FALSE"
1	bOut = 1	Constant value "TRUE"
2	bOut = bIn1 AND bIn2 AND bIn3	AND operation
3	bOut = bIn1 OR bIn2 OR bIn3	OR operation
4	bOut = f (truth table)	The truth table parameterised in C00821 is used.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00821

Parameter | Name: **C00821 | L_DigitalLogic_1: Truth table**		Data type: UNSIGNED_8 Index: 23754_d = $5CCA_h$
The L_DigitalLogic_1 FB: Parameterisation of the truth table		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C00821/1	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=0 0 0
C00821/2	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=0 0 1
C00821/3	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=0 1 0
C00821/4	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=0 1 1
C00821/5	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=1 0 0
C00821/6	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=1 0 1
C00821/7	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=1 1 0
C00821/8	0: FALSE	L_DigitalLogic_1: bOut: bIn3...bIn1=1 1 1
C00821/9	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=0 0 0
C00821/10	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=0 0 1
C00821/11	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=0 1 0
C00821/12	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=0 1 1
C00821/13	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=1 0 0
C00821/14	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=1 0 1
C00821/15	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=1 1 0
C00821/16	0: FALSE	L_DigitalLogic_1: bOut2: bIn3...bIn1=1 1 1
C00821/17	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=0 0 0
C00821/18	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=0 0 1
C00821/19	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=0 1 0
C00821/20	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=0 1 1
C00821/21	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=1 0 0
C00821/22	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=1 0 1
C00821/23	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=1 1 0
C00821/24	0: FALSE	L_DigitalLogic_1: bOut3: bIn3...bIn1=1 1 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00822

Parameter | Name: **C00822 | L_DigitalLogic_2: Function**		Data type: UNSIGNED_8 Index: 23753_d = $5CC9_h$
The L_DigitalLogic_2 FB: Selection of the internal logic function		
Selection list(Lenze setting printed in bold)		**Info**
0	**bOut = 0**	Constant value "FALSE"
1	bOut = 1	Constant value "TRUE"
2	bOut = bIn1 AND ... bIn3	AND operation
3	bOut = bIn1 OR ... bIn3	OR operation
4	bOut = f (truth table)	The truth table parameterised in C00823 is used.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00823

Parameter | Name: **C00823 | L_DigitalLogic_2: Truth table**		Data type: UNSIGNED_8 Index: 23752_d = $5CC8_h$
The L_DigitalLogic_2 FB: Parameterisation of the truth table		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C00823/1	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=0 0 0
C00823/2	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=0 0 1
C00823/3	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=0 1 0
C00823/4	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=0 1 1
C00823/5	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=1 0 0
C00823/6	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=1 0 1
C00823/7	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=1 1 0
C00823/8	0: FALSE	L_DigitalLogic_2: bOut: bIn3...bIn1=1 1 1
C00823/9	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=0 0 0
C00823/10	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=0 0 1
C00823/11	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=0 1 0
C00823/12	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=0 1 1
C00823/13	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=1 0 0
C00823/14	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=1 0 1
C00823/15	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=1 1 0
C00823/16	0: FALSE	L_DigitalLogic_2: bOut2: bIn3...bIn1=1 1 1
C00823/17	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=0 0 0
C00823/18	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=0 0 1
C00823/19	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=0 1 0
C00823/20	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=0 1 1
C00823/21	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=1 0 0
C00823/22	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=1 0 1
C00823/23	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=1 1 0
C00823/24	0: FALSE	L_DigitalLogic_2: bOut3: bIn3...bIn1=1 1 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00824

Parameter | Name: **C00824 | L_DigitalLogic5_1: Function**		Data type: UNSIGNED_8 Index: 23751_d = $5CC7_h$
FB L_DigitalLogic5_1: Selection of the interna logic operation		
Selection list(**Lenze setting printed in bold**)		**Info**
0	**bOut = 0**	
1	bOut = 1	
2	bOut = f (truth table)	The truth table parameterised in C00825 is used.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00825

Parameter | Name: **C00825 | L_DigitalLogic5_1: Truth table**		Data type: UNSIGNED_8 Index: 23750_d = $5CC6_h$
FB L_DigitalLogic5_1: Parameter setting of the truth table		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C00825/1	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 0 0 0
C00825/2	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 0 0 1
C00825/3	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 0 1 0
C00825/4	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 0 1 1
C00825/5	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 1 0 0
C00825/6	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 1 0 1
C00825/7	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 1 1 0
C00825/8	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 0 1 1 1
C00825/9	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 0 0 0
C00825/10	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 0 0 1
C00825/11	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 0 1 0
C00825/12	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 0 1 1
C00825/13	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 1 0 0
C00825/14	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 1 0 1
C00825/15	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 1 1 0
C00825/16	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=0 1 1 1 1
C00825/17	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 0 0 0
C00825/18	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 0 0 1
C00825/19	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 0 1 0
C00825/20	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 0 1 1
C00825/21	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 1 0 0
C00825/22	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 1 0 1
C00825/23	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 1 1 0
C00825/24	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 0 1 1 1
C00825/25	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 0 0 0
C00825/26	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 0 0 1
C00825/27	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 0 1 0
C00825/28	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 0 1 1
C00825/29	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 1 0 0
C00825/30	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 1 0 1
C00825/31	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 1 1 0
C00825/32	0: FALSE	L_DigitalLogic5_1: bIn5...bIn1=1 1 1 1 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00826

Parameter | Name: **C00826 | L_DigitalLogic5_2: Function**		Data type: UNSIGNED_8 Index: 23749_d = $5CC5_h$
FB L_DigitalLogic5_2: Selection of the interna logic operation		
Selection list(Lenze setting printed in bold)		**Info**
0	**bOut = 0**	
1	bOut = 1	
2	bOut = f (truth table)	The truth table parameterised in C00827 is used.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00827

Parameter | Name: **C00827 | L_DigitalLogic5_2: Truth table**		Data type: UNSIGNED_8 Index: 23748_d = $5CC4_h$
FB L_DigitalLogic5_2: Parameter setting of the truth table		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C00827/1	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 0 0 0
C00827/2	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 0 0 1
C00827/3	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 0 1 0
C00827/4	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 0 1 1
C00827/5	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 1 0 0
C00827/6	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 1 0 1
C00827/7	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 1 1 0
C00827/8	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 0 1 1 1
C00827/9	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 0 0 0
C00827/10	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 0 0 1
C00827/11	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 0 1 0
C00827/12	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 0 1 1
C00827/13	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 1 0 0
C00827/14	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 1 0 1
C00827/15	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 1 1 0
C00827/16	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=0 1 1 1 1
C00827/17	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 0 0 0
C00827/18	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 0 0 1
C00827/19	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 0 1 0
C00827/20	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 0 1 1
C00827/21	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 1 0 0
C00827/22	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 1 0 1
C00827/23	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 1 1 0
C00827/24	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 0 1 1 1
C00827/25	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 0 0 0
C00827/26	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 0 0 1
C00827/27	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 0 1 0
C00827/28	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 0 1 1

Parameter | Name: C00827 | L_DigitalLogic5_2: Truth table		Data type: UNSIGNED_8 Index: 23748_d = $5CC4_h$
C00827/29	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 1 0 0
C00827/30	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 1 0 1
C00827/31	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 1 1 0
C00827/32	0: FALSE	L_DigitalLogic5_2: bIn5...bIn1=1 1 1 1 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00828

Parameter | Name: C00828 | L_DigitalLogic_3: Function		Data type: UNSIGNED_8 Index: 23747_d = $5CC3_h$
From version 02.00.00 The L_DigitalLogic_3 FB: Selection of the internal logic function		
Selection list(Lenze setting printed in bold)		**Info**
0	**bOut = 0**	Constant value "FALSE"
1	bOut = 1	Constant value "TRUE"
2	bOut = bIn1 AND bIn2 AND bIn3	AND operation
3	bOut = bIn1 OR bIn2 OR bIn3	OR operation
4	bOut = f (truth table)	The truth table parameterised in C00829 is used.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00829

Parameter | Name: **C00829 | L_DigitalLogic_3: Truth table**		Data type: UNSIGNED_8 Index: 23746_d = $5CC2_h$
From version 02.00.00 The L_DigitalLogic_3 FB: Parameterisation of the truth table		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C00829/1	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=0 0 0
C00829/2	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=0 0 1
C00829/3	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=0 1 0
C00829/4	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=0 1 1
C00829/5	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=1 0 0
C00829/6	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=1 0 1
C00829/7	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=1 1 0
C00829/8	0: FALSE	L_DigitalLogic_3: bOut: bIn3...bIn1=1 1 1
C00829/9	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=0 0 0
C00829/10	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=0 0 1
C00829/11	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=0 1 0
C00829/12	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=0 1 1
C00829/13	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=1 0 0
C00829/14	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=1 0 1
C00829/15	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=1 1 0
C00829/16	0: FALSE	L_DigitalLogic_3: bOut2: bIn3...bIn1=1 1 1
C00829/17	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=0 0 0
C00829/18	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=0 0 1
C00829/19	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=0 1 0
C00829/20	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=0 1 1
C00829/21	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=1 0 0
C00829/22	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=1 0 1
C00829/23	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=1 1 0
C00829/24	0: FALSE	L_DigitalLogic_3: bOut3: bIn3...bIn1=1 1 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00830

Parameter | Name: **C00830 | 16-bit inputs [%]**			Data type: INTEGER_16 Index: 23745_d = $5CC1_h$
Display in percent of 16-bit input values of different blocks			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00830/1			L_Absolut_1: nIn_a
C00830/2			L_AddSub_1: nIn1_a
C00830/3			L_AddSub_1: nIn2_a
C00830/4			L_AddSub_1: nIn3_a

Parameter | Name: **C00830 | 16-bit inputs [%]**	Data type: INTEGER_16 Index: 23745_d = $5CC1_h$
C00830/5	L_OffsetGain_1: nIn_a
C00830/6	L_OffsetGain_1: nOffset_a
C00830/7	L_OffsetGain_1: nGain_a
C00830/8	L_Negation_1: nIn_a
C00830/9	L_GainOffset_1: nIn_a
C00830/10	L_GainOffset_1: nGain_a
C00830/11	L_GainOffset_1: nOffset_a
C00830/12	L_Arithmetik_1: nIn1_a
C00830/13	L_Arithmetik_1: nIn2_a
C00830/14	L_AnalogSwitch_1: nIn1_a
C00830/15	L_AnalogSwitch_1: nIn2_a
C00830/16	L_Compare_1: nIn1_a
C00830/17	L_Compare_1: nIn2_a
C00830/18	MCTRL: nTorqueLimitAdapt_a
C00830/19	Reserved
C00830/20	MCTRL: nPosCtrlPAdapt_a
C00830/21	MCTRL: nPosCtrlOutLimit_a
C00830/22	MCTRL: nSpeedSetValue_a
C00830/23	MCTRL: nSpeedLowLimit_a
C00830/24	MCTRL: nSpeedCtrlI_a
C00830/25	MCTRL: nSpeedCtrlPAdapt_a
C00830/26	MCTRL: nBoost_a
C00830/27	MCTRL: nTorqueSetValue_a
C00830/28	MCTRL: nTorqueGenLimit_a
C00830/29	MCTRL: nTorqueMotLimit_a
C00830/30	Reserved
C00830/31	MCTRL: nVoltageAdd_a
C00830/32	MCTRL: nPWMAngleOffset_a
C00830/33	L_NSet_1: nCInhVal_a
C00830/34	L_NSet_1: nNSet_a
C00830/35	L_NSet_1: nSet_a
C00830/36	L_NSet_1: nNAdd_a
C00830/37	DCTRL: wCANControl
C00830/38	DCTRL: wCCMControl
C00830/39	L_NLim_1: nIn_a
C00830/40	Reserved
C00830/41	L_Compare_2: nIn1_a
C00830/42	L_Compare_2: nIn2_a
C00830/43	L_Compare_3: nIn1_a
C00830/44	L_Compare_3: nIn2_a
C00830/45	L_AnalogSwitch_2: nIn1_a
C00830/46	L_AnalogSwitch_2: nIn2_a
C00830/47	L_AnalogSwitch_3: nIn1_a
C00830/48	L_AnalogSwitch_3: nIn2_a

Parameter | Name: **C00830 | 16-bit inputs [%]**	Data type: INTEGER_16 Index: 23745_d = $5CC1_h$
C00830/49	L_Arithmetik_2: nIn1_a
C00830/50	L_Arithmetik_2: nIn2_a
C00830/51	Reserved
C00830/52	Reserved
C00830/53	L_GainOffset_2: nIn_a
C00830/54	L_GainOffset_2: nGain_a
C00830/55	L_GainOffset_2: nOffset_a
C00830/56	L_OffsetGainP_1: nIn_a
C00830/57	L_OffsetGainP_2: nIn_a
C00830/58	L_OffsetGain_2: nIn_a
C00830/59	L_OffsetGain_2: nOffset_a
C00830/60	L_OffsetGain_2: nGain_a
C00830/61	L_PCTRL_1: nAct_a
C00830/62	L_PCTRL_1: nAdapt_a
C00830/63	L_PCTRL_1: nSet_a
C00830/64	L_PCTRL_1: nInfluence_a
C00830/65	MCK: nSpeedCtrlI_a
C00830/66	MCK: nPWMAngleOffset_a
C00830/67	Reserved
C00830/68	MCK: nMBrakeAddValue_a
C00830/69	MCK: nTorqueSetValue_a
C00830/70	MCK: nTorqueLimitAdapt_a
C00830/71	MCK: nSRampOverride_a
C00830/72	MCK: nSpeedSetValue_a
C00830/73	MCK: wMotionCtrl2
C00830/74	MCK: wMotionCtrl1
C00830/75	MCK: nSpeedOverride_a
C00830/76	MCK: nAccOverride_a
C00830/77	MCK: nSpeedAdd_v
C00830/78	MCK: wAuxCtrl
C00830/79	MCK: wSMCtrl
C00830/80	L_OffsetGainP_3: nIn_a
C00830/81	L_MPot_1: nIn_a
C00830/82	L_MulDiv_1: nIn_a
C00830/83	LS_DataAccess: wIn1 (Lenze-internal)
C00830/84	LS_DataAccess: wIn2 (Lenze-internal)
C00830/85	LS_DataAccess: wIn3 (Lenze-internal)
C00830/86	LS_DataAccess: wIn4 (Lenze-internal)
C00830/87	L_PT1_1: nIn_a
C00830/88	MCTRL: nSpeedHighLimit_a
C00830/89	L_PCTRL_1: nNSet_a

Parameter | Name: **C00830 | 16-bit inputs [%]**	Data type: INTEGER_16 Index: 23745_d = $5CC1_h$
C00830/90	L_PCTRL_1: nISet_a
C00830/91	L_Interpolator_1: nPhdIn_v
C00830/92	L_Interpolator_1: nNIn_a
C00830/93	Reserved
C00830/94	Reserved
C00830/95	Reserved
C00830/96	MCTRL: nInertiaAdapt_a
C00830/97	MCTRL: nSpeedSetValueInertia_a
C00830/98	MCK: nProcessIn1_a
C00830/99	MCK: nProcessIn2_a
C00830/100	MCK: nProcessIn3_a
C00830/101	MCK: nProcessIn4_a
C00830/102	MCK: nProcessIn5_a
C00830/103	MCK: nProcessIn6_a
C00830/104	MCK: nProcessIn7_a
C00830/105	MCK: nProcessIn8_a
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100	

C00831

Parameter | Name: **C00831 | 16-bit inputs**			Data type: UNSIGNED_16 Index: 23744_d = $5CC0_h$
Decimal/hexadecimal/bit coded display of 16 bit input values of various blocks			
Display area **(min. hex value | max. hex value)**			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
...	...		
Bit 15	Active		
Subcodes			**Info**
C00831/1			L_Absolut_1: nIn_a
C00831/2			L_AddSub_1: nIn1_a
C00831/3			L_AddSub_1: nIn2_a
C00831/4			L_AddSub_1: nIn3_a
C00831/5			L_OffsetGain_1: nIn_a
C00831/6			L_OffsetGain_1: nOffset_a
C00831/7			L_OffsetGain_1: nGain_a
C00831/8			L_Negation_1: nIn_a
C00831/9			L_GainOffset_1: nIn_a
C00831/10			L_GainOffset_1: nGain_a
C00831/11			L_GainOffset_1: nOffset_a
C00831/12			L_Arithmetik_1: nIn1_a
C00831/13			L_Arithmetik_1: nIn2_a
C00831/14			L_AnalogSwitch_1: nIn1_a
C00831/15			L_AnalogSwitch_1: nIn2_a

Parameter | Name: **C00831 | 16-bit inputs**	Data type: UNSIGNED_16 Index: 23744_d = $5CC0_h$
C00831/16	L_Compare_1: nIn1_a
C00831/17	L_Compare_1: nIn2_a
C00831/18	MCTRL: nTorqueLimitAdapt_a
C00831/19	Reserved
C00831/20	MCTRL: nPosCtrlPAdapt_a
C00831/21	MCTRL: nPosCtrlOutLimit_a
C00831/22	MCTRL: nSpeedSetValue_a
C00831/23	MCTRL: nSpeedLowLimit_a
C00831/24	MCTRL: nSpeedCtrlI_a
C00831/25	MCTRL: nSpeedCtrlPAdapt_a
C00831/26	MCTRL: nBoost_a
C00831/27	MCTRL: nTorqueSetValue_a
C00831/28	MCTRL: nTorqueGenLimit_a
C00831/29	MCTRL: nTorqueMotLimit_a
C00831/30	Reserved
C00831/31	MCTRL: nVoltageAdd_a
C00831/32	MCTRL: nPWMAngleOffset_a
C00831/33	L_NSet_1: nCInhVal_a
C00831/34	L_NSet_1: nNSet_a
C00831/35	L_NSet_1: nSet_a
C00831/36	L_NSet_1: nNAdd_a
C00831/37	DCTRL: wCANControl
C00831/38	DCTRL: wMCIControl
C00831/39	L_NLim_1: nIn_a
C00831/40	Reserved
C00831/41	L_Compare_2: nIn1_a
C00831/42	L_Compare_2: nIn2_a
C00831/43	L_Compare_3: nIn1_a
C00831/44	L_Compare_3: nIn2_a
C00831/45	L_AnalogSwitch_2: nIn1_a
C00831/46	L_AnalogSwitch_2: nIn2_a
C00831/47	L_AnalogSwitch_3: nIn1_a
C00831/48	L_AnalogSwitch_3: nIn2_a
C00831/49	L_Arithmetik_2: nIn1_a
C00831/50	L_Arithmetik_2: nIn2_a
C00831/51	Reserved
C00831/52	Reserved
C00831/53	L_GainOffset_2: nIn_a
C00831/54	L_GainOffset_2: nGain_a
C00831/55	L_GainOffset_2: nOffset_a
C00831/56	L_OffsetGainP_1: nIn_a
C00831/57	L_OffsetGainP_2: nIn_a
C00831/58	L_OffsetGain_2: nIn_a
C00831/59	L_OffsetGain_2: nOffset_a

Parameter | Name: **C00831 | 16-bit inputs**	Data type: UNSIGNED_16 Index: 23744_d = $5CC0_h$
C00831/60	L_OffsetGain_2: nGain_a
C00831/61	L_PCTRL_1: nAct_a
C00831/62	L_PCTRL_1: nAdapt_a
C00831/63	L_PCTRL_1: nSet_a
C00831/64	L_PCTRL_1: nInfluence_a
C00831/65	MCK: nSpeedCtrlI_a
C00831/66	MCK: nPWMAngleOffset_a
C00831/67	Reserved
C00831/68	MCK: nMBrakeAddValue_a
C00831/69	MCK: nTorqueSetValue_a
C00831/70	MCK: nTorqueLimitAdapt_a
C00831/71	MCK: nSRampOverride_a
C00831/72	MCK: nSpeedSetValue_a
C00831/73	MCK: wMotionCtrl2
C00831/74	MCK: wMotionCtrl1
C00831/75	MCK: nSpeedOverride_a
C00831/76	MCK: nAccOverride_a
C00831/77	MCK: nSpeedAdd_v
C00831/78	MCK: wAuxCtrl
C00831/79	MCK: wSMCtrl
C00831/80	L_OffsetGainP_3: nIn_a
C00831/81	L_MPot_1: nIn_a
C00831/82	L_MulDiv_1: nIn_a
C00831/83	LS_DataAccess: wIn1 (Lenze-internal)
C00831/84	LS_DataAccess: wIn2 (Lenze-internal)
C00831/85	LS_DataAccess: wIn3 (Lenze-internal)
C00831/86	LS_DataAccess: wIn4 (Lenze-internal)
C00831/87	L_PT1_1: nIn_a
C00831/88	MCTRL: nSpeedHighLimit_a
C00831/89	L_PCTRL_1: nNSet_a
C00831/90	L_PCTRL_1: nISet_a
C00831/91	L_Interpolator_1: nPhdIn_v
C00831/92	L_Interpolator_1: nNIn
C00831/93	Reserved
C00831/94	Reserved
C00831/95	Reserved
C00831/96	MCTRL: nInertiaAdapt_a
C00831/97	MCTRL: nSpeedSetValueInertia_a
C00831/98	MCK: nProcessIn1_a
C00831/99	MCK: nProcessIn2_a
C00831/100	MCK: nProcessIn3_a

Parameter | Name: **C00831 | 16-bit inputs**	Data type: UNSIGNED_16 Index: 23744_d = $5CC0_h$
C00831/101	MCK: nProcessIn4_a
C00831/102	MCK: nProcessIn5_a
C00831/103	MCK: nProcessIn6_a
C00831/104	MCK: nProcessIn7_a
C00831/105	MCK: nProcessIn8_a
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT	

C00832

Parameter | Name: **C00832 | 16-bit inputs [incr./ms]**			Data type: INTEGER_16 Index: 23743_d = $5CBF_h$
Display of 16-bit input values of different blocks in [rpm]			
Display range (min. value | unit | max. value)			
-32767	Incr./ms	32767	
Subcodes			**Info**
C00832/1			L_Absolut_1: nIn_a
C00832/2			L_AddSub_1: nIn1_a
C00832/3			L_AddSub_1: nIn2_a
C00832/4			L_AddSub_1: nIn3_a
C00832/5			L_OffsetGain_1: nIn_a
C00832/6			L_OffsetGain_1: nOffset_a
C00832/7			L_OffsetGain_1: nGain_a
C00832/8			L_Negation_1: nIn_a
C00832/9			L_GainOffset_1: nIn_a
C00832/10			L_GainOffset_1: nGain_a
C00832/11			L_GainOffset_1: nOffset_a
C00832/12			L_Arithmetik_1: nIn1_a
C00832/13			L_Arithmetik_1: nIn2_a
C00832/14			L_AnalogSwitch_1: nIn1_a
C00832/15			L_AnalogSwitch_1: nIn2_a
C00832/16			L_Compare_1: nIn1_a
C00832/17			L_Compare_1: nIn2_a
C00832/18			MCTRL: nTorqueLimitAdapt_a
C00832/19			Reserved
C00832/20			MCTRL: nPosCtrlPAdapt_a
C00832/21			MCTRL: nPosCtrlOutLimit_a
C00832/22			MCTRL: nSpeedSetValue_a
C00832/23			MCTRL: nSpeedLowLimit_a
C00832/24			MCTRL: nSpeedCtrlI_a
C00832/25			MCTRL: nSpeedCtrlPAdapt_a
C00832/26			MCTRL: nBoost_a
C00832/27			MCTRL: nTorqueSetValue_a
C00832/28			MCTRL: nTorqueGenLimit_a
C00832/29			MCTRL: nTorqueMotLimit_a
C00832/30			Reserved

Parameter | Name: **C00832 | 16-bit inputs [incr./ms]**	Data type: INTEGER_16 Index: 23743_d = $5CBF_h$
C00832/31	MCTRL: nVoltageAdd_a
C00832/32	MCTRL: nPWMAngleOffset_a
C00832/33	L_NSet_1: nCInhVal_a
C00832/34	L_NSet_1: nNSet_a
C00832/35	L_NSet_1: nSet_a
C00832/36	L_NSet_1: nNAdd_a
C00832/37	DCTRL: wCANControl
C00832/38	DCTRL: wCCMControl
C00832/39	L_NLim_1: nIn_a
C00832/40	Reserved
C00832/41	L_Compare_2: nIn1_a
C00832/42	L_Compare_2: nIn2_a
C00832/43	L_Compare_3: nIn1_a
C00832/44	L_Compare_3: nIn2_a
C00832/45	L_AnalogSwitch_2: nIn1_a
C00832/46	L_AnalogSwitch_2: nIn2_a
C00832/47	L_AnalogSwitch_3: nIn1_a
C00832/48	L_AnalogSwitch_3: nIn2_a
C00832/49	L_Arithmetik_2: nIn1_a
C00832/50	L_Arithmetik_2: nIn2_a
C00832/51	Reserved
C00832/52	Reserved
C00832/53	L_GainOffset_2: nIn_a
C00832/54	L_GainOffset_2: nGain_a
C00832/55	L_GainOffset_2: nOffset_a
C00832/56	L_OffsetGainP_1: nIn_a
C00832/57	L_OffsetGainP_2: nIn_a
C00832/58	L_OffsetGain_2: nIn_a
C00832/59	L_OffsetGain_2: nOffset_a
C00832/60	L_OffsetGain_2: nGain_a
C00832/61	L_PCTRL_1: nAct_a
C00832/62	L_PCTRL_1: nAdapt_a
C00832/63	L_PCTRL_1: nSet_a
C00832/64	L_PCTRL_1: nInfluence_a
C00832/65	MCK: nSpeedCtrlI_a
C00832/66	MCK: nPWMAngleOffset_a
C00832/67	Reserved
C00832/68	MCK: nMBrakeAddValue_a
C00832/69	MCK: nTorqueSetValue_a
C00832/70	MCK: nTorqueLimitAdapt_a
C00832/71	MCK: nSRampOverride_a
C00832/72	MCK: nSpeedSetValue_a
C00832/73	MCK: wMotionCtrl2
C00832/74	MCK: wMotionCtrl1

Parameter | Name: **C00832 | 16-bit inputs [incr./ms]**	Data type: INTEGER_16 Index: 23743_d = $5CBF_h$
C00832/75	MCK: nSpeedOverride_a
C00832/76	MCK: nAccOverride_a
C00832/77	MCK: nSpeedAdd_v
C00832/78	MCK: wAuxCtrl
C00832/79	MCK: wSMCtrl
C00832/80	L_OffsetGainP_3: nIn_a
C00832/81	L_MPot_1: nIn_a
C00832/82	L_MulDiv_1: nIn_a
C00832/83	LS_DataAccess: wIn1 (Lenze-internal)
C00832/84	LS_DataAccess: wIn2 (Lenze-internal)
C00832/85	LS_DataAccess: wIn3 (Lenze-internal)
C00832/86	LS_DataAccess: wIn4 (Lenze-internal)
C00832/87	L_PT1_1: nIn_a
C00832/88	MCTRL: nSpeedHighLimit_a
C00832/89	L_PCTRL_1: nNSet_a
C00832/90	L_PCTRL_1: nISet_a
C00832/91	L_Interpolator_1: nPhdIn_v
C00832/92	L_Interpolator_1: nNIn_a
C00832/93	Reserved
C00832/94	Reserved
C00832/95	Reserved
C00832/96	MCTRL: nInertiaAdapt_a
C00832/97	MCTRL: nSpeedSetValueInertia_a
C00832/98	MCK: nProcessIn1_a
C00832/99	MCK: nProcessIn2_a
C00832/100	MCK: nProcessIn3_a
C00832/101	MCK: nProcessIn4_a
C00832/102	MCK: nProcessIn5_a
C00832/103	MCK: nProcessIn6_a
C00832/104	MCK: nProcessIn7_a
C00832/105	MCK: nProcessIn8_a
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1	

C00833

Parameter | Name: **C00833 | Binary inputs**		Data type: UNSIGNED_8 Index: 23742_d = $5CBE_h$
Display of the signal status of the binary inputs of different blocks		
Selection list		
0	False	
1	True	
Subcodes		**Info**

Parameter | Name: **C00833 | Binary inputs**	Data type: UNSIGNED_8 Index: 23742_d = $5CBE_h$
C00833/1	L_And_1: bIn1
C00833/2	L_And_1: bIn2
C00833/3	L_And_1: bIn3
C00833/4	L_DFlipFlop_1: bD
C00833/5	L_DFlipFlop_1: bClk
C00833/6	L_DFlipFlop_1: bClr
C00833/7	L_Not_1: bIn
C00833/8	L_Or_1: bIn1
C00833/9	L_Or_1: bIn2
C00833/10	L_Or_1: bIn3
C00833/11	L_RLQ_1: bCw
C00833/12	L_RLQ_1: bCcw
C00833/13	L_AnalogSwitch_1: bSet
C00833/14	L_NSet_1: bRfgStop
C00833/15	L_NSet_1: bRfg0
C00833/16	L_NSet_1: bNSetInv
C00833/17	L_NSet_1: bJog1
C00833/18	L_NSet_1: bJog2
C00833/19	L_NSet_1: bJog4
C00833/20	L_NSet_1: bJog8
C00833/21	L_NSet_1: bTi1
C00833/22	L_NSet_1: bTi2
C00833/23	L_NSet_1: bTi4
C00833/24	L_NSet_1: bTi8
C00833/25	L_NSet_1: bLoad
C00833/26	L_NSet_1: bExternalCINH
C00833/27	MCTRL: bPosCtrlOn
C00833/28	MCTRL: bSpeedInterpolatorOn
C00833/29	MCTRL: bTorqueInterpolatorOn
C00833/30	MCTRL: bTorquemodeOn
C00833/31	MCTRL: bSpeedCtrlIOn
C00833/32	MCTRL: bAutoBoostOn
C00833/33	MCTRL: bQSPOn
C00833/34	MCTRL: bDcBrakeOn
C00833/35	MCTRL: bDeltaPosOn
C00833/36	DCTRL: bCINH
C00833/37	DCTRL: bFailReset
C00833/38	DCTRL: bStatus_B0
C00833/39	DCTRL: bStatus_B2
C00833/40	DCTRL: bStatus_B3
C00833/41	DCTRL: bStatus_B4
C00833/42	DCTRL: bStatus_B5
C00833/43	DCTRL: bStatus_B14
C00833/44	DCTRL: bStatus_B15

Parameter | Name: **C00833 | Binary inputs**	Data type: UNSIGNED_8 Index: 23742_d = $5CBE_h$
C00833/45	DCTRL: bFree_1
C00833/46	DCTRL: bFree_2
C00833/47	DCTRL: bFree_3
C00833/48	DCTRL: bFree_4
C00833/49	L_And_2: bIn1
C00833/50	L_And_2: bIn2
C00833/51	L_And_2: bIn3
C00833/52	L_And_3: bIn1
C00833/53	L_And_3: bIn2
C00833/54	L_And_3: bIn3
C00833/55	L_Or_2: bIn1
C00833/56	L_Or_2: bIn2
C00833/57	L_Or_2: bIn3
C00833/58	L_Or_3: bIn1
C00833/59	L_Or_3: bIn2
C00833/60	L_Or_3: bIn3
C00833/61	L_Not_2: bIn
C00833/62	L_Not_3: bIn
C00833/63	L_DigitalLogic_1: bIn1
C00833/64	L_DigitalLogic_1: bIn2
C00833/65	L_DigitalLogic_1: bIn3
C00833/66	L_DigitalDelay_1: bIn
C00833/67	MCTRL: bPosDerivativeOn
C00833/68	MCTRL: bSetRefValue
C00833/69	MCTRL: bSpeedCtrlPAdaptOn
C00833/70	L_AnalogSwitch_2: bSet
C00833/71	L_AnalogSwitch_3: bSet
C00833/72	L_MPot_1: bUp
C00833/73	L_MPot_1: bInAct
C00833/74	L_MPot_1: bDown
C00833/75	L_PCTRL_1: bPIDOff
C00833/76	L_PCTRL_1: bInAct
C00833/77	L_PCTRL_1: bIOff
C00833/78	MCK: bSpeedCtrlIOn
C00833/79	MCK: bDcBrakeOn
C00833/80	MCK: bMBrakeRelease
C00833/81	MCK: bMBrakeStartValue2
C00833/82	MCK: bMBrakeApplied
C00833/83	MCK: bLimitSwitchPos
C00833/84	MCK: bLimitSwitchNeg
C00833/85	MCK: bPosCtrlOn
C00833/86	MCK: bDeltaPosOn
C00833/87	MCK: bPosDerivativeOn
C00833/88	MCK: bReserved01

Parameter | Name: **C00833 | Binary inputs**	Data type: UNSIGNED_8 Index: 23742_d = $5CBE_h$
C00833/89	MCK: bQspOn
C00833/90	MCK: bTorquemodeOn
C00833/91	MCK: bTorqueLimitAdaptOn
C00833/92	MCK: bHomMark
C00833/93	L_Transient_1: bIn
C00833/94	L_Transient_2: bIn
C00833/95	L_Transient_3: bIn
C00833/96	L_Transient_4: bIn
C00833/97	Reserved
C00833/98	MCTRL: bTorqueLimitAdaptOn
C00833/99	L_NSet_1: bNAddInv
C00833/100	L_MPot_1: bEnable
C00833/101	L_NLim_1: bEnable
C00833/102	LS_DataAccess: bEnableIn1 (Lenze-internal)
C00833/103	LS_DataAccess: bEnableIn2 (Lenze-internal)
C00833/104	LS_DataAccess: bEnableIn3 (Lenze-internal)
C00833/105	LS_DataAccess: bEnableIn4 (Lenze-internal)
C00833/106	L_PCTRL_1: bEnableInfluenceRamp
C00833/107	LS_SetError_2: bSetError1
C00833/108	LS_SetError_2: bSetError2
C00833/109	LS_SetError_2: bSetError3
C00833/110	LS_SetError_2: bSetError4
C00833/111	L_JogCtrlExtension_1: bInputSel1
C00833/112	L_JogCtrlExtension_1: bInputSel2
C00833/113	L_JogCtrlExtension_1: bRfgIn
C00833/114	L_JogCtrlExtension_1: bJog1In
C00833/115	L_JogCtrlExtension_1: bJog2In
C00833/116	L_JogCtrlExtension_1: bSlowDown1
C00833/117	L_JogCtrlExtension_1: bStop1
C00833/118	L_JogCtrlExtension_1: bSlowDown2
C00833/119	L_JogCtrlExtension_1: bStop2
C00833/120	L_JogCtrlExtension_1: bSlowDown3
C00833/121	L_JogCtrlExtension_1: bStop3
C00833/122	L_PCTRL_1: bISet
C00833/123	L_Interpolator_1: bSpeedAct0
C00833/124	L_Or_4: bIn1
C00833/125	L_Or_4: bIn2
C00833/126	L_Or_4: bIn3
C00833/127	L_DigitalLogic_3: bIn1

Parameter | Name: **C00833 | Binary inputs**	Data type: UNSIGNED_8 Index: 23742_d = $5CBE_h$
C00833/128	L_DigitalLogic_3: bIn2
C00833/129	L_DigitalLogic_3: bIn3
C00833/130	MCTRL: bBrakeChopperOn
C00833/131	MCTRL: bVfcEcoDisable
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1	

C00834

Parameter | Name: **C00834 | 32-bit inputs [incr]**			Data type: INTEGER_32 Index: 23741_d = $5CBD_h$
Display in [increments] of 32 bit input values of various blocks			
Display range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes			**Info**
C00834/1			MCK: dnPosSetValue_p
C00834/2			MCK: dnPosRefValue_p
C00834/3			MCK: dnDeltaPos_p
C00834/4			MCTRL: dnDeltaPos_p
C00834/5			MCTRL: dnPosSetValue_p
C00834/6			MCTRL: dnPosRefValue_p
C00834/7			MCK: dnProfilePosition_p
C00834/8			L_Interpolator_1: dnPhiIn_p
C00834/9			MCK: dnProcessIn1_p
C00834/10			MCK: dnProcessIn2_p
C00834/11			MCK: dnProcessIn3_p
C00834/12			MCK: dnProcessIn4_p
C00834/13			L_Interpolator_1: dnPosSetCycle_p
C00834/14			MCK: dnPosSetCycle_p
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00835

Parameter | Name: **C00835 | 16-bit inputs [%] (Set2)**			Data type: INTEGER_16 Index: 23740_d = $5CBC_h$
Display in percent of 16-bit input values of different blocks			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00835/1			L_Absolut_2: nIn_a
C00835/2			L_AnalogSwitch_4: nIn1_a
C00835/3			L_AnalogSwitch_4: nIn2_a
C00835/4			L_AnalogSwitch_5: nIn1_a
C00835/5			L_AnalogSwitch_5: nIn2_a
C00835/6			L_Compare_4: nIn1_a
C00835/7			L_Compare_4: nIn2_a
C00835/8			L_Compare_5: nIn1_a

Parameter | Name: **C00835 | 16-bit inputs [%] (Set2)**	Data type: INTEGER_16 Index: 23740_d = $5CBC_h$
C00835/9	L_Compare_5: nIn2_a
C00835/10	L_Arithmetik_3: nIn1_a
C00835/11	L_Arithmetik_3: nIn2_a
C00835/12	L_Arithmetik_4: nIn1_a
C00835/13	L_Arithmetik_4: nIn2_a
C00835/14	L_Arithmetik_5: nIn1_a
C00835/15	L_Arithmetik_5: nIn2_a
C00835/16	L_Counter_2: wLdVal
C00835/17	L_Counter_2: wCmpVal
C00835/18	L_Counter_3: wLdVal
C00835/19	L_Counter_3: wCmpVal
C00835/20	L_PhaseIntK_1: nIn_v
C00835/21	L_Negation_2: nIn_a
C00835/22	L_NLim_2: nIn_a
C00835/23	L_OffsetGain_3: nIn_a
C00835/24	L_OffsetGain_3: nOffset_a
C00835/25	L_OffsetGain_3: nGain_a
C00835/26	L_PT1_2: nIn_a
C00835/27	L_PT1_3: nIn_a
C00835/28	L_PhaseIntK_2: nIn_v
C00835/29	L_SampleHold_1: nIn_a
C00835/30	L_SampleHold_2: nIn_a
C00835/31	L_Mux_1: wInSelect
C00835/32	L_GainOffset_3: nIn_a
C00835/33	L_GainOffset_3: nGain_a
C00835/34	L_GainOffset_3: nOffset_a
C00835/35	L_MulDiv_2: nIn_a
C00835/36	L_DT1_1: nIn_a
C00835/37	L_Counter_1: wLdVal
C00835/38	L_Counter_1: wCmpVal
C00835/39	L_GainOffsetP_1: nIn_a
C00835/40	L_GainOffsetP_2: nIn_a
C00835/41	L_GainOffsetP_3: nIn_a
C00835/42	L_Limit_1: nIn_a
C00835/43	L_Limit_2: nIn_a
C00835/44	L_MckCtrlInterface_1: wOperationMode
C00835/45	L_MckCtrlInterface_1: wPosMode
C00835/46	L_MckCtrlInterface_1: wProfileNo
C00835/47	L_MckCtrlInterface_1: wInMckPosCtrl_1
C00835/48	L_MckCtrlInterface_1: wInMckPosCtrl_2
C00835/49	L_MckStateInterface_1: wInMckPosState_1
C00835/50	L_MckStateInterface_1: wInMckPosState_2
C00835/51	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_1
C00835/52	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_2

Parameter | Name: **C00835 | 16-bit inputs [%] (Set2)**	Data type: INTEGER_16 Index: 23740_d = $5CBC_h$
C00835/53	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_3
C00835/54	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_4
C00835/55	L_ConvWordToBits_1: wInput
C00835/56	L_ConvWordToBits_2: wInput
C00835/57	L_ConvWordToBits_3: wInput
C00835/58	L_ConvWordsToDInt_1: wInLWord
C00835/59	L_ConvWordsToDInt_1: wInHWord
C00835/60	L_ConvWordsToDInt_2: wInLWord
C00835/61	L_ConvWordsToDInt_2: wInHWord
C00835/62	L_ConvWordsToDInt_3: wInLWord
C00835/63	L_ConvWordsToDInt_3: wInHWord
C00835/64	L_ConvUnitsToIncr_1: wInLWord
C00835/65	L_ConvUnitsToIncr_1: wInHWord
C00835/66	L_ConvUnitsToIncr_2: wInLWord
C00835/67	L_ConvUnitsToIncr_2: wInHWord
C00835/68	L_ConvUnitsToIncr_3: wInLWord
C00835/69	L_ConvUnitsToIncr_3: wInHWord
C00835/70	L_Curve_1: nIn_a
C00835/71	L_ConvW_1: wIn
C00835/72	L_ConvW_2: wIn
C00835/73	L_ConvW_3: wIn
C00835/74	L_ConvW_4: wIn
C00835/75	L_MckCtrlInterface_1: wPosSetHW
C00835/76	L_MckCtrlInterface_1: wPosSetLW
C00835/77	L_PhaseDiff_1: nIn_v
C00835/78	L_PhaseDiff_2: nIn_v
C00835/79	L_SRFG_1: nIn_a
C00835/80	L_SRFG_2: nIn_a
C00835/81	L_SRFG_1: nSet_a
C00835/82	L_SRFG_2: nSet_a
C00835/83	L_SignalSwitch_1: wIn1
C00835/84	L_SignalSwitch_2: wIn1
C00835/85	L_SignalSwitch_3: wIn1
C00835/86	L_SignalSwitch_4: wIn1
C00835/87	L_SignalSwitch_1: wIn2
C00835/88	L_SignalSwitch_2: wIn2
C00835/89	L_SignalSwitch_3: wIn2
C00835/90	L_SignalSwitch_4: wIn2
C00835/91	L_Odometer_1: nInSpeed_v
C00835/92	L_CalcDiameter_1: wDMax
C00835/93	L_CalcDiameter_1: wDMin
C00835/94	L_CalcDiameter_1: wVMax
C00835/95	L_CalcDiameter_1: nVLine_a
C00835/96	L_CalcDiameter_1: nMotorSpeedAct_v

Parameter | Name: **C00835 | 16-bit inputs [%] (Set2)**	Data type: INTEGER_16 Index: 23740_d = $5CBC_h$
C00835/97	L_CalcDiameter_1: wGearNum
C00835/98	L_CalcDiameter_1: wGearDenom
C00835/99	L_CalcDiameter_1: nSetD_a
C00835/100	L_ProcessCtrl_1: nVpAdapt_a
C00835/101	L_ProcessCtrl_1: nSet_a
C00835/102	L_ProcessCtrl_1: nAct_a
C00835/103	L_ProcessCtrl_1: nRTimeAdapt_a
C00835/104	L_ProcessCtrl_1: nInfluence_a
C00835/105	L_PhiIntegrator_1: nIn_v
C00835/106	L_PhiIntegrator_1: wGearNum
C00835/107	L_PhiIntegrator_1: wGearDenom
C00835/108	L_SwitchPoint_1: nActSpeed_v
C00835/109	L_PhiIntegrator_1: nSpeedAdd_v
C00835/110	L_DFSET_1: nSpeedTrim_v
C00835/111	L_DFSET_1: nSpeedTrim_a
C00835/112	L_DFSET_1: wGearNum
C00835/113	L_DFSET_1: wGainNum
C00835/114	L_DFSET_1: nSet_v
C00835/115	L_DFSET_1: wGainDenom
C00835/116	L_DFSET_1: wGearDenom
C00835/117	L_DFSET_1: nPositionTrimming
C00835/118	L_DFSET_1: nNAct_v
C00835/119	L_DFRFG_1: nIn_v
C00835/120	L_GearComp_1: nTorque_a
C00835/121	L_ConvAP_1: nIn_a
C00835/122	L_ConvAP_2: nIn_a
C00835/123	L_ConvAP_3: nIn_a
C00835/124	L_ConvX_1: nIn_a
C00835/125	L_ConvX_1: nNum
C00835/126	L_ConvX_1: wDenom
C00835/127	L_ConvX_2: nIn_a
C00835/128	L_ConvX_2: nNum
C00835/129	L_ConvX_2: wDenom
C00835/130	L_ConvX_3: nIn_a
C00835/131	L_ConvX_3: nNum
C00835/132	L_ConvX_3: wDenom
C00835/133	L_ConvPP_1: nNum
C00835/134	L_ConvPP_1: wDenom
C00835/135	L_ConvPP_2: nNum
C00835/136	L_ConvPP_2: wDenom
C00835/137	L_ConvPP_3: nNum
C00835/138	L_ConvPP_3: wDenom
C00835/139	L_Curve_2: nIn_a
C00835/140	L_Curve_3: nIn_a

Parameter | Name: **C00835 | 16-bit inputs [%] (Set2)**	Data type: INTEGER_16 Index: 23740_d = $5CBC_h$
C00835/141	L_Sequencer_1: wStartStep
C00835/142	L_Sequencer_1: wBranch1
C00835/143	L_Sequencer_1: wBranch2
C00835/144	L_Sequencer_1: wMotionState1
C00835/145	L_Sequencer_1: wMotionState2
C00835/146	L_Sequencer_1: wDigitalInputs
C00835/147	L_ConvActPos_1: nSetPos_a
C00835/148	L_ConvActPos_1: nPosIn_a
C00835/149	L_ConvActPos_1: wVMax
C00835/150	L_ConvActPos_1: nVLine_a
C00835/151	L_MFail_1: nAdapt_a
C00835/152	L_MFail_1: nConst_a
C00835/153	L_MFail_1: nDCSet_a
C00835/154	L_MFail_1: nNSet_a
C00835/155	L_MFail_1: nThreshold_a
C00835/156	L_MFail_1: nNAct_a
C00835/157	L_MFail_1: nSet_a
C00835/158	L_MFail_1: nDCVoltAct_a
C00835/159	L_Curve_3: nCurveCtrl_a
C00835/160	L_Curve_3: nD0_a
C00835/161	L_Curve_3: nDiameter_a
C00835/162	L_SwitchPointPar_1: nActSpeed_v
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100	

C00836

Parameter | Name: **C00836 | 16-bit inputs (Set2)**			Data type: UNSIGNED_16 Index: 23739_d = $5CBB_h$
Decimal/hexadecimal/bit coded display of 16 bit input values of various blocks			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
...	...		
Bit 15	Active		
Subcodes			**Info**
C00836/1			L_Absolut_2: nIn_a
C00836/2			L_AnalogSwitch_4: nIn1_a
C00836/3			L_AnalogSwitch_4: nIn2_a
C00836/4			L_AnalogSwitch_5: nIn1_a
C00836/5			L_AnalogSwitch_5: nIn2_a
C00836/6			L_Compare_4: nIn1_a
C00836/7			L_Compare_4: nIn2_a
C00836/8			L_Compare_5: nIn1_a
C00836/9			L_Compare_5: nIn2_a

Parameter | Name: **C00836 | 16-bit inputs (Set2)**	Data type: UNSIGNED_16 Index: 23739_d = $5CBB_h$
C00836/10	L_Arithmetik_3: nIn1_a
C00836/11	L_Arithmetik_3: nIn2_a
C00836/12	L_Arithmetik_4: nIn1_a
C00836/13	L_Arithmetik_4: nIn2_a
C00836/14	L_Arithmetik_5: nIn1_a
C00836/15	L_Arithmetik_5: nIn2_a
C00836/16	L_Counter_2: wLdVal
C00836/17	L_Counter_2: wCmpVal
C00836/18	L_Counter_3: wLdVal
C00836/19	L_Counter_3: wCmpVal
C00836/20	L_PhaseIntK_1: nIn_v
C00836/21	L_Negation_2: nIn_a
C00836/22	L_NLim_2: nIn_a
C00836/23	L_OffsetGain_3: nIn_a
C00836/24	L_OffsetGain_3: nOffset_a
C00836/25	L_OffsetGain_3: nGain_a
C00836/26	L_PT1_2: nIn_a
C00836/27	L_PT1_3: nIn_a
C00836/28	L_PhaseIntK_2: nIn_v
C00836/29	L_SampleHold_1: nIn_a
C00836/30	L_SampleHold_2: nIn_a
C00836/31	L_Mux_1: wInSelect
C00836/32	L_GainOffset_3: nIn_a
C00836/33	L_GainOffset_3: nGain_a
C00836/34	L_GainOffset_3: nOffset_a
C00836/35	L_MulDiv_2: nIn_a
C00836/36	L_DT1_1: nIn_a
C00836/37	L_Counter_1: wLdVal
C00836/38	L_Counter_1: wCmpVal
C00836/39	L_GainOffsetP_1: nIn_a
C00836/40	L_GainOffsetP_2: nIn_a
C00836/41	L_GainOffsetP_3: nIn_a
C00836/42	L_Limit_1: nIn_a
C00836/43	L_Limit_2: nIn_a
C00836/44	L_MckCtrlInterface_1: wOperationMode
C00836/45	L_MckCtrlInterface_1: wPosMode
C00836/46	L_MckCtrlInterface_1: wProfileNo
C00836/47	L_MckCtrlInterface_1: wInMckPosCtrl_1
C00836/48	L_MckCtrlInterface_1: wInMckPosCtrl_2
C00836/49	L_MckStateInterface_1: wInMckPosState_1
C00836/50	L_MckStateInterface_1: wInMckPosState_2
C00836/51	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_1
C00836/52	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_2
C00836/53	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_3

Parameter | Name: **C00836 | 16-bit inputs (Set2)**	Data type: UNSIGNED_16 Index: 23739_d = $5CBB_h$
C00836/54	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_4
C00836/55	L_ConvWordToBits_1: wInput
C00836/56	L_ConvWordToBits_2: wInput
C00836/57	L_ConvWordToBits_3: wInput
C00836/58	L_ConvWordsToDInt_1: wInLWord
C00836/59	L_ConvWordsToDInt_1: wInHWord
C00836/60	L_ConvWordsToDInt_2: wInLWord
C00836/61	L_ConvWordsToDInt_2: wInHWord
C00836/62	L_ConvWordsToDInt_3: wInLWord
C00836/63	L_ConvWordsToDInt_3: wInHWord
C00836/64	L_ConvUnitsToIncr_1: wInLWord
C00836/65	L_ConvUnitsToIncr_1: wInHWord
C00836/66	L_ConvUnitsToIncr_2: wInLWord
C00836/67	L_ConvUnitsToIncr_2: wInHWord
C00836/68	L_ConvUnitsToIncr_3: wInLWord
C00836/69	L_ConvUnitsToIncr_3: wInHWord
C00836/70	L_Curve_1: nIn_a
C00836/71	L_ConvW_1: wIn
C00836/72	L_ConvW_2: wIn
C00836/73	L_ConvW_3: wIn
C00836/74	L_ConvW_4: wIn
C00836/75	L_MckCtrlInterface_1: wPosSetHW
C00836/76	L_MckCtrlInterface_1: wPosSetLW
C00836/77	L_PhaseDiff_1: nIn_v
C00836/78	L_PhaseDiff_2: nIn_v
C00836/79	L_SRFG_1: nIn_a
C00836/80	L_SRFG_2: nIn_a
C00836/81	L_SRFG_1: nSet_a
C00836/82	L_SRFG_2: nSet_a
C00836/83	L_SignalSwitch_1: wIn1
C00836/84	L_SignalSwitch_2: wIn1
C00836/85	L_SignalSwitch_3: wIn1
C00836/86	L_SignalSwitch_4: wIn1
C00836/87	L_SignalSwitch_1: wIn2
C00836/88	L_SignalSwitch_2: wIn2
C00836/89	L_SignalSwitch_3: wIn2
C00836/90	L_SignalSwitch_4: wIn2
C00836/91	L_Odometer_1: nInSpeed_v
C00836/92	L_CalcDiameter_1: wDMax
C00836/93	L_CalcDiameter_1: wDMin
C00836/94	L_CalcDiameter_1: wVMax
C00836/95	L_CalcDiameter_1: nVLine_a
C00836/96	L_CalcDiameter_1: nMotorSpeedAct_v
C00836/97	L_CalcDiameter_1: wGearNum

Parameter | Name: **C00836 | 16-bit inputs (Set2)**	Data type: UNSIGNED_16 Index: 23739_d = $5CBB_h$
C00836/98	L_CalcDiameter_1: wGearDenom
C00836/99	L_CalcDiameter_1: nSetD_a
C00836/100	L_ProcessCtrl_1: nVpAdapt_a
C00836/101	L_ProcessCtrl_1: nSet_a
C00836/102	L_ProcessCtrl_1: nAct_a
C00836/103	L_ProcessCtrl_1: nRTimeAdapt_a
C00836/104	L_ProcessCtrl_1: nInfluence_a
C00836/105	L_PhiIntegrator_1: nIn_v
C00836/106	L_PhiIntegrator_1: wGearNum
C00836/107	L_PhiIntegrator_1: wGearDenom
C00836/108	L_SwitchPoint_1: nActSpeed_v
C00836/109	L_PhiIntegrator_1: nSpeedAdd_v
C00836/110	L_DFSET_1: nSpeedTrim_v
C00836/111	L_DFSET_1: nSpeedTrim_a
C00836/112	L_DFSET_1: wGearNum
C00836/113	L_DFSET_1: wGainNum
C00836/114	L_DFSET_1: nSet_v
C00836/115	L_DFSET_1: wGainDenom
C00836/116	L_DFSET_1: wGearDenom
C00836/117	L_DFSET_1: nPositionTrimming
C00836/118	L_DFSET_1: nNAct_v
C00836/119	L_DFRFG_1: nIn_v
C00836/120	L_GearComp_1: nTorque_a
C00836/121	L_ConvAP_1: nIn_a
C00836/122	L_ConvAP_2: nIn_a
C00836/123	L_ConvAP_3: nIn_a
C00836/124	L_ConvX_1: nIn_a
C00836/125	L_ConvX_1: nNum
C00836/126	L_ConvX_1: wDenom
C00836/127	L_ConvX_2: nIn_a
C00836/128	L_ConvX_2: nNum
C00836/129	L_ConvX_2: wDenom
C00836/130	L_ConvX_3: nIn_a
C00836/131	L_ConvX_3: nNum
C00836/132	L_ConvX_3: wDenom
C00836/133	L_ConvPP_1: nNum
C00836/134	L_ConvPP_1: wDenom
C00836/135	L_ConvPP_2: nNum
C00836/136	L_ConvPP_2: wDenom
C00836/137	L_ConvPP_3: nNum
C00836/138	L_ConvPP_3: wDenom
C00836/139	L_Curve_2: nIn_a
C00836/140	L_Curve_3: nIn_a
C00836/141	L_Sequencer_1: wStartStep

Parameter | Name: **C00836 | 16-bit inputs (Set2)**	Data type: UNSIGNED_16 Index: 23739_d = $5CBB_h$
C00836/142	L_Sequencer_1: wBranch1
C00836/143	L_Sequencer_1: wBranch2
C00836/144	L_Sequencer_1: wMotionState1
C00836/145	L_Sequencer_1: wMotionState2
C00836/146	L_Sequencer_1: wDigitalInputs
C00836/147	L_ConvActPos_1: nSetPos_a
C00836/148	L_ConvActPos_1: nPosIn_a
C00836/149	L_ConvActPos_1: wVMax
C00836/150	L_ConvActPos_1: nVLine_a
C00836/151	L_MFail_1: nAdapt_a
C00836/152	L_MFail_1: nConst_a
C00836/153	L_MFail_1: nDCSet_a
C00836/154	L_MFail_1: nNSet_a
C00836/155	L_MFail_1: nThreshold_a
C00836/156	L_MFail_1: nNAct_a
C00836/157	L_MFail_1: nSet_a
C00836/158	L_MFail_1: nDCVoltAct_a
C00836/159	L_Curve_3: nCurveCtrl_a
C00836/160	L_Curve_3: nD0_a
C00836/161	L_Curve_3: nDiameter_a
C00836/162	L_SwitchPointPar_1: nActSpeed_v
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT	

C00837

Parameter | Name: **C00837 | 16-bit inputs [incr./ms] (Set2)**			Data type: INTEGER_16 Index: 23738_d = $5CBA_h$
Display of 16-bit input values of different blocks in [rpm]			
Display range (min. value | unit | max. value)			
-32767	Incr./ms	32767	
Subcodes			**Info**
C00837/1			L_Absolut_2: nIn_a
C00837/2			L_AnalogSwitch_4: nIn1_a
C00837/3			L_AnalogSwitch_4: nIn2_a
C00837/4			L_AnalogSwitch_5: nIn1_a
C00837/5			L_AnalogSwitch_5: nIn2_a
C00837/6			L_Compare_4: nIn1_a
C00837/7			L_Compare_4: nIn2_a
C00837/8			L_Compare_5: nIn1_a
C00837/9			L_Compare_5: nIn2_a
C00837/10			L_Arithmetik_3: nIn1_a
C00837/11			L_Arithmetik_3: nIn2_a
C00837/12			L_Arithmetik_4: nIn1_a
C00837/13			L_Arithmetik_4: nIn2_a
C00837/14			L_Arithmetik_5: nIn1_a

Parameter | Name: **C00837 | 16-bit inputs [incr./ms] (Set2)**	Data type: INTEGER_16 Index: 23738_d = $5CBA_h$
C00837/15	L_Arithmetik_5: nIn2_a
C00837/16	L_Counter_2: wLdVal
C00837/17	L_Counter_2: wCmpVal
C00837/18	L_Counter_3: wLdVal
C00837/19	L_Counter_3: wCmpVal
C00837/20	L_PhaseIntK_1: nIn_v
C00837/21	L_Negation_2: nIn_a
C00837/22	L_NLim_2: nIn_a
C00837/23	L_OffsetGain_3: nIn_a
C00837/24	L_OffsetGain_3: nOffset_a
C00837/25	L_OffsetGain_3: nGain_a
C00837/26	L_PT1_2: nIn_a
C00837/27	L_PT1_3: nIn_a
C00837/28	L_PhaseIntK_2: nIn_v
C00837/29	L_SampleHold_1: nIn_a
C00837/30	L_SampleHold_2: nIn_a
C00837/31	L_Mux_1: wInSelect
C00837/32	L_GainOffset_3: nIn_a
C00837/33	L_GainOffset_3: nGain_a
C00837/34	L_GainOffset_3: nOffset_a
C00837/35	L_MulDiv_2: nIn_a
C00837/36	L_DT1_1: nIn_a
C00837/37	L_Counter_1: wLdVal
C00837/38	L_Counter_1: wCmpVal
C00837/39	L_GainOffsetP_1: nIn_a
C00837/40	L_GainOffsetP_2: nIn_a
C00837/41	L_GainOffsetP_3: nIn_a
C00837/42	L_Limit_1: nIn_a
C00837/43	L_Limit_2: nIn_a
C00837/44	L_MckCtrlInterface_1: wOperationMode
C00837/45	L_MckCtrlInterface_1: wPosMode
C00837/46	L_MckCtrlInterface_1: wProfileNo
C00837/47	L_MckCtrlInterface_1: wInMckPosCtrl_1
C00837/48	L_MckCtrlInterface_1: wInMckPosCtrl_2
C00837/49	L_MckStateInterface_1: wInMckPosState_1
C00837/50	L_MckStateInterface_1: wInMckPosState_2
C00837/51	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_1
C00837/52	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_2
C00837/53	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_3
C00837/54	L_PosiShaftCtrlInterface_1: wInPosiShaftCtrl_4
C00837/55	L_ConvWordToBits_1: wInput
C00837/56	L_ConvWordToBits_2: wInput
C00837/57	L_ConvWordToBits_3: wInput
C00837/58	L_ConvWordsToDInt_1: wInLWord

Parameter | Name: **C00837 | 16-bit inputs [incr./ms] (Set2)**	Data type: INTEGER_16 Index: 23738_d = $5CBA_h$
C00837/59	L_ConvWordsToDInt_1: wInHWord
C00837/60	L_ConvWordsToDInt_2: wInLWord
C00837/61	L_ConvWordsToDInt_2: wInHWord
C00837/62	L_ConvWordsToDInt_3: wInLWord
C00837/63	L_ConvWordsToDInt_3: wInHWord
C00837/64	L_ConvUnitsToIncr_1: wInLWord
C00837/65	L_ConvUnitsToIncr_1: wInHWord
C00837/66	L_ConvUnitsToIncr_2: wInLWord
C00837/67	L_ConvUnitsToIncr_2: wInHWord
C00837/68	L_ConvUnitsToIncr_3: wInLWord
C00837/69	L_ConvUnitsToIncr_3: wInHWord
C00837/70	L_Curve_1: nIn_a
C00837/71	L_ConvW_1: wIn
C00837/72	L_ConvW_2: wIn
C00837/73	L_ConvW_3: wIn
C00837/74	L_ConvW_4: wIn
C00837/75	L_MckCtrlInterface_1: wPosSetHW
C00837/76	L_MckCtrlInterface_1: wPosSetLW
C00837/77	L_PhaseDiff_1: nIn_v
C00837/78	L_PhaseDiff_2: nIn_v
C00837/79	L_SRFG_1: nIn_a
C00837/80	L_SRFG_2: nIn_a
C00837/81	L_SRFG_1: nSet_a
C00837/82	L_SRFG_2: nSet_a
C00837/83	L_SignalSwitch_1: wIn1
C00837/84	L_SignalSwitch_2: wIn1
C00837/85	L_SignalSwitch_3: wIn1
C00837/86	L_SignalSwitch_4: wIn1
C00837/87	L_SignalSwitch_1: wIn2
C00837/88	L_SignalSwitch_2: wIn2
C00837/89	L_SignalSwitch_3: wIn2
C00837/90	L_SignalSwitch_4: wIn2
C00837/91	L_Odometer_1: nInSpeed_v
C00837/92	L_CalcDiameter_1: wDMax
C00837/93	L_CalcDiameter_1: wDMin
C00837/94	L_CalcDiameter_1: wVMax
C00837/95	L_CalcDiameter_1: nVLine_a
C00837/96	L_CalcDiameter_1: nMotorSpeedAct_v
C00837/97	L_CalcDiameter_1: wGearNum
C00837/98	L_CalcDiameter_1: wGearDenom
C00837/99	L_CalcDiameter_1: nSetD_a
C00837/100	L_ProcessCtrl_1: nVpAdapt_a
C00837/101	L_ProcessCtrl_1: nSet_a
C00837/102	L_ProcessCtrl_1: nAct_a

Parameter | Name: **C00837 | 16-bit inputs [incr./ms] (Set2)**	Data type: INTEGER_16 Index: 23738_d = $5CBA_h$
C00837/103	L_ProcessCtrl_1: nRTimeAdapt_a
C00837/104	L_ProcessCtrl_1: nInfluence_a
C00837/105	L_PhiIntegrator_1: nIn_v
C00837/106	L_PhiIntegrator_1: wGearNum
C00837/107	L_PhiIntegrator_1: wGearDenom
C00837/108	L_SwitchPoint_1: nActSpeed_v
C00837/109	L_PhiIntegrator_1: nSpeedAdd_v
C00837/110	L_DFSET_1: nSpeedTrim_v
C00837/111	L_DFSET_1: nSpeedTrim_a
C00837/112	L_DFSET_1: wGearNum
C00837/113	L_DFSET_1: wGainNum
C00837/114	L_DFSET_1: nSet_v
C00837/115	L_DFSET_1: wGainDenom
C00837/116	L_DFSET_1: wGearDenom
C00837/117	L_DFSET_1: nPositionTrimming
C00837/118	L_DFSET_1: nNAct_v
C00837/119	L_DFRFG_1: nIn_v
C00837/120	L_GearComp_1: nTorque_a
C00837/121	L_ConvAP_1: nIn_a
C00837/122	L_ConvAP_2: nIn_a
C00837/123	L_ConvAP_3: nIn_a
C00837/124	L_ConvX_1: nIn_a
C00837/125	L_ConvX_1: nNum
C00837/126	L_ConvX_1: wDenom
C00837/127	L_ConvX_2: nIn_a
C00837/128	L_ConvX_2: nNum
C00837/129	L_ConvX_2: wDenom
C00837/130	L_ConvX_3: nIn_a
C00837/131	L_ConvX_3: nNum
C00837/132	L_ConvX_3: wDenom
C00837/133	L_ConvPP_1: nNum
C00837/134	L_ConvPP_1: wDenom
C00837/135	L_ConvPP_2: nNum
C00837/136	L_ConvPP_2: wDenom
C00837/137	L_ConvPP_3: nNum
C00837/138	L_ConvPP_3: wDenom
C00837/139	L_Curve_2: nIn_a
C00837/140	L_Curve_3: nIn_a
C00837/141	L_Sequencer_1: wStartStep
C00837/142	L_Sequencer_1: wBranch1
C00837/143	L_Sequencer_1: wBranch2
C00837/144	L_Sequencer_1: wMotionState1
C00837/145	L_Sequencer_1: wMotionState2
C00837/146	L_Sequencer_1: wDigitalInputs

Parameter | Name: **C00837 | 16-bit inputs [incr./ms] (Set2)**	Data type: INTEGER_16 Index: 23738_d = $5CBA_h$
C00837/147	L_ConvActPos_1: nSetPos_a
C00837/148	L_ConvActPos_1: nPosIn_a
C00837/149	L_ConvActPos_1: wVMax
C00837/150	L_ConvActPos_1: nVLine_a
C00837/151	L_MFail_1: nAdapt_a
C00837/152	L_MFail_1: nConst_a
C00837/153	L_MFail_1: nDCSet_a
C00837/154	L_MFail_1: nNSet_a
C00837/155	L_MFail_1: nThreshold_a
C00837/156	L_MFail_1: nNAct_a
C00837/157	L_MFail_1: nSet_a
C00837/158	L_MFail_1: nDCVoltAct_a
C00837/159	L_Curve_3: nCurveCtrl_a
C00837/160	L_Curve_3: nD0_a
C00837/161	L_Curve_3: nDiameter_a
C00837/162	L_SwitchPointPar_1: nActSpeed_v
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1	

C00838

Parameter | Name: **C00838 | Binary inputs (Set2)**		Data type: UNSIGNED_8 Index: 23737_d = $5CB9_h$
Display of the signal status of the binary inputs of different blocks • In addition to the parameter C00833.		
Selection list		
0	False	
1	True	
Subcodes		**Info**
C00838/1		L_And5_1: bIn1
C00838/2		L_And5_1: bIn2
C00838/3		L_And5_1: bIn3
C00838/4		L_And5_1: bIn4
C00838/5		L_And5_1: bIn5
C00838/6		L_And5_2: bIn1
C00838/7		L_And5_2: bIn2
C00838/8		L_And5_2: bIn3
C00838/9		L_And5_2: bIn4
C00838/10		L_And5_2: bIn5
C00838/11		L_AnalogSwitch_4: bSet
C00838/12		L_AnalogSwitch_5: bSet
C00838/13		L_DFlipFlop_2: bD
C00838/14		L_DFlipFlop_2: bClk
C00838/15		L_DFlipFlop_2: bClr
C00838/16		L_DigitalDelay_2: bIn
C00838/17		L_DigitalDelay_3: bIn

Parameter | Name: **C00838 | Binary inputs (Set2)**	Data type: UNSIGNED_8 Index: 23737_d = $5CB9_h$
C00838/18	L_PhaseIntK_1: bLoad
C00838/19	L_PhaseIntK_2: bLoad
C00838/20	L_DigitalLogic_2: bIn1
C00838/21	L_DigitalLogic_2: bIn2
C00838/22	L_DigitalLogic_2: bIn3
C00838/23	L_DigitalLogic5_1: bIn1
C00838/24	L_DigitalLogic5_1: bIn2
C00838/25	L_DigitalLogic5_1: bIn3
C00838/26	L_DigitalLogic5_1: bIn4
C00838/27	L_DigitalLogic5_1: bIn5
C00838/28	L_DigitalLogic5_2: bIn1
C00838/29	L_DigitalLogic5_2: bIn2
C00838/30	L_DigitalLogic5_2: bIn3
C00838/31	L_DigitalLogic5_2: bIn4
C00838/32	L_DigitalLogic5_2: bIn5
C00838/33	L_NLim_2: bEnable
C00838/34	L_Or5_1: bIn1
C00838/35	L_Or5_1: bIn2
C00838/36	L_Or5_1: bIn3
C00838/37	L_Or5_1: bIn4
C00838/38	L_Or5_1: bIn5
C00838/39	L_Or5_2: bIn1
C00838/40	L_Or5_2: bIn2
C00838/41	L_Or5_2: bIn3
C00838/42	L_Or5_2: bIn4
C00838/43	L_Or5_2: bIn5
C00838/44	L_Not_4: bIn
C00838/45	L_Not_5: bIn
C00838/46	L_Not_6: bIn
C00838/47	L_Not_7: bIn
C00838/48	L_RSFlipFlop_1: bSet
C00838/49	L_RSFlipFlop_1: bReset
C00838/50	L_RSFlipFlop_2: bSet
C00838/51	L_RSFlipFlop_2: bReset
C00838/52	L_SampleHold_1: bLoad
C00838/53	L_SampleHold_2: bLoad
C00838/54	L_Counter_2: bClkUp
C00838/55	L_Counter_2: bClkDown
C00838/56	L_Counter_2: bLoad
C00838/57	L_Counter_3: bClkUp
C00838/58	L_Counter_3: bClkDown
C00838/59	L_Counter_3: bLoad
C00838/60	L_Transient_5: bIn
C00838/61	L_Transient_6: bIn

Parameter | Name: **C00838 | Binary inputs (Set2)**	Data type: UNSIGNED_8 Index: 23737_d = $5CB9_h$
C00838/62	L_Transient_7: bIn
C00838/63	L_Transient_8: bIn
C00838/64	L_Counter_1: bCountUp
C00838/65	L_Counter_1: bCountDown
C00838/66	L_Counter_1: bLoad
C00838/67	L_ConvBitsToWord_1: bBit0
C00838/68	L_ConvBitsToWord_1: bBit1
C00838/69	L_ConvBitsToWord_1: bBit2
C00838/70	L_ConvBitsToWord_1: bBit3
C00838/71	L_ConvBitsToWord_1: bBit4
C00838/72	L_ConvBitsToWord_1: bBit5
C00838/73	L_ConvBitsToWord_1: bBit6
C00838/74	L_ConvBitsToWord_1: bBit7
C00838/75	L_ConvBitsToWord_1: bBit8
C00838/76	L_ConvBitsToWord_1: bBit9
C00838/77	L_ConvBitsToWord_1: bBit10
C00838/78	L_ConvBitsToWord_1: bBit11
C00838/79	L_ConvBitsToWord_1: bBit12
C00838/80	L_ConvBitsToWord_1: bBit13
C00838/81	L_ConvBitsToWord_1: bBit14
C00838/82	L_ConvBitsToWord_1: bBit15
C00838/83	L_ConvBitsToWord_2: bBit0
C00838/84	L_ConvBitsToWord_2: bBit1
C00838/85	L_ConvBitsToWord_2: bBit2
C00838/86	L_ConvBitsToWord_2: bBit3
C00838/87	L_ConvBitsToWord_2: bBit4
C00838/88	L_ConvBitsToWord_2: bBit5
C00838/89	L_ConvBitsToWord_2: bBit6
C00838/90	L_ConvBitsToWord_2: bBit7
C00838/91	L_ConvBitsToWord_2: bBit8
C00838/92	L_ConvBitsToWord_2: bBit9
C00838/93	L_ConvBitsToWord_2: bBit10
C00838/94	L_ConvBitsToWord_2: bBit11
C00838/95	L_ConvBitsToWord_2: bBit12
C00838/96	L_ConvBitsToWord_2: bBit13
C00838/97	L_ConvBitsToWord_2: bBit14
C00838/98	L_ConvBitsToWord_2: bBit15
C00838/99	L_ConvBitsToWord_3: bBit0
C00838/100	L_ConvBitsToWord_3: bBit1
C00838/101	L_ConvBitsToWord_3: bBit2
C00838/102	L_ConvBitsToWord_3: bBit3
C00838/103	L_ConvBitsToWord_3: bBit4
C00838/104	L_ConvBitsToWord_3: bBit5
C00838/105	L_ConvBitsToWord_3: bBit6

Parameter | Name: **C00838 | Binary inputs (Set2)**	Data type: UNSIGNED_8 Index: 23737_d = $5CB9_h$
C00838/106	L_ConvBitsToWord_3: bBit7
C00838/107	L_ConvBitsToWord_3: bBit8
C00838/108	L_ConvBitsToWord_3: bBit9
C00838/109	L_ConvBitsToWord_3: bBit10
C00838/110	L_ConvBitsToWord_3: bBit11
C00838/111	L_ConvBitsToWord_3: bBit12
C00838/112	L_ConvBitsToWord_3: bBit13
C00838/113	L_ConvBitsToWord_3: bBit14
C00838/114	L_ConvBitsToWord_3: bBit15
C00838/115	L_MckCtrlInterface_1: bManJogPos
C00838/116	L_MckCtrlInterface_1: bManJogNeg
C00838/117	L_MckCtrlInterface_1: bManJogExecute2ndVel
C00838/118	L_MckCtrlInterface_1: bReleaseLimitSwitch
C00838/119	L_MckCtrlInterface_1: bHomingStartStop
C00838/120	L_MckCtrlInterface_1: bHomingSetPos
C00838/121	L_MckCtrlInterface_1: bHomingResetPos
C00838/122	L_MckCtrlInterface_1: bEnableVelOverride
C00838/123	L_MckCtrlInterface_1: bEnableAccOverride
C00838/124	L_MckCtrlInterface_1: bDisableSShaping
C00838/125	L_MckCtrlInterface_1: bPosExecute
C00838/126	L_MckCtrlInterface_1: bPosExecuteFinish
C00838/127	L_MckCtrlInterface_1: bPosDisableFollowProfile
C00838/128	L_MckCtrlInterface_1: bPosStop
C00838/129	L_MckCtrlInterface_1: bPosTeachSetPos
C00838/130	L_MckCtrlInterface_1: bPosTeachActPos
C00838/131	L_MckCtrlInterface_1: bProfileNo_1
C00838/132	L_MckCtrlInterface_1: bProfileNo_2
C00838/133	L_MckCtrlInterface_1: bProfileNo_4
C00838/134	L_MckCtrlInterface_1: bProfileNo_8
C00838/135	L_MckCtrlInterface_1: bOperationMode_1
C00838/136	L_MckCtrlInterface_1: bOperationMode_2
C00838/137	L_MckCtrlInterface_1: bOperationMode_4
C00838/138	L_MckCtrlInterface_1: bOperationMode_8
C00838/139	L_PhaseDiff_1: bEnable
C00838/140	L_PhaseDiff_2: bEnable
C00838/141	L_PhaseDiff_1: bReset
C00838/142	L_PhaseDiff_2: bReset
C00838/143	L_SRFG_1: bLoad
C00838/144	L_SRFG_2: bLoad
C00838/145	L_SignalSwitch_1: bSet
C00838/146	L_SignalSwitch_2: bSet
C00838/147	L_SignalSwitch_3: bSet
C00838/148	L_SignalSwitch_4: bSet
C00838/149	L_Odometer_1: bTriggerPulse

Parameter | Name: **C00838 | Binary inputs (Set2)**	Data type: UNSIGNED_8 Index: 23737_d = $5CB9_h$
C00838/150	L_Odometer_1: bReset
C00838/151	L_FixSet_a_1: bSelect1
C00838/152	L_FixSet_a_1: bSelect2
C00838/153	L_FixSet_a_1: bSelect4
C00838/154	L_FixSet_a_1: bSelect8
C00838/155	L_FixSet_w_1: bSelect1
C00838/156	L_FixSet_w_1: bSelect2
C00838/157	L_FixSet_w_1: bSelect4
C00838/158	L_FixSet_w_1: bSelect8
C00838/159	L_FixSet_w_2: bSelect1
C00838/160	L_FixSet_w_2: bSelect2
C00838/161	L_FixSet_w_2: bSelect4
C00838/162	L_FixSet_w_2: bSelect8
C00838/163	L_CalcDiameter_1: bResetPos
C00838/164	L_CalcDiameter_1: bHoldD
C00838/165	L_CalcDiameter_1: bUnidirect
C00838/166	L_CalcDiameter_1: bUnwind
C00838/167	L_CalcDiameter_1: bLoadDiameter
C00838/168	L_CalcDiameter_1: bCalcRef
C00838/169	L_ProcessCtrl_1: bLoadAct
C00838/170	L_ProcessCtrl_1: bIOff
C00838/171	L_ProcessCtrl_1: bReset
C00838/172	L_PhiIntegrator_1: bTPReceived
C00838/173	L_PhiIntegrator_1: bReset
C00838/174	L_PhiIntegrator_1: bLoad
C00838/175	L_PosCtrlLin_1: bExecute
C00838/176	L_PosCtrlLin_1: bSetPos0
C00838/177	L_PosCtrlLin_1: bPosMode
C00838/178	L_PosCtrlLin_1: bEnable
C00838/179	L_PosCtrlLin_2: bExecute
C00838/180	L_PosCtrlLin_2: bSetPos0
C00838/181	L_PosCtrlLin_2: bPosMode
C00838/182	L_PosCtrlLin_2: bEnable
C00838/183	L_SwitchPoint_1: bDisable
C00838/184	L_DFSET_1: bZeroPulse
C00838/185	L_DFSET_1: bSetTPReceived
C00838/186	L_DFSET_1: bActTPReceived
C00838/187	L_DFSET_1: bSetActIntegrator
C00838/188	L_DFSET_1: bResetAllIntegrators
C00838/189	L_DFRFG_1: bSetTPReceived
C00838/190	L_DFRFG_1: bRfg0
C00838/191	L_DFRFG_1: bRfgStop
C00838/192	L_DFRFG_1: bReset
C00838/193	L_ConvX_1: bInvers

Parameter | Name: **C00838 | Binary inputs (Set2)**	Data type: UNSIGNED_8 Index: 23737_d = $5CB9_h$
C00838/194	L_ConvX_2: bInvers
C00838/195	L_ConvX_3: bInvers
C00838/196	L_ConvPP_1: bAct
C00838/197	L_ConvPP_2: bAct
C00838/198	L_ConvPP_3: bAct
C00838/199	L_SignalSwitch32_1: bSet
C00838/200	L_SignalSwitch32_2: bSet
C00838/201	L_SignalSwitch32_3: bSet
C00838/202	L_Sequencer_1: bStart
C00838/203	L_Sequencer_1: bPause
C00838/204	L_Sequencer_1: bNextStep
C00838/205	L_Sequencer_1: bCancel
C00838/206	L_Sequencer_1: bBreak
C00838/207	L_Sequencer_1: bReset
C00838/208	L_ConvActPos_1: bWriteMaxPos
C00838/209	L_ConvActPos_1: bWriteMinPos
C00838/210	L_MFail_1: bFault
C00838/211	L_MFail_1: bReset
C00838/212	L_Curve_3: bEnableTensionCurve
C00838/213	L_SwitchPointPar_1: bDisable
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1	

C00839

Parameter | Name: **C00839 | 32-bit inputs [incr] (Set2)**			Data type: INTEGER_32 Index: 23736_d = $5CB8_h$
Display in [increments] of 32 bit input values of various blocks • In addition to the parameter C00834.			
Display range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes			**Info**
C00839/1			L_ComparePhi_1: dnIn1_p
C00839/2			L_ComparePhi_1: dnIn2_p
C00839/3			L_ComparePhi_2: dnIn1_p
C00839/4			L_ComparePhi_2: dnIn2_p
C00839/5			L_ComparePhi_3: dnIn1_p
C00839/6			L_ComparePhi_3: dnIn2_p
C00839/7			L_ComparePhi_4: dnIn1_p
C00839/8			L_ComparePhi_4: dnIn2_p
C00839/9			L_ComparePhi_5: dnIn1_p
C00839/10			L_ComparePhi_5: dnIn2_p
C00839/11			L_ArithmetikPhi_1: dnIn1_p
C00839/12			L_ArithmetikPhi_1: dnIn2_p
C00839/13			L_ArithmetikPhi_2: dnIn1_p
C00839/14			L_ArithmetikPhi_2: dnIn2_p

Parameter | Name: **C00839 | 32-bit inputs [incr] (Set2)**	Data type: INTEGER_32 Index: 23736_d = $5CB8_h$
C00839/15	L_ArithmetikPhi_3: dnIn1_p
C00839/16	L_ArithmetikPhi_3: dnIn2_p
C00839/17	L_GainOffsetPhiP_1: dnIn_p
C00839/18	L_GainOffsetPhiP_2: dnIn_p
C00839/19	L_LimitPhi_1: dnIn_p
C00839/20	L_LimitPhi_2: dnIn_p
C00839/21	L_LimitPhi_3: dnIn_p
C00839/22	L_OffsetGainPhiP_1: dnIn_p
C00839/23	L_OffsetGainPhiP_2: dnIn_p
C00839/24	L_PhaseIntK_1: dnSet_p
C00839/25	L_PhaseIntK_2: dnSet_p
C00839/26	L_Mux_1: dnInput1_p
C00839/27	L_Mux_1: dnInput2_p
C00839/28	L_Mux_1: dnInput3_p
C00839/29	L_Mux_1: dnInput4_p
C00839/30	L_Mux_1: dnInput5_p
C00839/31	L_Mux_1: dnInput6_p
C00839/32	L_Mux_1: dnInput7_p
C00839/33	L_Mux_1: dnInput8_p
C00839/34	L_SOrt_1: dnInput_p
C00839/35	L_ConvDIntToWords_1: dnInput_p
C00839/36	L_ConvDIntToWords_2: dnInput_p
C00839/37	L_ConvDIntToWords_3: dnInput_p
C00839/38	L_MckCtrlInterface_1: dnPosSetIn_p
C00839/39	L_PhaseDiff_1: dnSet_p
C00839/40	L_PhaseDiff_2: dnSet_p
C00839/41	L_PhaseDiff_1: dnAdd_p
C00839/42	L_PhaseDiff_2: dnAdd_p
C00839/43	L_MckStateInterface_1: dnPosIn_p
C00839/44	L_Odometer_1: dnInPosition_p
C00839/45	L_PhiIntegrator_1: dnTPPosition_p
C00839/46	L_PhiIntegrator_1: dnLoadVal_p
C00839/47	L_PosCtrlLin_1: dnSet_p
C00839/48	L_PosCtrlLin_1: dnAct_p
C00839/49	L_PosCtrlLin_2: dnSet_p
C00839/50	L_PosCtrlLin_2: dnAct_p
C00839/51	L_SwitchPoint_1: dnActPos_p
C00839/52	L_SwitchPoint_1: dnSwitchPoint1_p
C00839/53	L_SwitchPoint_1: dn2ndPoint_Size1_p
C00839/54	L_SwitchPoint_1: dnSwitchPoint2_p
C00839/55	L_SwitchPoint_1: dn2ndPoint_Size2_p
C00839/56	L_SwitchPoint_1: dnSwitchPoint3_p
C00839/57	L_SwitchPoint_1: dn2ndPoint_Size3_p
C00839/58	L_SwitchPoint_1: dnSwitchPoint4_p

Parameter | Name: **C00839 | 32-bit inputs [incr] (Set2)**	Data type: INTEGER_32 Index: 23736_d = $5CB8_h$
C00839/59	L_SwitchPoint_1: dn2ndPoint_Size4_p
C00839/60	L_DFSET_1: dnSetTPPos_p
C00839/61	L_DFSET_1: dnActTPPos_p
C00839/62	L_DFSET_1: dnPosOffset
C00839/63	L_DFRFG_1: dnSetTPPos_p
C00839/64	L_GearComp_1: dnPhiIn_p
C00839/65	L_ConvPA_1: dnIn_p
C00839/66	L_ConvPA_2: dnIn_p
C00839/67	L_ConvPA_3: dnIn_p
C00839/68	L_ConvPP_1: dnIn_p
C00839/69	L_ConvPP_2: dnIn_p
C00839/70	L_ConvPP_3: dnIn_p
C00839/71	L_SignalSwitch32_1: dnIn1
C00839/72	L_SignalSwitch32_1: dnIn2
C00839/73	L_SignalSwitch32_2: dnIn1
C00839/74	L_SignalSwitch32_2: dnIn2
C00839/75	L_SignalSwitch32_3: dnIn1
C00839/76	L_SignalSwitch32_3: dnIn2
C00839/77	L_ArithmetikPhi_4: adnIn1_p
C00839/78	L_ArithmetikPhi_4: adnIn2_p
C00839/79	L_ArithmetikPhi_5: adnIn1_p
C00839/80	L_ArithmetikPhi_5: adnIn2_p
C00839/81	L_ArithmetikPhi_6: adnIn1_p
C00839/82	L_ArithmetikPhi_6: adnIn2_p
C00839/83	L_PhiDiv_1: dnIn_p
C00839/84	L_PhiAdd_1: dnIn1_p
C00839/85	L_PhiAdd_1: dnIn2_p
C00839/86	L_PhiAdd_1: dnIn3_p
C00839/87	L_DFSET_1: dnDeltaPosIn_p
C00839/88	L_SwitchPoint_1: dnSwitchPoint5_p
C00839/89	L_SwitchPoint_1: dn2ndPoint_Size5_p
C00839/90	L_SwitchPoint_1: dnSwitchPoint6_p
C00839/91	L_SwitchPoint_1: dn2ndPoint_Size6_p
C00839/92	L_SwitchPoint_1: dnSwitchPoint7_p
C00839/93	L_SwitchPoint_1: dn2ndPoint_Size7_p
C00839/94	L_SwitchPoint_1: dnSwitchPoint8_p
C00839/95	L_SwitchPoint_1: dn2ndPoint_Size8_p
C00839/96	L_SwitchPointPar_1: dnActPos_p
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1	

C00840

Parameter | Name: **C00840 | 16-bit inputs I/O level [%]**			Data type: INTEGER_16 Index: 23735_d = $5CB7_h$
Display in percent of 16 bit input values of various blocks of the I/O level			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00840/1			LS_AnalogOutput: nOut1_a (V)
C00840/2			LP_CanOut1: wState
C00840/3			LP_CanOut1: wOut2
C00840/4			LP_CanOut1: wOut3
C00840/5			LP_CanOut1: wOut4
C00840/6			LP_CanOut2: wOut1
C00840/7			LP_CanOut2: wOut2
C00840/8			LP_CanOut2: wOut3
C00840/9			LP_CanOut2: wOut4
C00840/10			LP_CanOut3: wOut1
C00840/11			LP_CanOut3: wOut2
C00840/12			LP_CanOut3: wOut3
C00840/13			LP_CanOut3: wOut4
C00840/14			LS_DisFree_a: nDis1_a
C00840/15			LS_DisFree_a: nDis2_a
C00840/16			LS_DisFree_a: nDis3_a
C00840/17			LS_DisFree_a: nDis4_a
C00840/18			LS_DisFree: wDis1
C00840/19			LS_DisFree: wDis2
C00840/20			LS_DisFree: wDis3
C00840/21			LS_DisFree: wDis4
C00840/22			LP_MciOut: wState
C00840/23			LP_MciOut: wOut2
C00840/24			LP_MciOut: wOut3
C00840/25			LP_MciOut: wOut4
C00840/26			LP_MciOut: wOut5
C00840/27			LP_MciOut: wOut6
C00840/28			LP_MciOut: wOut7
C00840/29			LP_MciOut: wOut8
C00840/30			LP_MciOut: wOut9
C00840/31			LP_MciOut: wOut10
C00840/32			LP_MciOut: wOut11
C00840/33			LP_MciOut: wOut12
C00840/34			LP_MciOut: wOut13
C00840/35			LP_MciOut: wOut14
C00840/36			LP_MciOut: wOut15
C00840/37			LP_MciOut: wOut16
C00840/38			LS_AnalogOutput: nOut2_a (V)

Parameter | Name: **C00840 | 16-bit inputs I/O level [%]**	Data type: INTEGER_16 Index: 23735_d = $5CB7_h$
C00840/39	LS_AnalogOutput: nOut1_a (I)
C00840/40	LS_AnalogOutput: nOut2_a (I)
C00840/41	LS_DisFree_a: nDis5_a
C00840/42	LS_DisFree_a: nDis6_a
C00840/43	LS_DisFree_a: nDis7_a
C00840/44	LS_DisFree_a: nDis8_a
C00840/45	LS_DisFree: wDis5
C00840/46	LS_DisFree: wDis6
C00840/47	LS_DisFree: wDis7
C00840/48	LS_DisFree: wDis8
C00840/49	LS_ParReadWrite_1: wParIndex
C00840/50	LS_ParReadWrite_1: wParSubindex
C00840/51	LS_ParReadWrite_1: wInHWord
C00840/52	LS_ParReadWrite_1: wInLWord
C00840/53	LS_ParReadWrite_2: wParIndex
C00840/54	LS_ParReadWrite_2: wParSubindex
C00840/55	LS_ParReadWrite_2: wInHWord
C00840/56	LS_ParReadWrite_2: wInLWord
C00840/57	LS_ParReadWrite_3: wParIndex
C00840/58	LS_ParReadWrite_3: wParSubindex
C00840/59	LS_ParReadWrite_3: wInHWord
C00840/60	LS_ParReadWrite_3: wInLWord
C00840/61	LS_ParReadWrite_4: wParIndex
C00840/62	LS_ParReadWrite_4: wParSubindex
C00840/63	LS_ParReadWrite_4: wInHWord
C00840/64	LS_ParReadWrite_4: wInLWord
C00840/65	LS_ParReadWrite_5: wParIndex
C00840/66	LS_ParReadWrite_5: wParSubindex
C00840/67	LS_ParReadWrite_5: wInHWord
C00840/68	LS_ParReadWrite_5: wInLWord
C00840/69	LS_ParReadWrite_6: wParIndex
C00840/70	LS_ParReadWrite_6: wParSubindex
C00840/71	LS_ParReadWrite_6: wInHWord
C00840/72	LS_ParReadWrite_6: wInLWord
C00840/73	Reserved
C00840/74	Reserved
C00840/75	Reserved
C00840/76	Reserved
C00840/77	Reserved
C00840/78	Reserved
C00840/79	Reserved
C00840/80	Reserved
C00840/81	Reserved
C00840/82	Reserved

Parameter | Name: **C00840 | 16-bit inputs I/O level [%]**	Data type: INTEGER_16 Index: 23735_d = $5CB7_h$
C00840/83	Reserved
C00840/84	Reserved
C00840/85	Reserved
C00840/86	Reserved
C00840/87	Reserved
C00840/88	Reserved
C00840/89	LS_RetainData: wIn1
C00840/90	LS_RetainData: wIn2
C00840/91	LS_RetainData: wIn3
C00840/92	LS_RetainData: wIn4
C00840/93	LS_AxisBusOut: wLine1
C00840/94	LS_AxisBusOut: wLine2
C00840/95	LS_AxisBusOut: wLine3
C00840/96	LS_AxisBusOut: wCas1
C00840/97	LS_AxisBusOut: wCas2
C00840/98	LS_AxisBusOut: wCas3
C00840/99	LS_AxisBusOut: wCas4
C00840/100	LS_AxisBusAux: wAuxOut1
C00840/101	LS_AxisBusAux: wAuxOut2
C00840/102	LS_AxisBusAux: wAuxOut3
C00840/103	LS_AxisBusAux: wAuxOut4
C00840/104	LS_AxisBusAux: wSelectSlave
C00840/105	LS_DFOut: nOut_v
C00840/106	LS_MultiEncoder: wActPosExternalHW
C00840/107	LS_MultiEncoder: wActPosExternalLW
C00840/108	LP_CanOut4: wOut1
C00840/109	LP_CanOut4: wOut2
C00840/110	LP_CanOut4: wOut3
C00840/111	LP_CanOut4: wOut4
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100	

C00841

Parameter | Name: **C00841 | 16-bit inputs I/O level**			Data type: UNSIGNED_16 Index: 23734_d = $5CB6_h$
Decimal/hexadecimal/bit coded display of 16 bit input values of various blocks of the I/O level			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
...	...		
Bit 15	Active		
Subcodes			**Info**
C00841/1			LS_AnalogOutput: nOut1_a (V)
C00841/2			LP_CanOut1: wState

Parameter | Name: **C00841 | 16-bit inputs I/O level**	Data type: UNSIGNED_16 Index: 23734_d = $5CB6_h$
C00841/3	LP_CanOut1: wOut2
C00841/4	LP_CanOut1: wOut3
C00841/5	LP_CanOut1: wOut4
C00841/6	LP_CanOut2: wOut1
C00841/7	LP_CanOut2: wOut2
C00841/8	LP_CanOut2: wOut3
C00841/9	LP_CanOut2: wOut4
C00841/10	LP_CanOut3: wOut1
C00841/11	LP_CanOut3: wOut2
C00841/12	LP_CanOut3: wOut3
C00841/13	LP_CanOut3: wOut4
C00841/14	LS_DisFree_a: nDis1_a
C00841/15	LS_DisFree_a: nDis2_a
C00841/16	LS_DisFree_a: nDis3_a
C00841/17	LS_DisFree_a: nDis4_a
C00841/18	LS_DisFree: wDis1
C00841/19	LS_DisFree: wDis2
C00841/20	LS_DisFree: wDis3
C00841/21	LS_DisFree: wDis4
C00841/22	LP_MciOut: wState
C00841/23	LP_MciOut: wOut2
C00841/24	LP_MciOut: wOut3
C00841/25	LP_MciOut: wOut4
C00841/26	LP_MciOut: wOut5
C00841/27	LP_MciOut: wOut6
C00841/28	LP_MciOut: wOut7
C00841/29	LP_MciOut: wOut8
C00841/30	LP_MciOut: wOut9
C00841/31	LP_MciOut: wOut10
C00841/32	LP_MciOut: wOut11
C00841/33	LP_MciOut: wOut12
C00841/34	LP_MciOut: wOut13
C00841/35	LP_MciOut: wOut14
C00841/36	LP_MciOut: wOut15
C00841/37	LP_MciOut: wOut16
C00841/38	LS_AnalogOutput: nOut2_a (V)
C00841/39	LS_AnalogOutput: nOut1_a (I)
C00841/40	LS_AnalogOutput: nOut2_a (I)
C00841/41	LS_DisFree_a: nDis5_a
C00841/42	LS_DisFree_a: nDis6_a
C00841/43	LS_DisFree_a: nDis7_a
C00841/44	LS_DisFree_a: nDis8_a
C00841/45	LS_DisFree: wDis5
C00841/46	LS_DisFree: wDis6

Parameter | Name: **C00841 | 16-bit inputs I/O level**	Data type: UNSIGNED_16 Index: 23734_d = $5CB6_h$
C00841/47	LS_DisFree: wDis7
C00841/48	LS_DisFree: wDis8
C00841/49	LS_ParReadWrite_1: wParIndex
C00841/50	LS_ParReadWrite_1: wParSubindex
C00841/51	LS_ParReadWrite_1: wInHWord
C00841/52	LS_ParReadWrite_1: wInLWord
C00841/53	LS_ParReadWrite_2: wParIndex
C00841/54	LS_ParReadWrite_2: wParSubindex
C00841/55	LS_ParReadWrite_2: wInHWord
C00841/56	LS_ParReadWrite_2: wInLWord
C00841/57	LS_ParReadWrite_3: wParIndex
C00841/58	LS_ParReadWrite_3: wParSubindex
C00841/59	LS_ParReadWrite_3: wInHWord
C00841/60	LS_ParReadWrite_3: wInLWord
C00841/61	LS_ParReadWrite_4: wParIndex
C00841/62	LS_ParReadWrite_4: wParSubindex
C00841/63	LS_ParReadWrite_4: wInHWord
C00841/64	LS_ParReadWrite_4: wInLWord
C00841/65	LS_ParReadWrite_5: wParIndex
C00841/66	LS_ParReadWrite_5: wParSubindex
C00841/67	LS_ParReadWrite_5: wInHWord
C00841/68	LS_ParReadWrite_5: wInLWord
C00841/69	LS_ParReadWrite_6: wParIndex
C00841/70	LS_ParReadWrite_6: wParSubindex
C00841/71	LS_ParReadWrite_6: wInHWord
C00841/72	LS_ParReadWrite_6: wInLWord
C00841/73	Reserved
C00841/74	Reserved
C00841/75	Reserved
C00841/76	Reserved
C00841/77	Reserved
C00841/78	Reserved
C00841/79	Reserved
C00841/80	Reserved
C00841/81	Reserved
C00841/82	Reserved
C00841/83	Reserved
C00841/84	Reserved
C00841/85	Reserved
C00841/86	Reserved
C00841/87	Reserved
C00841/88	Reserved
C00841/89	LS_RetainData: wIn1
C00841/90	LS_RetainData: wIn2

Parameter | Name: **C00841 | 16-bit inputs I/O level**	Data type: UNSIGNED_16 Index: 23734_d = $5CB6_h$
C00841/91	LS_RetainData: wIn3
C00841/92	LS_RetainData: wIn4
C00841/93	LS_AxisBusOut: wLine1
C00841/94	LS_AxisBusOut: wLine2
C00841/95	LS_AxisBusOut: wLine3
C00841/96	LS_AxisBusOut: wCas1
C00841/97	LS_AxisBusOut: wCas2
C00841/98	LS_AxisBusOut: wCas3
C00841/99	LS_AxisBusOut: wCas4
C00841/100	LS_AxisBusAux: wAuxOut1
C00841/101	LS_AxisBusAux: wAuxOut2
C00841/102	LS_AxisBusAux: wAuxOut3
C00841/103	LS_AxisBusAux: wAuxOut4
C00841/104	LS_AxisBusAux: wSelectSlave
C00841/105	LS_DFOut: nOut_v
C00841/106	LS_MultiEncoder: wActPosExternalHW
C00841/107	LS_MultiEncoder: wActPosExternalLW
C00841/108	LP_CanOut4: wOut1
C00841/109	LP_CanOut4: wOut2
C00841/110	LP_CanOut4: wOut3
C00841/111	LP_CanOut4: wOut4
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT	

C00843

Parameter | Name: **C00843 | Binary inputs I/O level**		Data type: UNSIGNED_8 Index: 23732_d = $5CB4_h$
Display of the signal status of the binary inputs of different I/O level blocks		
Selection list		
0	False	
1	True	
Subcodes		**Info**
C00843/1		LS_DigitalOutput: bRelay
C00843/2		LS_DigitalOutput: bOut1
C00843/3		LS_DigitalInput: bCountIn1_Reset
C00843/4		LS_DigitalInput: bCountIn1_LoadStartValue
C00843/5		LP_CanOut1: bState_B0
C00843/6		LP_CanOut1: bState_B1
C00843/7		LP_CanOut1: bState_B2
C00843/8		LP_CanOut1: bState_B3
C00843/9		LP_CanOut1: bState_B4
C00843/10		LP_CanOut1: bState_B5
C00843/11		LP_CanOut1: bState_B6
C00843/12		LP_CanOut1: bState_B7
C00843/13		LP_CanOut1: bState_B8

Parameter | Name: **C00843 | Binary inputs I/O level**	Data type: UNSIGNED_8 Index: 23732_d = $5CB4_h$
C00843/14	LP_CanOut1: bState_B9
C00843/15	LP_CanOut1: bState_B10
C00843/16	LP_CanOut1: bState_B11
C00843/17	LP_CanOut1: bState_B12
C00843/18	LP_CanOut1: bState_B13
C00843/19	LP_CanOut1: bState_B14
C00843/20	LP_CanOut1: bState_B15
C00843/21	LS_DisFree_b: bDis1
C00843/22	LS_DisFree_b: bDis2
C00843/23	LS_DisFree_b: bDis3
C00843/24	LS_DisFree_b: bDis4
C00843/25	LS_DisFree_b: bDis5
C00843/26	LS_DisFree_b: bDis6
C00843/27	LS_DisFree_b: bDis7
C00843/28	LS_DisFree_b: bDis8
C00843/29	LP_CanOut2: bOut1_B0
C00843/30	LP_CanOut2: bOut1_B1
C00843/31	LP_CanOut2: bOut1_B2
C00843/32	LP_CanOut2: bOut1_B3
C00843/33	LP_CanOut2: bOut1_B4
C00843/34	LP_CanOut2: bOut1_B5
C00843/35	LP_CanOut2: bOut1_B6
C00843/36	LP_CanOut2: bOut1_B7
C00843/37	LP_CanOut2: bOut1_B8
C00843/38	LP_CanOut2: bOut1_B9
C00843/39	LP_CanOut2: bOut1_B10
C00843/40	LP_CanOut2: bOut1_B11
C00843/41	LP_CanOut2: bOut1_B12
C00843/42	LP_CanOut2: bOut1_B13
C00843/43	LP_CanOut2: bOut1_B14
C00843/44	LP_CanOut2: bOut1_B15
C00843/45	LP_CanOut3: bOut1_B0
C00843/46	LP_CanOut3: bOut1_B1
C00843/47	LP_CanOut3: bOut1_B2
C00843/48	LP_CanOut3: bOut1_B3
C00843/49	LP_CanOut3: bOut1_B4
C00843/50	LP_CanOut3: bOut1_B5
C00843/51	LP_CanOut3: bOut1_B6
C00843/52	LP_CanOut3: bOut1_B7
C00843/53	LP_CanOut3: bOut1_B8
C00843/54	LP_CanOut3: bOut1_B9
C00843/55	LP_CanOut3: bOut1_B10
C00843/56	LP_CanOut3: bOut1_B11
C00843/57	LP_CanOut3: bOut1_B12

Parameter | Name: **C00843 | Binary inputs I/O level**	Data type: UNSIGNED_8 Index: 23732_d = $5CB4_h$
C00843/58	LP_CanOut3: bOut1_B13
C00843/59	LP_CanOut3: bOut1_B14
C00843/60	LP_CanOut3: bOut1_B15
C00843/61	LP_MciOut: bState_B0
C00843/62	LP_MciOut: bState_B1
C00843/63	LP_MciOut: bState_B2
C00843/64	LP_MciOut: bState_B3
C00843/65	LP_MciOut: bState_B4
C00843/66	LP_MciOut: bState_B5
C00843/67	LP_MciOut: bState_B6
C00843/68	LP_MciOut: bState_B7
C00843/69	LP_MciOut: bState_B8
C00843/70	LP_MciOut: bState_B9
C00843/71	LP_MciOut: bState_B10
C00843/72	LP_MciOut: bState_B11
C00843/73	LP_MciOut: bState_B12
C00843/74	LP_MciOut: bState_B13
C00843/75	LP_MciOut: bState_B14
C00843/76	LP_MciOut: bState_B15
C00843/77	LP_MciOut: bOut2_B0
C00843/78	LP_MciOut: bOut2_B1
C00843/79	LP_MciOut: bOut2_B2
C00843/80	LP_MciOut: bOut2_B3
C00843/81	LP_MciOut: bOut2_B4
C00843/82	LP_MciOut: bOut2_B5
C00843/83	LP_MciOut: bOut2_B6
C00843/84	LP_MciOut: bOut2_B7
C00843/85	LP_MciOut: bOut2_B8
C00843/86	LP_MciOut: bOut2_B9
C00843/87	LP_MciOut: bOut2_B10
C00843/88	LP_MciOut: bOut2_B11
C00843/89	LP_MciOut: bOut2_B12
C00843/90	LP_MciOut: bOut2_B13
C00843/91	LP_MciOut: bOut2_B14
C00843/92	LP_MciOut: bOut2_B15
C00843/93	LS_SetError_1: bSetError1
C00843/94	LS_SetError_1: bSetError2
C00843/95	LS_SetError_1: bSetError3
C00843/96	LS_SetError_1: bSetError4
C00843/97	LS_DigitalInput: bCountIn6_Reset
C00843/98	LS_DigitalInput: bCountIn6_LoadStartValue
C00843/99	LS_DigitalOutput: bOut2
C00843/100	LS_DigitalOutput: bOut3
C00843/101	LS_DigitalOutput: bOut HighCurrent

Parameter | Name: **C00843 | Binary inputs I/O level**	Data type: UNSIGNED_8 Index: 23732_d = $5CB4_h$
C00843/102	LS_DisFree_b: bDis9
C00843/103	LS_DisFree_b: bDis10
C00843/104	LS_DisFree_b: bDis11
C00843/105	LS_DisFree_b: bDis12
C00843/106	LS_DisFree_b: bDis13
C00843/107	LS_DisFree_b: bDis14
C00843/108	LS_DisFree_b: bDis15
C00843/109	LS_DisFree_b: bDis16
C00843/110	Reserved
C00843/111	LS_ParReadWrite_1: bExecute
C00843/112	LS_ParReadWrite_1: bReadWrite
C00843/113	LS_ParReadWrite_2: bExecute
C00843/114	LS_ParReadWrite_2: bReadWrite
C00843/115	LS_ParReadWrite_3: bExecute
C00843/116	LS_ParReadWrite_3: bReadWrite
C00843/117	LS_ParReadWrite_4: bExecute
C00843/118	LS_ParReadWrite_4: bReadWrite
C00843/119	LS_ParReadWrite_5: bExecute
C00843/120	LS_ParReadWrite_5: bReadWrite
C00843/121	LS_ParReadWrite_6: bExecute
C00843/122	LS_ParReadWrite_6: bReadWrite
C00843/123	LS_WriteParamList: bExecute
C00843/124	LS_WriteParamList: bSelectWriteValue_1
C00843/125	LS_WriteParamList: bSelectWriteValue_2
C00843/126	LS_CANManagement: bResetNode
C00843/127	LS_CANManagement: bReInitCAN
C00843/128	LS_DigitalInput: bPosIn12_Load
C00843/129	Reserved
C00843/130	Reserved
C00843/131	Reserved
C00843/132	Reserved
C00843/133	Reserved
C00843/134	Reserved
C00843/135	Reserved
C00843/136	Reserved
C00843/137	Reserved
C00843/138	Reserved
C00843/139	Reserved
C00843/140	Reserved
C00843/141	Reserved
C00843/142	Reserved
C00843/143	Reserved
C00843/144	Reserved
C00843/145	Reserved

Parameter | Name: **C00843 | Binary inputs I/O level**	Data type: UNSIGNED_8 Index: 23732_d = $5CB4_h$
C00843/146	Reserved
C00843/147	LS_TouchProbe: bDisableTPDigIn3
C00843/148	LS_TouchProbe: bDisableTPDigIn4
C00843/149	LS_TouchProbe: bDisableTPDigIn5
C00843/150	LS_TouchProbe: bDisableTPDigIn6
C00843/151	LS_TouchProbe: bDisableTPDigIn7
C00843/152	LS_TouchProbe: bDisableTPDigIn3_Rising
C00843/153	LS_TouchProbe: bDisableTPDigIn3_Falling
C00843/154	LS_TouchProbe: bDisableTPDigIn4_Rising
C00843/155	LS_TouchProbe: bDisableTPDigIn4_Falling
C00843/156	LS_TouchProbe: bDisableTPDigIn5_Rising
C00843/157	LS_TouchProbe: bDisableTPDigIn5_Falling
C00843/158	LS_TouchProbe: bDisableTPDigIn3Window
C00843/159	LS_TouchProbe: bDisableTPDigIn4Window
C00843/160	LS_TouchProbe: bDisableTPDigIn5Window
C00843/161	LS_AxisBusIO: bSetFail
C00843/162	LS_AxisBusIO: bResetFail
C00843/163	Reserved
C00843/164	LS_MultiEncoder: bSetRef
C00843/165	LS_RetainData: bSetRetain_1
C00843/166	LS_RetainData: bSetRetain_2
C00843/167	LS_RetainData: bSetRetain_3
C00843/168	LS_RetainData: bLoadParams
C00843/169	LS_RetainData: bIn1
C00843/170	LS_RetainData: bIn2
C00843/171	LS_RetainData: bIn3
C00843/172	LS_RetainData: bIn4
C00843/173	LS_AxisBusAux: bReadWrite
C00843/174	LS_AxisBusAux: bExecute
C00843/175	LS_AxisBusAux: bStop
C00843/176	LS_DigitalOutput: bUserLED
C00843/177	LS_TouchProbe: bDisableTPEncoderWindow
C00843/178	LS_TouchProbe: bDisableTPResolverWindow
C00843/179	LS_TouchProbe: bDisableTPEncoder
C00843/180	LS_TouchProbe: bDisableTPResolver
C00843/181	LS_DFOut: bSynRdy
C00843/182	Reserved
C00843/183	Reserved
C00843/184	LP_CanOut4: bOut1_B0
C00843/185	LP_CanOut4: bOut1_B1
C00843/186	LP_CanOut4: bOut1_B2
C00843/187	LP_CanOut4: bOut1_B3
C00843/188	LP_CanOut4: bOut1_B4
C00843/189	LP_CanOut4: bOut1_B5

Parameter | Name: **C00843 | Binary inputs I/O level**	Data type: UNSIGNED_8 Index: 23732_d = $5CB4_h$
C00843/190	LP_CanOut4: bOut1_B6
C00843/191	LP_CanOut4: bOut1_B7
C00843/192	LP_CanOut4: bOut1_B8
C00843/193	LP_CanOut4: bOut1_B9
C00843/194	LP_CanOut4: bOut1_B10
C00843/195	LP_CanOut4: bOut1_B11
C00843/196	LP_CanOut4: bOut1_B12
C00843/197	LP_CanOut4: bOut1_B13
C00843/198	LP_CanOut4: bOut1_B14
C00843/199	LP_CanOut4: bOut1_B15
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1	

C00844

Parameter | Name: **C00844 | 32-bit inputs I/O level [incr]**			Data type: INTEGER_32 Index: 23731_d = $5CB3_h$
Display of 32-bit input values of different I/O level blocks in [increments]			
Display range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes			**Info**
C00844/1			LS_DisFree_p: dnDis1_p
C00844/2			LS_DisFree_p: dnDis2_p
C00844/3			LS_DisFree_p: dnDis3_p
C00844/4			LS_DisFree_p: dnDis4_p
C00844/5			LS_DisFree_p: dnDis5_p
C00844/6			LS_DisFree_p: dnDis6_p
C00844/7			LS_DisFree_p: dnDis7_p
C00844/8			LS_DisFree_p: dnDis8_p
C00844/9			LP_CanOut1: dnOut34_p
C00844/10			LP_CanOut2: dnOut34_p
C00844/11			LP_CanOut3: dnOut34_p
C00844/12			LP_MciOut: dnOut34_p
C00844/13			LS_DigitalInput: dnPosIn12_Set_p
C00844/14			Reserved
C00844/15			Reserved
C00844/16			LS_RetainData: dnIn1
C00844/17			LS_RetainData: dnIn2
C00844/18			LS_RetainData: dnIn3
C00844/19			LS_RetainData: dnIn4
C00844/20			LS_AxisBusOut: dnLine12
C00844/21			LS_AxisBusOut: dnCas12
C00844/22			LS_AxisBusOut: dnCas34
C00844/23			LS_AxisBusAux: dnAuxOut12
C00844/24			LS_AxisBusAux: dnAuxOut34
C00844/25			LP_CanOut4: dnOut34_p
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00866

Parameter | Name: **C00866 | CAN input words**			Data type: UNSIGNED_16 Index: 23709_d = $5C9D_h$
Display of the 16 bit input values of the CAN interface ▶ "CAN on board" system bus			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
...	...		
Bit 15	Active		
Subcodes			**Info**
C00866/1			LP_CanIn1: wCtrl
C00866/2			LP_CanIn1: wIn2
C00866/3			LP_CanIn1: wIn3
C00866/4			LP_CanIn1: wIn4
C00866/5			LP_CanIn2: wIn1
C00866/6			LP_CanIn2: wIn2
C00866/7			LP_CanIn2: wIn3
C00866/8			LP_CanIn2: wIn4
C00866/9			LP_CanIn3: wIn1
C00866/10			LP_CanIn3: wIn2
C00866/11			LP_CanIn3: wIn3
C00866/12			LP_CanIn3: wIn4
C00866/13			LP_CanIn4: wIn1 • From version 15.00.00
C00866/14			LP_CanIn4: wIn2 • From version 15.00.00
C00866/15			LP_CanIn4: wIn3 • From version 15.00.00
C00866/16			LP_CanIn4: wIn4 • From version 15.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00868

Parameter | Name: **C00868 | CAN output words**			Data type: UNSIGNED_16 Index: 23707_d = $5C9B_h$
Display of the 16 bit output values of the CAN interface			▶ "CAN on board" system bus
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
Bit 1	Active		
Bit 2	Active		
Bit 3	Active		
Bit 4	Active		
Bit 5	Active		
Bit 6	Active		
Bit 7	Active		
Bit 8	Active		
Bit 9	Active		
Bit 10	Active		
Bit 11	Active		
Bit 12	Active		
Bit 13	Active		
Bit 14	Active		
Bit 15	Active		
Subcodes			**Info**
C00868/1			LP_CanOut1: wState
C00868/2			LP_CanOut1: wOut2
C00868/3			LP_CanOut1: wOut3
C00868/4			LP_CanOut1: wOut4
C00868/5			LP_CanOut2: wOut1
C00868/6			LP_CanOut2: wOut2
C00868/7			LP_CanOut2: wOut3
C00868/8			LP_CanOut2: wOut4
C00868/9			LP_CanOut3: wOut1
C00868/10			LP_CanOut3: wOut2
C00868/11			LP_CanOut3: wOut3
C00868/12			LP_CanOut3: wOut4
C00868/13			LP_CanOut4: wOut1 • From version 15.00.00
C00868/14			LP_CanOut4: wOut2 • From version 15.00.00
C00868/15			LP_CanOut4: wOut3 • From version 15.00.00
C00868/16			LP_CanOut4: wOut4 • From version 15.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00876

Parameter | Name: **C00876 | MCI input words**			Data type: UNSIGNED_16 Index: 23699_d = $5C93_h$
Display of the 16 bit input values of the communication module			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
...	...		
Bit 15	Active		
Subcodes			**Info**
C00876/1			LP_MciIn: wCtrl
C00876/2			LP_MciIn: wIn2
C00876/3			LP_MciIn: wIn3
C00876/4			LP_MciIn: wIn4
C00876/5			LP_MciIn: wIn5
C00876/6			LP_MciIn: wIn6
C00876/7			LP_MciIn: wIn7
C00876/8			LP_MciIn: wIn8
C00876/9			LP_MciIn: wIn9
C00876/10			LP_MciIn: wIn10
C00876/11			LP_MciIn: wIn11
C00876/12			LP_MciIn: wIn12
C00876/13			LP_MciIn: wIn13
C00876/14			LP_MciIn: wIn14
C00876/15			LP_MciIn: wIn15
C00876/16			LP_MciIn: wIn16
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00877

Parameter | Name: **C00877 | MCI output words**			Data type: UNSIGNED_16 Index: 23698$_d$ = 5C92$_h$
Display of the 16 bit output values of the communication module			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Active		
...	...		
Bit 15	Active		
Subcodes			**Info**
C00877/1			LP_MciOut: wState
C00877/2			LP_MciOut: wOut2
C00877/3			LP_MciOut: wOut3
C00877/4			LP_MciOut: wOut4
C00877/5			LP_MciOut: wOut5
C00877/6			LP_MciOut: wOut6
C00877/7			LP_MciOut: wOut7
C00877/8			LP_MciOut: wOut8
C00877/9			LP_MciOut: wOut9
C00877/10			LP_MciOut: wOut10
C00877/11			LP_MciOut: wOut11
C00877/12			LP_MciOut: wOut12
C00877/13			LP_MciOut: wOut13
C00877/14			LP_MciOut: wOut14
C00877/15			LP_MciOut: wOut15
C00877/16			LP_MciOut: wOut16
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C00890

Parameter | Name: **C00890 | MCI_InOut: Inversion**			Data type: UNSIGNED_16 Index: 23685$_d$ = 5C85$_h$
This parameter serves to invert the control/status bits of the MCI port blocks.			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Active		Bit set = inversion active
...	...		
Bit 15	Active		
Subcodes	**Lenze setting**		**Info**
C00890/1	0x0000		LP_MciIn: Invert.Ctrl_B0..15
C00890/2	0x0000		LP_MciOut: Invert.State_B0..15
C00890/3	0x0000		LP_MciIn: Invert. In2_B0..15
C00890/4	0x0000		LP_MciOut: Invert.Out2_B0..15
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C00905

Parameter | Name: **C00905 | Motor phase direction of rotation**		Data type: UNSIGNED_8 Index: 23670_d = $5C76_h$
To correct such misconnected motor phases, the rotating field of the inverter's output can be reversed by selecting "1: Inverted". In this case, a phase will be reversed at the output of the inverter. **Note:** Up to and including version 07.00.00, this function may only be activated for the following motor control types: • V/f characteristic control (VFCplus) • Energy-saving V/f characteristic control (VFCplusEco) For all other motor control types, this function must not be activated because the set control mode would not work in that case! From version 12.00.00, this function may be activated for all motor control types with the exception of the ones for synchronous motor. The activation of this function does not affect the control types for synchronous motor since these control types require an in-phase connection of the synchronous motor.		
Selection list(Lenze setting printed in bold)		
0	**Not inverted**	
1	Inverted	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00909

Parameter | Name: **C00909 | Speed limitation**			Data type: INTEGER_16 Index: 23666_d = $5C72_h$
Max. positive/negative speed for all motor control modes			
Setting range (min. value | unit | max. value)			
0.00	%	175.00	
Subcodes	**Lenze setting**		**Info**
C00909/1	120.00 %		Max. pos. speed
C00909/2	120.00 %		Max. neg. speed
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00910

Parameter | Name: **C00910 | Frequency limitation**			Data type: UNSIGNED_16 Index: 23665_d = $5C71_h$
Max. positive/negative output frequency for all motor control modes • For dependencies see chapters "Override" and "Profile entry".			
Setting range (min. value | unit | max. value)			
0	Hz	1300	
Subcodes	**Lenze setting**		**Info**
C00910/1	1000 Hz		Max. pos. output frequency
C00910/2	1000 Hz		Max. neg. output frequency
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00915

Parameter | Name: **C00915 | Motor cable length**			Data type: UNSIGNED_16 Index: 23660_d = $5C6C_h$
Single motor cable length for calculating the motor cable resistance • The calculated motor cable resistance is displayed in C00917.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	m	1000.0	**5.0 m**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00916

Parameter | Name: **C00916 | Motor cable cross-section**			Data type: UNSIGNED_16 Index: 23659_d = $5C6B_h$
Motor cable cross-section of a phase/cable for calculating the motor cable resistance • The calculated motor cable resistance is displayed in C00917.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.50	mm^2	100.00	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00917

Parameter | Name: **C00917 | Motor cable resistance**			Data type: UNSIGNED_16 Index: 23658_d = $5C6A_h$
Display of the motor cable resistance of a motor cable phase • The motor cable resistance is calculated from the motor cable length set in C00915 and the motor cable cross-section set in C00916.			
Display range (min. value | unit | max. value)			
0	mohm	64000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00918

Parameter | Name: **C00918 | SC: Start motor magnetising current**			Data type: UNSIGNED_8 Index: 23657_d = $5C69_h$
If the servo control (SC) mode is selected and the inverter is enabled, the speed setpoint for motor control will only be enabled when the motor magnetising current has reached the threshold value set here. • The adjustable percentage threshold value refers to the rated magnetising current (C00095).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	90	**87 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00919

Parameter | Name: **C00919 | Moment of inertia from load**			Data type: UNSIGNED_32 Index: 23656_d = $5C68_h$
From version 12.00.00 In order to take account of mass inertias changing during the process (e.g. reels) when optimising the response to setpoint changes, the maximum value of the changing moment of inertia must be set here. The process signal *nInertiaAdapt_a* of the LS_MotorInterface SB can then be used in the process to dynamically control what percentage of it is to be taken into account for the setpoint feedforward control.			
Setting range (min. value | unit | max. value)			
0.00	kg cm^2	6000000.00	
Subcodes	**Lenze setting**		**Info**
C00919/1	0.00 kg cm^2		Load moment of inertia
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00920

Parameter | Name: **C00920 | Rated device currents**			Data type: UNSIGNED_16 Index: 23655_d = $5C67_h$
Display range (min. value | unit | max. value)			
0.0	A	6000.0	
Subcodes			**Info**
C00920/1			Rated current 3ph 400V/1ph 230V
C00920/2			Rated current 3ph 440V
C00920/3			Rated current 3ph 480V
C00920/4			Rated current 3ph 500V
C00920/5			Increased rated current 3ph 400V/1ph 230V
C00920/6			Increased rated current 3ph 440V
C00920/7			Increased rated curent 3ph 480V
C00920/8			Increased rated current 3ph 500V
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00922

Parameter | Name: **C00922 | ICM_DiagnosticCounter**	Data type: UNSIGNED_16 Index: 23653_d = $5C65_h$
This code is used device-internally and must not be written by the user side!	

C00925

Parameter | Name: **C00925 | LS_Resolver: Number of pole pairs**			Data type: UNSIGNED_8 Index: 23650_d = $5C62_h$
			▸ Encoder/feedback system
Setting range (min. value | unit | max. value)			**Lenze setting**
1		10	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00926

Parameter | Name: **C00926 | Pole position**			Data type: INTEGER_16 Index: 23649_d = $5C61_h$
			▶ Encoder/feedback system
Setting range (min. value | unit | max. value)			
-179.9	°	179.9	
Subcodes	**Lenze setting**		**Information**
C00926/1	-90.0 °		LS_Resolver: Pole position
C00926/2	0.0 °		LS_MultiEncoder: Pole position
C00926/3	0.0 °		LS_Resolver: NP Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00927

Parameter | Name: **C00927 | Motor rotor position**			Data type: UNSIGNED_16 Index: 23648_d = $5C60_h$
			▶ Encoder/feedback system
Display range (min. value | unit | max. value)			
0		2047	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00936

Parameter | Name: **C00936 | SLPSM: Load value — speed controller**			Data type: UNSIGNED_16 Index: 23639_d = $5C57_h$
Setting range (min. value | unit | max. value)			
0.00	%	200.00	
Subcodes	**Lenze setting**		**Info**
C00936/1	0.00 %		SLPSM: Load value — speed controller
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00937

Parameter | Name: **C00937 | Field-oriented motor currents**			Data type: INTEGER_16 Index: 23638_d = $5C56_h$
From version 02.00.00			▶ Field weakening for synchronous motors
Display range (min. value | unit | max. value)			
-320.00	A	320.00	
Subcodes			**Info**
C00937/1			Field-producing current
C00937/2			Torque-producing current
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00938

Parameter | Name: **C00938 | PSM: Maximum motor current field weakening**			Data type: UNSIGNED_16 Index: 23637$_d$ = 5C55$_h$
From version 02.00.00			▸ Field weakening for synchronous motors
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	500.00	**30.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00939

Parameter | Name: **C00939 | Ultimate motor current**			Data type: UNSIGNED_16 Index: 23636$_d$ = 5C54$_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	A	3000.0	**3000.0 A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00940

Parameter | Name: **C00940 | L_ConvW numerator**			Data type: INTEGER_16 Index: 23635$_d$ = 5C53$_h$
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C00940/1	1		L_ConvW_1: Numerator
C00940/2	1		L_ConvW_2: Numerator
C00940/3	1		L_ConvW_3: Numerator
C00940/4	1		L_ConvW_4: Numerator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00941

Parameter | Name: **C00941 | L_ConvW denominator**			Data type: INTEGER_16 Index: 23634$_d$ = 5C52$_h$
Setting range (min. value | unit | max. value)			
1		32767	
Subcodes	**Lenze setting**		**Info**
C00941/1	1		L_ConvW_1: Denominator
C00941/2	1		L_ConvW_2: Denominator
C00941/3	1		L_ConvW_3: Denominator
C00941/4	1		L_ConvW_4: Denominator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00942

Parameter | Name: **C00942 | L_ConvW conversion method**		Data type: UNSIGNED_8 Index: 23633_d = $5C51_h$
Selection list		
0	no conversion	
1	from [%] into [incr./ms]	
2	from [incr./ms] into [%]	
3	Factors signed	
4	Factors unsigned	
Subcodes	**Lenze setting**	**Info**
C00942/1	0: No conversion	L_ConvW_1: Conversion method
C00942/2	0: No conversion	L_ConvW_2: Conversion method
C00942/3	0: No conversion	L_ConvW_3: Conversion method
C00942/4	0: No conversion	L_ConvW_4: Conversion method
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00950

Parameter | Name: **C00950 | L_Interpolator_1: Activation FB functions**		Data type: UNSIGNED_8 Index: 23625_d = $5C49_h$
The L_Interpolator_1 FB: Activation of signal interpolation and signal monitoring		
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C00950/1	0: Off	L_Interpolator_1: Signal interpolation
C00950/2	0: Off	L_Interpolator_1: Signal monitoring
C00950/3	0: Off	L_Interpolator_1: Master value monitoring
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00951

Parameter | Name: **C00951 | L_Interpolator_1: No. of interpolation steps**			Data type: UNSIGNED_16 Index: 23624_d = $5C48_h$
The L_Interpolator_1 FB: No. of interpolation steps			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00952

Parameter | Name: **C00952 | L_Interpolator_1: Limit value - error cycles**			Data type: UNSIGNED_16 Index: 23623_d = $5C47_h$
The L_Interpolator_1 FB: Limit value for missing data telegrams			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**5**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00953

Parameter | Name: **C00953 | L_Interpolator_1: Speed-up**			Data type: UNSIGNED_8 Index: 23622_d = $5C46_h$
FB L_Interpolator_1: Limitation of the number of correction increments per cycle (speed-up)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		100	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00954

Parameter | Name: **C00954 | L_Interpolator_1: Synchronisation mode**		Data type: UNSIGNED_8 Index: 23621_d = $5C45_h$
Selection list(Lenze setting printed in bold)		**Info**
0	**No synchronisation**	
1	CAN PDO 1	
2	CAN PDO 2	
3	CAN PDO 3	
4	CAN PDO 4	
5	CAN Sync	
10	MCI Sync	
20	Input bStart	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00959

Parameter | Name: **C00959 | L_Curve: Current output value**			Data type: INTEGER_16 Index: 23616_d = $5C40_h$
From version 02.00.00 FB L_Curve_1: Display of the current output value *nOut_a*			
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Info**
C00959/1			L_Curve_1: Current output value
C00959/2			L_Curve_2: Current output value
C00959/3			L_Curve_3: Current output value
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00960

Parameter | Name: **C00960 | L_Curve_1: Selected curve type**		Data type: UNSIGNED_8 Index: 23615_d = $5C3F_h$
FB L_Curve_1: Selected curve type		
Selection list		
0	Out = 0	
1	Out = In	
2	Out = f(In)	
3	Out = f(table)	
Subcodes	**Lenze setting**	**Info**
C00960/1	1: Out = In	L_Curve_1: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C00961

Parameter | Name: **C00961 | L_Curve_1: Input limitation**			Data type: INTEGER_16 Index: 23614_d = $5C3E_h$
FB L_Curve_1: Upper and lower limit for input value			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C00961/1	199.99 %		L_Curve_1: Max. input
C00961/2	-199.99 %		L_Curve_1: Min. input
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00963

Parameter | Name: **C00963 | L_Curve_1: Table X-values**			Data type: INTEGER_16 Index: 23612_d = $5C3C_h$
FB L_Curve_1: X-values for characteristic function			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C00963/1	0		X values 1 ... 32 for characteristic function
C00963/...			
C00963/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00964

Parameter | Name: **C00964 | L_Curve_1: Table Y-values**			Data type: INTEGER_16 Index: 23611_d = $5C3B_h$
FB L_Curve_1: Y-value for characteristic function			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C00964/1	0		Y values 1 ... 32 for characteristic function
C00964/...			
C00964/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00965

Parameter | Name: **C00965 | Max. motor speed**			Data type: UNSIGNED_16 Index: 23610_d = $5C3A_h$
When the drive reaches the motor speed set here: • The "Fault" error response takes place, i.e. the motor is shut down immediately. • The error message "oS2: Max. motor speed reached" is entered into the logbook.			
Setting range (min. value | unit | max. value)			**Lenze setting**
50	rpm	65000	**60000 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00966

Parameter | Name: **C00966 | VFC: Time const. slip comp.**			Data type: UNSIGNED_16 Index: 23609_d = $5C39_h$
Filter time constant of the slip compensation for V/f characteristic control (VFCplus) • The time constant of slip compensation serves to specify the dynamics of slip compensation for V/f characteristic control without feedback. • The lower the selected time constant, the higher the dynamic performance of the slip compensation.			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	ms	6000	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00967

Parameter | Name: **C00967 | VFC: Frequency interpol. point n**			Data type: INTEGER_16 Index: 23608_d = $5C38_h$
Selection of the interpolation points (frequency values) for the V/f characteristic control (VFCplus) with user-definable V/f characteristic (C00006 = "10")			
Setting range (min. value | unit | max. value)			
-2600.0	Hz	2600.0	
Subcodes	**Lenze setting**		**Info**
C00967/1	-50.0 Hz		VFC: Frequency interpol. point 1
C00967/2	-40.0 Hz		VFC: Frequency interpol. point 2
C00967/3	-30.0 Hz		VFC: Frequency interpol. point 3
C00967/4	-20.0 Hz		VFC: Frequency interpol. point 4
C00967/5	-10.0 Hz		VFC: Frequency interpol. point 5
C00967/6	0.0 Hz		VFC: Frequency interpol. point 6
C00967/7	10.0 Hz		VFC: Frequency interpol. point 7
C00967/8	20.0 Hz		VFC: Frequency interpol. point 8
C00967/9	30.0 Hz		VFC: Frequency interpol. point 9
C00967/10	40.0 Hz		VFC: Frequency interpol. point 10
C00967/11	50.0 Hz		VFC: Frequency interpol. point 11
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00968

Parameter | Name: **C00968 | VFC: Voltage interpol. point n**			Data type: UNSIGNED_16 Index: 23607_d = $5C37_h$
Selection of the interpolation points (voltage values) for the V/f characteristic control (VFCplus) with user-definable V/f characteristic (C00006 = "10")			
Setting range (min. value | unit | max. value)			
0.00	V	600.00	
Subcodes	**Lenze setting**		**Info**
C00968/1	400.00 V		VFC: Voltage interpol. point 1
C00968/2	320.00 V		VFC: Voltage interpol. point 2
C00968/3	240.00 V		VFC: Voltage interpol. point 3
C00968/4	160.00 V		VFC: Voltage interpol. point 4
C00968/5	80.00 V		VFC: Voltage interpol. point 5
C00968/6	0.00 V		VFC: Voltage interpol. point 6
C00968/7	80.00 V		VFC: Voltage interpol. point 7
C00968/8	160.00 V		VFC: Voltage interpol. point 8
C00968/9	240.00 V		VFC: Voltage interpol. point 9
C00968/10	320.00 V		VFC: Voltage interpol. point 10
C00968/11	400.00 V		VFC: Voltage interpol. point 11
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00969

Parameter | Name: **C00969 | Motorparameter**			Data type: UNSIGNED_8 Index: 23606_d = $5C36_h$
From version 12.00.00			
Display range (min. value | unit | max. value)			
1		255	
Subcodes			**Info**
C00969/1			Motor - number of pole pairs
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00970

Parameter | Name: **C00970 | Rated device voltage**			Data type: UNSIGNED_16 Index: 23605_d = $5C35_h$
Display range (min. value | unit | max. value)			
0	V	1000	
Subcodes			**Info**
C00970/1			Rated device voltage
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00971

Parameter | Name: **C00971 | VFC: Limitation V/f +encoder**			Data type: UNSIGNED_16 Index: 23604_d = $5C34_h$
Limitation of the output frequency of the slip regulator and limitation of the injected stator frequency for the V/f control (VFCplus+encoder)			
Setting range (min. value | unit | max. value)			
0.00	Hz	100.00	
Subcodes	**Lenze setting**		**Info**
C00971/1	10.00 Hz		Maximum output / correcting variable of the slip regulator • The slip regulator output is limited to the value set here in motor/generator mode. • We recommend defining a limit value of one or two times the motor slip frequency.
C00971/2	100.00 Hz		Maximum frequency deviation between the rotational frequency (speed) measured mechanically by the encoder and the injected stator frequency. • A limitation may e.g. avoid overcurrent interruption when traversing to a fixed limit stop.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00972

Parameter | Name: **C00972 | VFC: Vp V/f +encoder**			Data type: UNSIGNED_16 Index: 23603_d = $5C33_h$
Proportional gain of the slip regulator for V/f control (VFCplus+encoder) • The gain must be selected depending on the drive system and the sensor resolution (range: 0.005 ... 5). • A high gain requires a high number of increments.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	Hz/Hz	64.000	**0.100 Hz/Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1000			

C00973

Parameter | Name: **C00973 | VFC: Ti V/f +encoder**			Data type: UNSIGNED_16 Index: 23602_d = $5C32_h$
Integral time constant of the slip regulator for V/f control (VFCplus+encoder) • In general, the time constant should be selected in a range of 20 ms (high dynamics) to 200 (low dynamics).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	ms	6000.0	**100.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00975

Parameter | Name: **C00975 | VFC-ECO: Vp CosPhi controller**			Data type: UNSIGNED_16 Index: 23600_d = $5C30_h$
Proportional gain of the Cos-Phi controller for energy-saving V/f characteristic control (VFCplusEco)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	Hz/Hz	64.000	**0.500 Hz/Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1000			

C00976

Parameter | Name: **C00976 | VFC-ECO: Ti CosPhi controller**			Data type: UNSIGNED_16 Index: 23599_d = $5C2F_h$
Reset time of the Cos-Phi controller for energy-saving V/f characteristic control (VFCplusEco)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	ms	6000.0	**200.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00977

Parameter | Name: **C00977 | VFC-ECO: Minimum voltage V/f**			Data type: INTEGER_16 Index: 23598_d = $5C2E_h$
Minimum voltage V/f of the Cos-Phi controller for energy-saving V/f characteristic control (VFCplusEco)			
Setting range (min. value | unit | max. value)			**Lenze setting**
20.00	%	100.00	**20.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00978

Parameter | Name: **C00978 | VFC-ECO: Voltage reduction**			Data type: INTEGER_16 Index: 23597_d = $5C2D_h$
Display of the voltage reduction with energy-saving V/f characteristic control (VFCplusEco)			
Display range (min. value | unit | max. value)			
-1000	V	1000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00979

Parameter | Name: **C00979 | Cosine phi**			Data type: INTEGER_16 Index: 23596_d = $5C2C_h$
Display of the cosφ setpoint and actual value with energy-saving V/f characteristic control (VFCplusEco)			
Display range (min. value | unit | max. value)			
-1.00		1.00	
Subcodes			**Info**
C00979/1			Cosine phi act
C00979/2			Cosine phi set
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00980

Parameter | Name: **C00980 | Performance indication**			Data type: INTEGER_32 Index: 23595_d = $5C2B_h$
Display parameter for an energy analysis in the prevailing application. From this, decisions can be deduced whether a measure for energy optimisation is economic.			
Display range (min. value | unit | max. value)			
-180.000	kW	180.000	
Subcodes			**Info**
C00980/1			Active output power
C00980/2			Apparent output power
C00980/3			Rated device power • From version 17.00.00 onwards
C00980/4			Input power • From version 17.00.00 onwards
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C00981

Parameter | Name: **C00981 | Energy display**			Data type: INTEGER_32 Index: 23594_d = $5C2A_h$
Display parameter for an energy analysis in the prevailing application. From this, decisions can be deduced whether a measure for energy optimisation is economic. • The values are saved to the device by switching off the mains and cannot be reset.			
Display range (min. value | unit | max. value)			
0.00	kWh	21474836.47	
Subcodes			**Info**
C00981/1			Output energy in motor mode
C00981/2			Output energy in generator mode
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00982

Parameter | Name: **C00982 | VFC-ECO: Voltage reduction ramp**			Data type: UNSIGNED_8 Index: 23593_d = $5C29_h$
Voltage ramp for cancelling V-Sub with energy-saving V/f characteristic control (VFCplusEco)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	s	5.0	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00983

Parameter | Name: **C00983 | Delay**			Data type: UNSIGNED_16 Index: 23592_d = $5C28_h$
Setting range (min. value | unit | max. value)			
0	ms	4000	
Subcodes	**Lenze setting**		**Info**
C00983/1	50 ms		Delay Imax
C00983/2	1000 ms		VFC: Voltage ramp after overvoltage oU • From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C00985

Parameter | Name: **C00985 | SLVC: Gain of field current controller**			Data type: INTEGER_16 Index: 23590_d = $5C26_h$
Gain of the direct-axis current difference (Id) between setpoint and actual current for the voltage model of the sensorless vector control (SLVC) • The gain should be selected within the range 0 ...1 %.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	20.00	**0.50 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00986

Parameter | Name: **C00986 | SLVC: Gain of cross current controller**			Data type: INTEGER_16 Index: 23589_d = $5C25_h$
Gain of the IQ difference for the voltage model of the sensorless vector control (SLVC)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	20.00	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00987

Parameter | Name: **C00987 | Inverter motor brake: nAdd**			Data type: INTEGER_16 Index: 23588_d = $5C24_h$
Speed lift which is connected in pulses to the brake ramp when the motor is braked. ▸ Inverter motor brake			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	1000	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00988

Parameter | Name: **C00988 | Inverter motor brake: PT1 filter time**			Data type: INTEGER_16 Index: 23587_d = $5C23_h$
PT1 filter time for smoothing the speed lift which is added in pulses (C00987) ▸ Inverter motor brake			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	ms	100.0	**0.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C00989

Parameter | Name: **C00989 | Restart on the fly: Flying restart frequency fd_add**			Data type: INTEGER_16 Index: 23586_d = $5C22_h$
From version 15.00.00			▸ Flying restart fct.
Setting range (min. value | unit | max. value)			
0.00	Hz	5.00	
Subcodes	**Lenze setting**		**Info**
C00989/1	0.00 Hz		Restart on the fly: Flying restart frequency fd_add
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00990

Parameter | Name: **C00990 | Flying restart fct.: Activation**		Data type: UNSIGNED_8 Index: 23585_d = $5C21_h$
Switch on/activate flying restart circuit for non-feedback drive systems		▸ Flying restart fct.
Selection list(Lenze setting printed in bold)		
0	**Off**	
1	On	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00991

Parameter | Name: **C00991 | Flying restart fct.: Process**		Data type: UNSIGNED_16 Index: 23584_d = $5C20_h$
Selection of the starting value and the speed search range for the flying restart function		▸ Flying restart fct.
Selection list(Lenze setting printed in bold)		**Info**
0	0...+n | Start: +10 Hz	Search positive speed range (0 ... +n) with a start frequency of +10 Hz
1	-n...0 | Start: -10 Hz	Search negative speed range (-n ... 0) with a start frequency of -10 Hz
2	**-n...+n | Start: +10 Hz**	Search negative and positive speed range (-n ... n) with a start frequency of +10 Hz
3	-n...+n | Start: -10 Hz	Search negative and positive speed range (-n ... n) with a start frequency of -10 Hz
4	-n...+n | Start: Cx992	Search the negative and positive speed range (-n ... n) with the start frequency set in C00992
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C00992

Parameter | Name: **C00992 | Flying restart fct.: Start frequency**			Data type: INTEGER_16 Index: 23583_d = $5C1F_h$
Manual selection of the starting value for the flying restart function • Only active if C00991 = 4			▸ Flying restart fct.
Setting range (min. value | unit | max. value)			**Lenze setting**
-200	Hz	200	**10 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C00993

Parameter | Name: **C00993 | Flying restart fct: Int. time**			Data type: UNSIGNED_16 Index: 23582_d = $5C1E_h$
Time constant of the angular difference controller of the flying restart function • The time constant is to amount between 60 ... 300 ms. ▸ Flying restart fct.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	ms	6000.0	power-related (see table)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00994

Parameter | Name: **C00994 | Flying restart fct.: Current**			Data type: INTEGER_16 Index: 23581_d = $5C1D_h$
Current to be injected during the flying restart process • 100 % ≡ rated motor current (C00088). • The flying restart current should amount to 10 ... 25 % of the rated motor current. ▸ Flying restart fct.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**25.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00995

Parameter | Name: **C00995 | SLPSM: Controlled current setpoint**			Data type: UNSIGNED_16 Index: 23580_d = $5C1C_h$
▸ Sensorless control for synchronous motors (SLPSM)			
Setting range (min. value | unit | max. value)			
5.00	%	400.00	
Subcodes	**Lenze setting**		**Info**
C00995/1	100.00 %		SLPSM: Controlled accelerating current
C00995/2	20.00 %		SLPSM: Controlled standstill current
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C00996

Parameter | Name: **C00996 | SLPSM: Switching speed**			Data type: INTEGER_16 Index: 23579_d = $5C1B_h$
▸ Sensorless control for synchronous motors (SLPSM)			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C00996/1	13.00 %		SLPSM: Switching speed, closed-loop control
C00996/2	8.00 %		SLPSM: Switching speed, open-loop control
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00997

Parameter | Name: **C00997 | SLPSM: Filter cutoff frequency**			Data type: INTEGER_16 Index: 23578_d = $5C1A_h$
			▸ Sensorless control for synchronous motors (SLPSM)
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**5.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C00998

Parameter | Name: **C00998 | SLPSM: Filter time rotor position**			Data type: INTEGER_16 Index: 23577_d = $5C19_h$
			▸ Sensorless control for synchronous motors (SLPSM)
Setting range (min. value | unit | max. value)			
0.5	ms	20.0	
Subcodes	**Lenze setting**		**Info**
C00998/1	3.0 ms		SLPSM: Filter time rotor position
C00998/2	5.0 ms		SLPSM: Filter time actual speed value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C00999

Parameter | Name: **C00999 | SLPSM: PLL gain**			Data type: INTEGER_16 Index: 23576_d = $5C18_h$
			▸ Sensorless control for synchronous motors (SLPSM)
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	1000	**400 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C01000

Parameter | Name: **C01000 | MCTRL: Status**			Data type: UNSIGNED_16 Index: 23575_d = $5C17_h$
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	SLPSM: Speed closed-loop control		1 ≡ The sensorless control for synchronous motors is in the closed-loop controlled operation ($\|n_{Setpoint}\|$ > n_{C00996}). ▶ Sensorless control for synchronous motors (SLPSM)
Bit 1	SLPSM: Speed open-loop control		From version 12.00.00 1 ≡ The sensorless control for synchronous motors is in the open-loop controlled operation ($\|n_{Setpoint}\|$ > n_{C00996}). ▶ Sensorless control for synchronous motors (SLPSM)
Bit 2	Motor ASM		From version 12.00.00 1 ≡ Set motor type is "Asynchronous motor"
Bit 3	Motor PSM		From version 12.00.00 1 ≡ Set motor type is "Synchronous motor"
Bit 4	Encoder open circuit: V/f linear active		From version 15.00.00 1 ≡ Due to an encoder open circuit, it has been internally switched to the encoderless V/f characteristic control to avoid impermissible motor movements.
Bit 5	Speed limit. C909/C910/fdmax		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C01001

Parameter | Name: **C01001 | Manual entry of motor type**		Data type: UNSIGNED_8 Index: 23574_d = $5C16_h$
From version 14.00.00		
Selection list		
0	Automatic	
1	ASM	
2	PSM	
Subcodes	**Lenze setting**	**Info**
C01001/1	0: Automatic	Manual entry of motor type
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C01004

Parameter | Name: **C01004 | Device command: Configuration**			Data type: UNSIGNED_16 Index: 23571_d = $5C13_h$
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Communication module		
Bit 1	CAN OnBoard		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C01004/1	0x0000		Load Lenze setting without:
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C01010

Parameter | Name: **C01010 | L_ArithmetikPhi 1-3: Function**		Data type: UNSIGNED_8 Index: 23565_d = $5C0D_h$
Selection of the internal arithmetics		
Selection list		
0	dnIn1_p	
1	dnIn1_p + dnIn2_p	
2	dnIn1_p - dnIn2_p	
3	dnIn1_p * dnIn2_p	
4	dnIn1_p / dnIn2_p	
Subcodes	**Lenze setting**	**Info**
C01010/1	0: dnIn1_p	L_ArithmetikPhi_1: Function
C01010/2	0: dnIn1_p	L_ArithmetikPhi_2: Function
C01010/3	0: dnIn1_p	L_ArithmetikPhi_3: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01011

Parameter | Name: **C01011 | L_ArithmetikPhi 4-6: Function**		Data type: UNSIGNED_8 Index: 23564_d = $5C0C_h$
From version 12.00.00 Selection of the internal arithmetics		
Selection list		
0	dnIn1_p	
1	dnIn1_p + dnIn2_p	
2	dnIn1_p - dnIn2_p	
3	dnIn1_p * dnIn2_p	
4	dnIn1_p / dnIn2_p	
5	dnIn1_p % dnIn2_p	
6	dnIn1_p + dnIn2_p (no limit.)	
7	dnIn1_p - dnIn2_p (no limit.)	
Subcodes	**Lenze setting**	**Information**
C01011/1	0: dnIn1_p	L_ArithmetikPhi_4: Function
C01011/2	0: dnIn1_p	L_ArithmetikPhi_5: Function
C01011/3	0: dnIn1_p	L_ArithmetikPhi_6: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01012

Parameter | Name: **C01012 | L_PhiDiv: byDivision**			Data type: INTEGER_8 Index: 23563_d = $5C0B_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
-31		31	
Subcodes	**Lenze setting**		**Information**
C01012/1	0		L_PhiDiv_1: byDivision
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01020

Parameter | Name: **C01020 | L_Odometer_1: Memory length**		Data type: UNSIGNED_8 Index: 23555_d = $5C03_h$
FB L_Odometer_1: No. of measurements		
Selection list (Lenze setting printed in bold)		
1	1 measurement	
2	2 measurements	
3	3 measurements	
4	4 measurements	
5	5 measurements	
6	6 measurements	
7	**7 measurements**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01021

Parameter | Name: C01021 | L_Odometer_1: Memory type		Data type: UNSIGNED_8 Index: 23554_d = $5C02_h$
FB L_Odometer_1: If "ring buffer" is selected, it is started again after the measurements set in C01020 and the old values are overwritten. Otherwise, the measurement stops.		
Selection list(Lenze setting printed in bold)		
0	**No ring buffer**	
1	Ring buffer	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01022

Parameter | Name: C01022 | L_Odometer_1: Input selection		Data type: UNSIGNED_8 Index: 23553_d = $5C01_h$
FB L_Odometer_1: Selection of position or speed input		
Selection list(Lenze setting printed in bold)		
0	**Pos input**	
1	V input	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01023

Parameter | Name: C01023 | L_Odometer_1: Edge selection		Data type: UNSIGNED_8 Index: 23552_d = $5C00_h$
FB L_Odometer_1: No. of the edge triggering the measurement		
Selection list(Lenze setting printed in bold)		
0	**High edge**	
1	Low edge	
2	High and low edge	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01025

Parameter | Name: C01025 | L_Curve_2: Selected curve type		Data type: UNSIGNED_8 Index: 23550_d = $5BFE_h$
From version 02.00.00 FB L_Curve_2: Selected curve type		
Selection list		
0	Out = 0	
1	Out = In	
2	Out = f(In)	
3	Out = f(table)	
Subcodes	**Lenze setting**	**Info**
C01025/1	1: Out = In	L_Curve_2: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01026

Parameter | Name: **C01026 | L_Curve_2: Input limitation**			Data type: INTEGER_16 Index: 23549_d = $5BFD_h$
From version 02.00.00 FB L_Curve_2: Upper and lower limit for input value			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C01026/1	199.99 %		L_Curve_2: Max. input
C01026/2	-199.99 %		L_Curve_2: Min. input
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01028

Parameter | Name: **C01028 | L_Curve_2: Table X-values**			Data type: INTEGER_16 Index: 23547_d = $5BFB_h$
From version 02.00.00 FB L_Curve_2: X-values for characteristic function			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C01028/1	0		X values 1 ... 32 for characteristic function
C01028/...			
C01028/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01029

Parameter | Name: **C01029 | L_Curve_2: Table Y-values**			Data type: INTEGER_16 Index: 23546_d = $5BFA_h$
From version 02.00.00 FB L_Curve_2: Y-value for characteristic function			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C01029/1	0		Y values 1 ... 32 for characteristic function
C01029/...			
C01029/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01030

Parameter | Name: **C01030 | L_Curve_3: Selected curve type**			Data type: UNSIGNED_8 Index: 23545$_d$ = 5BF9$_h$
From version 02.00.00 FB L_Curve_3: Selected curve type			
Selection list		**Info**	
0	Out = 0		
1	Out = In		
2	Out = f(In)		
3	Out = f(table)		
4	Out = f(characteristic)	From version 12.00.00	
Subcodes	**Lenze setting**	**Info**	
C01030/1	1: Out = In	L_Curve_3: Function	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01031

Parameter | Name: **C01031 | L_Curve_3: Input limitation**			Data type: INTEGER_16 Index: 23544$_d$ = 5BF8$_h$
From version 02.00.00 FB L_Curve_3: Upper and lower limit for input value			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C01031/1	199.99 %		L_Curve_3: Max. input
C01031/2	-199.99 %		L_Curve_3: Min. input
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01033

Parameter | Name: **C01033 | L_Curve_3: Table X-values**			Data type: INTEGER_16 Index: 23542$_d$ = 5BF6$_h$
From version 02.00.00 FB L_Curve_3: X-values for characteristic function			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C01033/1	0		X values 1 ... 32 for characteristic function
C01033/...			
C01033/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01034

Parameter | Name: **C01034 | L_Curve_3: Table Y-values**			Data type: INTEGER_16 Index: 23541_d = $5BF5_h$
From version 02.00.00 FB L_Curve_3: Y-value for characteristic function			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C01034/1	0		Y values 1 ... 32 for characteristic function
C01034/...			
C01034/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01035

Parameter | Name: **C01035 | L_Curve_3: SelectCurve**		Data type: UNSIGNED_16 Index: 23540_d = $5BF4_h$
From version 12.00.00 FB L_Curve_3: Selection of the tensile force profile		
Selection list		**Info**
0	Linear tensile force profile	▶ Use of L_Curve_3 for tensile force characteristic
1	Linear torque profile	
2	Tensile force profile according to characteristic	
Subcodes	**Lenze setting**	**Info**
C01035/1	0: Linear tensile force profile	L_Curve_3: SelectCurve
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01040

Parameter | Name: **C01040 | L_SRFG_1..2 linear ramp time**			Data type: UNSIGNED_32 Index: 23535_d = $5BEF_h$
Symmetrical acceleration/deceleration time for the ramp function generator			
Setting range (min. value | unit | max. value)			
0.001	s	999.999	
Subcodes	**Lenze setting**		**Info**
C01040/1	100.000 s		L_SRFG_1: linear ramp time
C01040/2	100.000 s		L_SRFG_2: linear ramp time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01041

Parameter | Name: **C01041 | L_SRFG_1..2 S-ramp time**			Data type: UNSIGNED_32 Index: 23534_d = $5BEE_h$
S-ramp time for jerk-free acceleration			
Setting range (min. value | unit | max. value)			
0.001	s	50.000	
Subcodes	**Lenze setting**		**Info**
C01041/1	0.200 s		L_SRFG_1: S-ramp time
C01041/2	0.200 s		L_SRFG_2: S-ramp time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01042

Parameter | Name: **C01042 | L_SRFG_1..2 limitations of output values**			Data type: INTEGER_16 Index: 23533_d = $5BED_h$
Limitation of the output values			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C01042/1	100.00 %		L_SRFG_1: Pos. Limit
C01042/2	-100.00 %		L_SRFG_1: Neg. Limit
C01042/3	100.00 %		L_SRFG_2: Pos. Limit
C01042/4	-100.00 %		L_SRFG_2: Neg. Limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01045

Parameter | Name: **C01045 | L_ConvAP 1-3: Numerator/denominator**			Data type: INTEGER_16 Index: 23530_d = $5BEA_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C01045/1	1		L_ConvAP_1: Numerator
C01045/2	1		L_ConvAP_1: Denominator
C01045/3	1		L_ConvAP_2: Numerator
C01045/4	1		L_ConvAP_2: Denominator
C01045/5	1		L_ConvAP_3: Numerator
C01045/6	1		L_ConvAP_3: Denominator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01046

Parameter | Name: C01046 | L_ConvPA 1-3: byDivision			Data type: INTEGER_8 Index: 23529_d = $5BE9_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0		31	
Subcodes	**Lenze setting**		**Info**
C01046/1	1		L_ConvPA_1: Division factor
C01046/2	1		L_ConvPA_2: Division factor
C01046/3	1		L_ConvPA_3: Division factor
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01047

Parameter | Name: C01047 | L_GearComp_1: Offset			Data type: INTEGER_16 Index: 23528_d = $5BE8_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-16383		16383	
Subcodes	**Lenze setting**		**Info**
C01047/1	0		L_GearComp_1: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01048

Parameter | Name: C01048 | L_GearComp_1: Num_Denom			Data type: INTEGER_16 Index: 23527_d = $5BE7_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C01048/1	1		L_GearComp_1: Numerator
C01048/2	1		L_GearComp_1: Denominator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01049

Parameter | Name: C01049 | L_CalcDiameter_1: Status			Data type: INTEGER_16 Index: 23526_d = $5BE6_h$
From version 02.00.00			
Display range (min. value | unit | max. value)			
-10		30	
Subcodes			**Info**
C01049/1			L_CalcDiameter_1: Status
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01050

Parameter | Name: C01050 | L_CalcDiameter_1: Diameter recalculation			Data type: UNSIGNED_32 Index: 23525d = 5BE5h
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.001	Rev.	1000.000	
Subcodes	**Lenze setting**		**Info**
C01050/1	1.000 rev.		L_CalcDiameter_1: Diameter recalculation 0
C01050/2	0.100 rev.		L_CalcDiameter_1: Diameter recalculation 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01051

Parameter | Name: C01051 | L_CalcDiameter_1: Filter time constant			Data type: UNSIGNED_16 Index: 23524d = 5BE4h
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.010	s	3.000	
Subcodes	**Lenze setting**		**Info**
C01051/1	1.000 s		L_CalcDiameter_1: Filter time constant
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01052

Parameter | Name: C01052 | L_CalcDiameter_1: Web break monitoring			Data type: INTEGER_16 Index: 23523d = 5BE3h
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C01052/1	10.00 %		L_CalcDiameter_1: Permissible diameter change
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01053

Parameter | Name: C01053 | L_ProcessCtrl_1: Controller times			Data type: UNSIGNED_16 Index: 23522d = 5BE2h
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.000	s	30.000	
Subcodes	**Lenze setting**		**Info**
C01053/1	0.000 s		L_ProcessCtrl_1: Acceleration/deceleration time
C01053/2	0.000 s		L_ProcessCtrl_1: Filter time constant
C01053/3	0.000 s		L_ProcessCtrl_1: Rate time
C01053/4	1.000 s		L_ProcessCtrl_1: Reset time
C01053/5	0.000 s		L_ProcessCtrl_1: Rate action
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01054

Parameter | Name: **C01054 | L_ProcessCtrl_1: System deviation**			Data type: INTEGER_16 Index: 23521_d = $5BE1_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C01054/1	100.00 %		L_ProcessCtrl_1: Gain of system deviation
C01054/2	0.00 %		L_ProcessCtrl_1: Area of system deviation
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01055

Parameter | Name: **C01055 | L_ProcessCtrl_1: Correcting variable limitation**		Data type: UNSIGNED_8 Index: 23520_d = $5BE0_h$
From version 02.00.00		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C01055/1	0: FALSE	L_ProcessCtrl_1: Correcting variable limitation
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01056

Parameter | Name: **C01056 | L_ProcessCtrl_1: Controller gain**			Data type: UNSIGNED_16 Index: 23519_d = $5BDF_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.00		100.00	
Subcodes	**Lenze setting**		**Info**
C01056/1	0.10		L_ProcessCtrl_1: Controller gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01057

Parameter | Name: **C01057 | L_CalcDiameter_1: Current diameter**			Data type: UNSIGNED_32 Index: 23518_d = $5BDE_h$
From version 02.00.00			
Display range (min. value | unit | max. value)			
0.000	mm	10000.000	
Subcodes			**Info**
C01057/1			L_CalcDiameter_1: Current diameter
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01058

Parameter | Name: **C01058 | L_PosCtrlLin 1-2: Limit stop**		Data type: UNSIGNED_8 Index: 23517_d = $5BDD_h$
From version 02.00.00		
Selection list		
0	deactivated	
1	activated	
Subcodes	**Lenze setting**	**Info**
C01058/1	0: Deactivated	L_PosCtrlLin_1: Limit stop
C01058/2	0: Deactivated	L_PosCtrlLin_2: Limit stop
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01059

Parameter | Name: **C01059 | L_PosCtrlLin 1-2: Positioning behaviour**		Data type: UNSIGNED_8 Index: 23516_d = $5BDC_h$
From version 02.00.00		
Selection list		
0	dnOut_p = 0	
1	dnOut_p/nOut_v follow dnAct_p	
2	dnOut_p/nOut_v follow dnSet_p	
3	dnOut_p/nOut_v follow dnAct_p (without limitation)	
Subcodes	**Lenze setting**	**Info**
C01059/1	0: dnOut_p = 0	L_PosCtrlLin_1: Positioning behaviour
C01059/2	0: dnOut_p = 0	L_PosCtrlLin_2: Positioning behaviour
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01060

Parameter | Name: **C01060 | L_PosCtrlLin 1-2: Ramps**			Data type: INTEGER_32 Index: 23515_d = $5BDB_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.010	s	130.000	
Subcodes	**Lenze setting**		**Info**
C01060/1	1.000 s		L_PosCtrlLin_1: Deceleration of set position
C01060/2	1.000 s		L_PosCtrlLin_1: Acceleration ramp
C01060/3	1.000 s		L_PosCtrlLin_1: Deceleration ramp
C01060/4	1.000 s		L_PosCtrlLin_2: Deceleration of set position
C01060/5	1.000 s		L_PosCtrlLin_2: Acceleration ramp
C01060/6	1.000 s		L_PosCtrlLin_2: Deceleration ramp
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01061

Parameter | Name: **C01061 | L_PosCtrlLin 1-2: Traversing speeds**			Data type: INTEGER_16 Index: 23514_d = $5BDA_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-15000	rpm	15000	
Subcodes	**Lenze setting**		**Info**
C01061/1	199 rpm		L_PosCtrlLin_1: Forward motion
C01061/2	199 rpm		L_PosCtrlLin_1: Return motion
C01061/3	199 rpm		L_PosCtrlLin_2: Forward motion
C01061/4	199 rpm		L_PosCtrlLin_2: Return motion
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01062

Parameter | Name: **C01062 | L_SwitchPoint_1: Dead time**			Data type: UNSIGNED_16 Index: 23513_d = $5BD9_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C01062/1	0		L_SwitchPoint_1: Dead time 1
C01062/2	0		L_SwitchPoint_1: Dead time 2
C01062/3	0		L_SwitchPoint_1: Dead time 3
C01062/4	0		L_SwitchPoint_1: Dead time 4
C01062/5	0		L_SwitchPoint_1: dead time 5
C01062/6	0		L_SwitchPoint_1: Dead time 6
C01062/7	0		L_SwitchPoint_1: dead time 7
C01062/8	0		L_SwitchPoint_1: Dead time 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01063

Parameter | Name: **C01063 | L_SwitchPoint_1: Hysteresis**			Data type: UNSIGNED_16 Index: 23512_d = $5BD8_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0	Incr.	65535	
Subcodes	**Lenze setting**		**Info**
C01063/1	0 incr.		L_SwitchPoint_1: Hysteresis 1
C01063/2	0 incr.		L_SwitchPoint_1: Hysteresis 2
C01063/3	0 incr.		L_SwitchPoint_1: Hysteresis 3
C01063/4	0 incr.		L_SwitchPoint_1: Hysteresis 4
C01063/5	0 incr.		L_SwitchPoint_1: Hysteresis 5
C01063/6	0 incr.		L_SwitchPoint_1: hysteresis 6
C01063/7	0 incr.		L_SwitchPoint_1: hysteresis 7
C01063/8	0 incr.		L_SwitchPoint_1: Hysteresis 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01064

Parameter | Name: **C01064 | L_SwitchPoint_1: CenterMode**		Data type: UNSIGNED_8 Index: 23511_d = $5BD7_h$
From version 02.00.00		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C01064/1	0: FALSE	L_SwitchPoint_1: CenterMode 1
C01064/2	0: FALSE	L_SwitchPoint_1: CenterMode 2
C01064/3	0: FALSE	L_SwitchPoint_1: CenterMode 3
C01064/4	0: FALSE	L_SwitchPoint_1: CenterMode 4
C01064/5	0: FALSE	L_SwitchPoint_1: CenterMode 5
C01064/6	0: FALSE	L_SwitchPoint_1: CenterMode 6
C01064/7	0: FALSE	L_SwitchPoint_1: CenterMode 7
C01064/8	0: FALSE	L_SwitchPoint_1: CenterMode 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01065

Parameter | Name: **C01065 | L_SwitchPoint_1: Running time**			Data type: UNSIGNED_16 Index: 23510_d = $5BD6_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0	ms	60000	
Subcodes	**Lenze setting**		**Info**
C01065/1	0 ms		L_SwitchPoint_1: Running time 1
C01065/2	0 ms		L_SwitchPoint_1: Running time 2
C01065/3	0 ms		L_SwitchPoint_1: Running time 3
C01065/4	0 ms		L_SwitchPoint_1: Running time 4
C01065/5	0 ms		L_SwitchPoint_1: Running time 5
C01065/6	0 ms		L_SwitchPoint_1: Running time 6
C01065/7	0 ms		L_SwitchPoint_1: runtime 7
C01065/8	0 ms		L_SwitchPoint_1: runtime 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01066

Parameter | Name: **C01066 | L_SwitchPoint_1: Status**		Data type: INTEGER_16 Index: 23509_d = $5BD5_h$
From version 02.00.00		
Selection list		
0	OK	
10	FB not active	
100	Switching points not plausible	
Subcodes		**Info**
C01066/1		L_SwitchPoint_1: Status 1
C01066/2		L_SwitchPoint_1: Status 2
C01066/3		L_SwitchPoint_1: Status 3
C01066/4		L_SwitchPoint_1: Status 4
C01066/5		L_SwitchPoint_1: Status 5
C01066/6		L_SwitchPoint_1: Status 6
C01066/7		L_SwitchPoint_1: Status 7
C01066/8		L_SwitchPoint_1: Status 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01067

Parameter | Name: **C01067 | Inversion of gearbox stages**		Data type: UNSIGNED_8 Index: 23508_d = $5BD4_h$
From version 02.00.00		
Selection list		
0	Not inverted	
1	Inverted	
2	Automatically from MCK	
Subcodes	**Lenze setting**	**Info**
C01067/1	0: Not inverted	L_PhiIntegrator_1: Invert. gearbox nSet_v
C01067/2	0: Not inverted	L_DFSET_1: Invert. gearbox nSet_v
C01067/3	0: Not inverted	L_CalcDiameter_1: Invert. gearbox nMotorSpeedAct_v
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01068

Parameter | Name: **C01068 | L_SwitchPoint_1: Invert**			Data type: UNSIGNED_16 Index: 23507_d = $5BD3_h$
From version 12.00.00			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Invert Output1		Bit set = inversion active
Bit 1	Invert Output2		
Bit 2	Invert Output3		
Bit 3	Invert Output4		
Bit 4	Invert Output5		
Bit 5	Invert Output6		
Bit 6	Invert Output7		
Bit 7	Invert Output8		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C01068/1	0x0000		L_SwitchPoint_1: Inversion of outputs
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01069

Parameter | Name: **C01069 | L_DFSET_1: Ramp settings**			Data type: UNSIGNED_16 Index: 23506_d = $5BD2_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0	Incr./ms	32767	
Subcodes	**Lenze setting**		**Info**
C01069/1	10 incr./ms		L_DFSET_1: Ramp SpeedTrim (1/100) • From version 12.00.00 onwards, this setting has a resolution increased by the factor 100: A value of 32767 is internally evaluated with 327.67. Hence, the unit is [incr./100 ms].
C01069/2	100 incr./ms		L_DFSET_1: Ramp angle compensation
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01070

Parameter | Name: **C01070 | L_DFSET_1: Angular trimming**			Data type: INTEGER_32 Index: 23505_d = $5BD1_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-134217728	Incr.	134217728	
Subcodes	**Lenze setting**		**Info**
C01070/1	0 incr.		L_DFSET_1: Speed-dependent angle adjustment
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01071

Parameter | Name: **C01071 | L_DFSET_1: Following error limit**			Data type: UNSIGNED_32 Index: 23504_d = $5BD0_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
10	Incr.	2147483647	
Subcodes	**Lenze setting**		**Info**
C01071/1	32768 incr.		L_DFSET_1: Following error limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01072

Parameter | Name: **C01072 | L_DFSET_1: Multiplier - angular trimming**			Data type: INTEGER_16 Index: 23503_d = $5BCF_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-20000		20000	
Subcodes	**Lenze setting**		**Info**
C01072/1	1		L_DFSET_1: Multiplier - angular trimming
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01073

Parameter | Name: **C01073 | L_DFSET_1: Adjustment**			Data type: UNSIGNED_8 Index: 23502_d = $5BCE_h$
From version 02.00.00			
Setting range (min. hex value | max. hex value)			
0x00		0xFF	
Value is bit-coded:			**Info**
Bit 0	Setpoint angle without gearbox factor		0 ≡ Evaluation of the setpoint angle with gearbox factor 1 ≡ Evaluation of the setpoint angle without gearbox factor
Bit 1	Angle correction with polynomial		0 ≡ Angle correction without downstream polynomial 1 ≡ Angle correction with downstream polynomial
Bit 2	External angle correction		0 ≡ Internal angle correction 1 ≡ External angle correction • From version 13.00.00
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Subcodes	**Lenze setting**		**Info**
C01073/1	0x02		L_DFSET_1: Adjustment
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01074

Parameter | Name: **C01074 | L_DFSET_1: Zero pulse divider**			Data type: UNSIGNED_16 Index: 23501_d = $5BCD_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0		16384	
Subcodes	**Lenze setting**		**Info**
C01074/1	0		L_DFSET_1: Divider for actual value zero pulse
C01074/2	0		L_DFSET_1: Divider for setpoint zero pulse
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01075

Parameter | Name: **C01075 | L_DFSET_1: Synchronisation mode**			Data type: UNSIGNED_8 Index: 23500_d = $5BCC_h$
From version 02.00.00			
Selection list			
0	inactive		
1	Permanent sync. without enable (bZeroPulse)		
2	Permanent sync. with enable (bZeroPulse)		
10	1x sync. - Angular diff. shortest path setpoint		
11	1x Sync. - Angular diff. Cw		
12	1x Sync. - Angular diff. Ccw		
13	1x Sync. - Angular diff. shortest path act. value		
Subcodes	**Lenze setting**	**Info**	
C01075/1	0: Inactive	L_DFSET_1: Synchronisation mode	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01076

Parameter | Name: **C01076 | L_DFRFG_1: Times**			Data type: UNSIGNED_32 Index: 23499_d = $5BCB_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0.000	s	999.900	
Subcodes	**Lenze setting**	**Info**	
C01076/1	1.000 s	L_DFRFG_1: Acceleration and deceleration time	
C01076/2	0.000 s	L_DFRFG_1: Deceleration time for quick stop	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01077

Parameter | Name: **C01077 | L_DFRFG_1: Max. speed-up**			Data type: INTEGER_16 Index: 23498_d = $5BCA_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
1	rpm	15000	
Subcodes	**Lenze setting**	**Info**	
C01077/1	3000 rpm	L_DFRFG_1: Max. speed-up	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01078

Parameter | Name: **C01078 | L_DFRFG_1: Following error**			Data type: UNSIGNED_32 Index: 23497_d = $5BC9_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
10	Incr.	2000000000	
Subcodes	**Lenze setting**		**Info**
C01078/1	2000000000 incr.		L_DFRFG_1: Following error limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01079

Parameter | Name: **C01079 | L_DFRFG_1: Synchronisation window**			Data type: UNSIGNED_16 Index: 23496_d = $5BC8_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
0	Incr.	65535	
Subcodes	**Lenze setting**		**Info**
C01079/1	100 incr.		L_DFRFG_1: Synchronisation window (position)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01080

Parameter | Name: **C01080 | L_DFRFG_1: Offset**			Data type: INTEGER_32 Index: 23495_d = $5BC7_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
Subcodes	**Lenze setting**		**Info**
C01080/1	0 incr.		L_DFRFG_1: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01081

Parameter | Name: **C01081 | L_DFRFG_1: Sync. direction / TP function**		Data type: UNSIGNED_8 Index: 23494_d = $5BC6_h$
From version 02.00.00		
Selection list		
1	cw/ccw - without TP	
2	cw - without TP	
3	ccw - without TP	
4	cw/ccw - with TP	
5	cw - with TP	
6	ccw - with TP	
Subcodes	**Lenze setting**	**Info**
C01081/1	1: cw/ccw - without TP	L_DFRFG_1: Sync. direction / TP function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01082

Parameter | Name: **C01082 | LS_WriteParamList: Execute Mode**		Data type: UNSIGNED_8 Index: 23493_d = $5BC5_h$
Parameter change-over: Selection of the activation method		
Selection list(Lenze setting printed in bold)		**Info**
0	**by Execute**	The writing of the parameter list is activated by a FALSE/TRUE edge at the *bExecute* input.
1	by Input Select	The writing of the parameter list is carried out if a change is made at the select inputs and if the inverter is initialised.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01083

Parameter | Name: **C01083 | LS_WriteParamList: Error status**			Data type: UNSIGNED_16 Index: 23492_d = $5BC4_h$
Parameter change-over: Error status: • 0 = no error • 33803 | 0x840B = invalid data type (e.g. STRING) • 33804 | 0x840C = limit violation • 33806 | 0x840E = invalid code • 33813 | 0x8415 = no element of the selection list • 33815 | 0x8417 = writing of the parameter not permitted • 33816 | 0x8418 = writing of the parameter only permitted if controller is inhibited • 33829 | 0x8425 = invalid subcode • 33865 | 0x8449 = no parameter with subcodes			
Display range (min. value | unit | max. value)			
0		34000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01084

Parameter | Name: **C01084 | LS_WriteParamList: Error line**			Data type: UNSIGNED_8 Index: 23491_d = $5BC3_h$
Parameter change-over: Display of the number of list entry where the error occurred (in connection with the value set selected via *bSelectWriteValue_1* and *bSelectWriteValue_2*).			
Display range (min. value | unit | max. value)			
0		32	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01085

Parameter | Name: **C01085 | LS_WriteParamList: Index**			Data type: INTEGER_32 Index: 23490_d = $5BC2_h$
Parameter change-over: Parameter for entry 1 ... 32			
Setting range (min. value | unit | max. value)			
0.000		16000.000	
Subcodes	**Lenze setting**		**Info**
C01085/1 C01085/... C01085/32	0.000		Parameter for entries 1 ... 32 • Format: <code number>.<subcode number> • Examples: "12.000" = C00012; "26.001" = C00026/1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01086

Parameter | Name: **C01086 | LS_WriteParamList: WriteValue_1**			Data type: INTEGER_32 Index: 23489_d = $5BC1_h$
Parameter change-over: Parameter values - value set 1			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C01086/1	0		Parameter values - value set 1 • Parameter values for the parameters defined in C01085/1 ... 32.
C01086/...			
C01086/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01087

Parameter | Name: **C01087 | LS_WriteParamList: WriteValue_2**			Data type: INTEGER_32 Index: 23488_d = $5BC0_h$
Parameter change-over: Parameter values - value set 2			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C01087/1	0		Parameter values - value set 2 • Parameter values for the parameters defined in C01085/1 ... 32.
C01087/...			
C01087/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01088

Parameter | Name: **C01088 | LS_WriteParamList: WriteValue_3**			Data type: INTEGER_32 Index: 23487_d = $5BBF_h$
Parameter change-over: Parameter values - value set 3			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C01088/1	0		Parameter values - value set 3 • Parameter values for the parameters defined in C01085/1 ... 32.
C01088/...			
C01088/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01089

Parameter | Name: **C01089 | LS_WriteParamList: WriteValue_4**			Data type: INTEGER_32 Index: 23486_d = $5BBE_h$
Parameter change-over: Parameter values - value set 4			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C01089/1	0		Parameter values - value set 4 • Parameter values for the parameters defined in C01085/1 ... 32.
C01089/...			
C01089/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01090

Parameter | Name: **C01090 | LS_ParReadWrite 1-6: Index**			Data type: INTEGER_32 Index: 23485_d = $5BBD_h$
Parameter to be read or written. • Format: <code number>,<subcode number> • For a setting of "0,000", inputs *wParIndex* and *wParSubindex* are effective for addressing purposes instead.			
Setting range (min. value | unit | max. value)			
0.000		16000.000	
Subcodes	**Lenze setting**		**Info**
C01090/1	0.000		LS_ParReadWrite_1: Index
C01090/2	0.000		LS_ParReadWrite_2: Index
C01090/3	0.000		LS_ParReadWrite_3: Index
C01090/4	0.000		LS_ParReadWrite_4: Index
C01090/5	0.000		LS_ParReadWrite_5: Index
C01090/6	0.000		LS_ParReadWrite_6: Index
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01091

Parameter | Name: **C01091 | LS_ParReadWrite 1-6: Cycle time**		Data type: UNSIGNED_16 Index: 23484_d = $5BBC_h$
Time interval for cyclic reading/writing		
Selection list		
0	0 (by Execute)	
20	20 ms	
50	50 ms	
100	100 ms	
200	200 ms	
500	500 ms	
1000	1 s	
2000	2 s	
5000	5 s	
10000	10 s	
Subcodes	**Lenze setting**	**Info**
C01091/1	0: 0 (by Execute)	LS_ParReadWrite_1: Cycle time
C01091/2	0: 0 (by Execute)	LS_ParReadWrite_2: Cycle time
C01091/3	0: 0 (by Execute)	LS_ParReadWrite_3: Cycle time
C01091/4	0: 0 (by Execute)	LS_ParReadWrite_4: Cycle time
C01091/5	0: 0 (by Execute)	LS_ParReadWrite_5: Cycle time
C01091/6	0: 0 (by Execute)	LS_ParReadWrite_6: Cycle time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01092

Parameter | Name: **C01092 | LS_ParReadWrite 1-6: FailState**			Data type: UNSIGNED_16 Index: 23483_d = $5BBB_h$
Error status: • 0 = no error • 33803 | 0x840B = invalid data type (e.g. STRING) • 33804 | 0x840C = limit violation • 33806 | 0x840E = invalid code • 33813 | 0x8415 = no element of the selection list • 33815 | 0x8417 = writing of the parameter not permitted • 33816 | 0x8418 = writing of the parameter only permitted if controller is inhibited • 33829 | 0x8425 = invalid subcode • 33865 | 0x8449 = no parameter with subcodes			
Display range **(min. value | unit | max. value)**			
0		34000	
Subcodes			**Info**
C01092/1			LS_ParReadWrite_1: Error status
C01092/2			LS_ParReadWrite_2: Error status
C01092/3			LS_ParReadWrite_3: Error status
C01092/4			LS_ParReadWrite_4: Error status
C01092/5			LS_ParReadWrite_5: Error status
C01092/6			LS_ParReadWrite_6: Error status
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01093

Parameter | Name: C01093 - LS_ParReadWrite 1-6: Arithmetic mode		Data type: UNSIGNED_8 Index: 23482_d = $5BBA_h$
The integrated arithmetic function allows for easy arithmetic conversion of the process value to be written or which was read into the format of the target parameter via parameterisable factors and without the need for an additional arithmetic FB.		
Selection list		
0	No arithmetic	
1	In16Bit: LW=+/-32767	
2	In16Bit: HW=+/-; LW=0..65535	
3	In32Bit: HW_LW=+/-2147483647	
Subcodes	**Lenze setting**	**Info**
C01093/1	0: no arithmetic	LS_ParReadWrite_1: Arithmetic mode
C01093/2	0: no arithmetic	LS_ParReadWrite_2: Arithmetic mode
C01093/3	0: no arithmetic	LS_ParReadWrite_3: Arithmetic mode
C01093/4	0: no arithmetic	LS_ParReadWrite_4: Arithmetic mode
C01093/5	0: no arithmetic	LS_ParReadWrite_5: Arithmetic mode
C01093/6	0: no arithmetic	LS_ParReadWrite_6: Arithmetic mode
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01094

Parameter | Name: C01094 - LS_ParReadWrite 1-6: Numerator			Data type: INTEGER_16 Index: 23481_d = $5BB9_h$
Arithmetic function: Factor (numerator) for internal conversion in arithmetic modes 1 ... 3.			
Setting range (min. value | unit | max. value)			
-32767		32767	
Subcodes	**Lenze setting**		**Info**
C01094/1	1		LS_ParReadWrite_1: Numerator
C01094/2	1		LS_ParReadWrite_2: Numerator
C01094/3	1		LS_ParReadWrite_3: Numerator
C01094/4	1		LS_ParReadWrite_4: Numerator
C01094/5	1		LS_ParReadWrite_5: Numerator
C01094/6	1		LS_ParReadWrite_6: Numerator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01095

Parameter | Name: **C01095 - LS_ParReadWrite 1-6: Denominator**			Data type: INTEGER_16 Index: 23480_d = $5BB8_h$
Arithmetic function: Factor (denominator) for internal conversion in arithmetic modes 1 ... 3.			
Setting range (min. value | unit | max. value)			
1		32767	
Subcodes	**Lenze setting**		**Info**
C01095/1	1		LS_ParReadWrite_1: Denominator
C01095/2	1		LS_ParReadWrite_2: Denominator
C01095/3	1		LS_ParReadWrite_3: Denominator
C01095/4	1		LS_ParReadWrite_4: Denominator
C01095/5	1		LS_ParReadWrite_5: Denominator
C01095/6	1		LS_ParReadWrite_6: Denominator
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01098

Parameter | Name: **C01098 | LS_ParReadWrite 1-6: Configuration**			Data type: UNSIGNED_16 Index: 23477_d = $5BB5_h$
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Keeping the value		
Bit 1	Keeping the value in the event of an error		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C01098/1	0x0000		LS_ParReadWrite_1: Configuration
C01098/2	0x0000		LS_ParReadWrite_2: Configuration
C01098/3	0x0000		LS_ParReadWrite_3: Configuration
C01098/4	0x0000		LS_ParReadWrite_4: Configuration
C01098/5	0x0000		LS_ParReadWrite_5: Configuration
C01098/6	0x0000		LS_ParReadWrite_6: Configuration
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01100

Parameter | Name: **C01100 | Function L_Counter 1-3**		Data type: UNSIGNED_8 Index: 23475_d = $5BB3_h$
Selection of reset function		
Selection list		
0	Normal counting	
1	Auto reset	
2	Manual reset	
Subcodes	**Lenze setting**	**Info**
C01100/1	0: Normal counting	L_Counter_1: Function
C01100/2	0: Normal counting	L_Counter_2: Function
C01100/3	0: Normal counting	L_Counter_3: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01101

Parameter | Name: **C01101 | Comparison L_Counter 1-3**		Data type: UNSIGNED_8 Index: 23474_d = $5BB2_h$
Selection of comparison operation		
Selection list		
0	Greater than or equal to	
1	Less than or equal to	
2	equal to	
Subcodes	**Lenze setting**	**Info**
C01101/1	0: Greater than or equal to	L_Counter_1: Comparison
C01101/2	0: Greater than or equal to	L_Counter_2: Comparison
C01101/3	0: Greater than or equal to	L_Counter_3: Comparison
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01108

Parameter | Name: **C01108 | L_SwitchPoint_1: Dead time factor**		Data type: UNSIGNED_8 Index: 23467_d = $5BAB_h$
From version 14.00.00		
Selection list		
1	1 µs	
2	10 µs	
3	100 µs	
4	1000 µs	
Subcodes	**Lenze setting**	**Info**
C01108/1	2: 10 µs	L_SwitchPoint_1: Deadtime factor 1
C01108/2	2: 10 µs	L_SwitchPoint_1: Deadtime factor 2
C01108/3	2: 10 µs	L_SwitchPoint_1: Deadtime factor 3
C01108/4	2: 10 µs	L_SwitchPoint_1: Deadtime factor 4
C01108/5	2: 10 µs	L_SwitchPoint_1: Deadtime factor 5
C01108/6	2: 10 µs	L_SwitchPoint_1: Deadtime factor 6
C01108/7	2: 10 µs	L_SwitchPoint_1: Deadtime factor 7
C01108/8	2: 10 µs	L_SwitchPoint_1: Deadtime factor 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01109

Parameter | Name: **C01109 | L_SwitchPointPar_1: Dead time factor**		Data type: UNSIGNED_8 Index: 23466_d = $5BAA_h$
Ab Version 14.00.00		
Selection list		
1	1 µs	
2	10 µs	
3	100 µs	
4	1000 µs	
Subcodes	**Lenze setting**	**Information**
C01109/1	2: 10 µs	L_SwitchPointPar_1: Dead time factor 1
C01109/2	2: 10 µs	L_SwitchPointPar_1: Dead time factor 2
C01109/3	2: 10 µs	L_SwitchPointPar_1: Dead time factor 3
C01109/4	2: 10 µs	L_SwitchPointPar_1: Dead time factor 4
C01109/5	2: 10 µs	L_SwitchPointPar_1: Dead time factor 5
C01109/6	2: 10 µs	L_SwitchPointPar_1: Dead time factor 6
C01109/7	2: 10 µs	L_SwitchPointPar_1: Dead time factor 7
C01109/8	2: 10 µs	L_SwitchPointPar_1: Dead time factor 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01110

Parameter | Name: **C01110 | LS_MultiEncoder: Solid measure**		Data type: UNSIGNED_8 Index: 23465_d = $5BA9_h$
From version 02.00.00 Configuration examples can be found in the following chapters: • SinCos absolute value encoder with HIPERFACE® protocol • SSI absolute value encoder with Stegmann-SSI protocol		
Selection list(Lenze setting printed in bold)		**Info**
0	**rotatively unipolar**	
1	linearly unipolar	
2	rotatively bipolar	
3	linearly bipolar	
4	Inverted rotatively in unipolar mode	From version 15.00.00
5	Inverted linearly in unipolar mode	
6	Inverted rotatively in bipolar mode	From version 15.00.00
7	Inverted linearly in bipolar mode	From version 15.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C01111

Parameter | Name: **C01111 | LS_MultiEncoder: Encoder constant**			Data type: INTEGER_32 Index: 23464_d = $5BA8_h$
From version 02.00.00 Configuration examples can be found in the following chapters: • SinCos absolute value encoder with HIPERFACE® protocol • SSI absolute value encoder with Stegmann-SSI protocol			
Setting range (min. value | unit | max. value)			
1		2147483647	
Subcodes	**Lenze setting**		**Information**
C01111/1	1		LS_MultiEncoder: Difference - traverse path
C01111/2	1		LS_MultiEncoder: Difference - encoder value
C01111/3	1		LS_MultiEncoder: Traverse path - encoder increment
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C01112

Parameter | Name: **C01112 | LS_MultiEncoder: Position values**			Data type: INTEGER_32 Index: 23463_d = $5BA7_h$
			▸ Encoder/feedback system: Multi-Encoder
Setting range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes	**Lenze setting**		**Information**
C01112/1	0.0000 units		LS_MultiEncoder: Position offset
C01112/2	-214748.3647 units		LS_MultiEncoder: Data area min
C01112/3	214748.3647 units		LS_MultiEncoder: Data area max
C01112/4	0.0000 units		LS_MultiEncoder: Max. position jump • Ab Version 14.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10000			

_ _

C01119

Parameter | Name: **C01119 | LS_MultiEncoder: Current position**			Data type: INTEGER_32 Index: 23456_d = $5BA0_h$
From version 02.00.00 ▶ Encoder/feedback system: Multi-Encoder			
Display range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes			**Information**
C01119/1			LS_MultiEncoder: Current position
C01119/2			LS_MultiEncoder: Maximum travel distance
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01120

Parameter | Name: **C01120 | Sync signal source**		Data type: UNSIGNED_8 Index: 23455_d = $5B9F_h$
Selection of the signal source for device synchronisation • The drive can only be synchronised by one source. ▶ Synchronisation of the internal time base		
Selection list(Lenze setting printed in bold)		**Info**
0	**Off**	Synchronisation off
1	CAN on board	Synchronisation via CAN bus ▶ Sync telegram
2	AxisBusIO	Synchronisation via axis bus ▶ Axis bus
4	MCI	Synchronisation via MCI (communication module)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01121

Parameter | Name: **C01121 | Sync cycle time setpoint**			Data type: UNSIGNED_16 Index: 23454_d = $5B9E_h$
Cycle time setpoint for device synchronisation • Time interval at which the phase control loop (PLL) in the inverter expects the synchronisation signals. • The cycle time setpoint must be set according to the cycle of the respective synchronisation source. **Note:** • Only integer multiples of 1000 µs can be set. • Intelligent communication modules usually define the cycle time setpoint derived from the bus cycle. In this case, a manual change is not possible. Example: For the CAN bus, a distance of 2 ms has been set between two synchronisation signals. If the CAN bus is to be used as synchronisation source, a synchronisation cycle of 2000 µs must be set in C01121. ▶ Synchronisation of the internal time base			
Setting range (min. value | unit | max. value)			**Lenze setting**
1000	µs	20000	**1000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01122

Parameter | Name: **C01122 | Sync phase position**			Data type: UNSIGNED_16 Index: 23453_d = $5B9D_h$
Phase position for device synchronisation • The phase position determines the zero-time of the internal system cycle with regard to the synchronisation signal (bus cycle). Since PDO processing is an inherent part of the system part of the application, the instant of acceptance of the PDOs is postponed as well by a changed phase position. • With a setting = 0, the system cycle starts simultaneously with the synchronisation signal. • With a setting > 0, the internal system cycle starts earlier by the set time with regard to the synchronisation signal (the phase position acts negatively). • Intelligent communication modules define the optimal time with activated synchronisation by themselves. In this case, a manual change is not possible. • The decisive factor for defining C01122 is the time where all nodes are provided with valid PDOs. Example: If the phase position is set to 550 µs, the system part of the application starts 550 µs before the arrival of the synchronisation signal. ▶ Synchronisation of the internal time base			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	1000	**0 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01123

Parameter | Name: **C01123 | Sync window**			Data type: UNSIGNED_16 Index: 23452_d = $5B9C_h$
Time slot for monitoring the synchronisation signal or the phase position • The synchronisation signal or the current phase position must be within this time slot around the corresponding expected value (C01122). • With the setting "1000 µs" there will be no monitoring. ▶ Synchronisation of the internal time base			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	10000	**100 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01124

Parameter | Name: **C01124 | Sync correction width**		Data type: UNSIGNED_8 Index: 23451_d = $5B9B_h$
Correction increment for device synchronisation • If the cycle times of the synchronisation signal differs and phase-locken loop (PLL) differ from each other, this setting defines the measure the phase-locking loop is reset with. • If synchronisation is not reached, select a higher correction constant. • The optimum setting depends on quartz precision and must be determined empirically if required. ▶ Synchronisation of the internal time base		
Selection list(Lenze setting printed in bold)		
1	80ns	
2	160ns	
3	240ns	
4	**320ns**	
5	400ns	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01130

Parameter | Name: **C01130 | LS_RetainData: Selection**		Data type: UNSIGNED_16 Index: 23445_d = $5B95_h$
From version 02.00.00		
Setting range (min. hex value | max. hex value)		
0x0000	0xFFFF	
Value is bit-coded:		
Bit 0	wIn1	
Bit 1	wIn2	
Bit 2	wIn3	
Bit 3	wIn4	
Bit 4	dnIn1	
Bit 5	dnIn2	
Bit 6	dnIn3	
Bit 7	dnIn4	
Bit 8	bIn1	
Bit 9	bIn2	
Bit 10	bIn3	
Bit 11	bIn4	
Bit 12	Reserved	
Bit 13	Reserved	
Bit 14	Reserved	
Bit 15	Reserved	
Subcodes	**Lenze setting**	**Info**
C01130/1	0x000F	LS_RetainData: Selection bSetRetain_1
C01130/2	0x00F0	LS_RetainData: Selection bSetRetain_2
C01130/3	0x0F00	LS_RetainData: Selection bSetRetain_3
C01130/4	0x0FFF	LS_RetainData: Selection bLoadParams
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT		

C01131

Parameter | Name: **C01131 | LS_RetainData: 16Bit data**		Data type: UNSIGNED_16 Index: 23444_d = $5B94_h$
From version 02.00.00		
Setting range (min. value | unit | max. value)		
0	65535	
Subcodes	**Lenze setting**	**Info**
C01131/1	0	LS_RetainData: wInitValue1
C01131/2	0	LS_RetainData: wInitValue2
C01131/3	0	LS_RetainData: wInitValue3
C01131/4	0	LS_RetainData: wInitValue4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01132

Parameter | Name: **C01132 | LS_RetainData: 32Bit data**			Data type: INTEGER_32 Index: 23443_d = $5B93_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C01132/1	0		LS_RetainData: dnInitValue1
C01132/2	0		LS_RetainData: dnInitValue2
C01132/3	0		LS_RetainData: dnInitValue3
C01132/4	0		LS_RetainData: dnInitValue4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01133

Parameter | Name: **C01133 | LS_RetainData: Bool data**		Data type: UNSIGNED_8 Index: 23442_d = $5B92_h$
From version 02.00.00		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C01133/1	0: FALSE	LS_RetainData: bInitValue1
C01133/2	0: FALSE	LS_RetainData: bInitValue2
C01133/3	0: FALSE	LS_RetainData: bInitValue3
C01133/4	0: FALSE	LS_RetainData: bInitValue4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01134

Parameter | Name: **C01134 | LS_RetainData: 16bit data**			Data type: UNSIGNED_16 Index: 23441_d = $5B91_h$
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C01134/1	0		LS_RetainData: wValue1
C01134/2	0		LS_RetainData: wValue2
C01134/3	0		LS_RetainData: wValue3
C01134/4	0		LS_RetainData: wValue4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01135

Parameter | Name: **C01135 | LS_RetainData: 32bit data**			Data type: INTEGER_32 Index: 23440$_d$ = 5B90$_h$
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Info**
C01135/1	0		LS_RetainData: dnValue1
C01135/2	0		LS_RetainData: dnValue2
C01135/3	0		LS_RetainData: dnValue3
C01135/4	0		LS_RetainData: dnValue4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01136

Parameter | Name: **C01136 | LS_RetainData: Bool data**		Data type: UNSIGNED_8 Index: 23439$_d$ = 5B8F$_h$
Selection list		**Info**
0	False	
1	True	
Subcodes	**Lenze setting**	**Info**
C01136/1	0: FALSE	LS_RetainData: bValue1
C01136/2	0: FALSE	LS_RetainData: bValue2
C01136/3	0: FALSE	LS_RetainData: bValue3
C01136/4	0: FALSE	LS_RetainData: bValue4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01138

Parameter | Name: **C01138 | L_Transient 1-4: Function**		Data type: UNSIGNED_8 Index: 23437$_d$ = 5B8D$_h$
Selection of edge evaluation		
Selection list		
0	High edge	
1	Low edge	
2	High and low edge	
Subcodes	**Lenze setting**	**Info**
C01138/1	0: High edge	L_Transient_1: Function
C01138/2	0: High edge	L_Transient_2: Function
C01138/3	0: High edge	L_Transient_3: Function
C01138/4	0: High edge	L_Transient_4: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01139

Parameter | Name: **C01139 | L_Transient 1-4: Pulse duration**			Data type: UNSIGNED_16 Index: 23436_d = $5B8C_h$
Setting range (min. value | unit | max. value)			
0.000	s	60.000	
Subcodes	**Lenze setting**		**Info**
C01139/1	0.000 s		L_Transient_1: Pulse duration
C01139/2	0.000 s		L_Transient_2: Pulse duration
C01139/3	0.000 s		L_Transient_3: Pulse duration
C01139/4	0.000 s		L_Transient_4: Pulse duration
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01140

Parameter | Name: **C01140 | L_Transient 5-8: Function**		Data type: UNSIGNED_8 Index: 23435_d = $5B8B_h$
Selection of edge evaluation		
Selection list		
0	High edge	
1	Low edge	
2	High and low edge	
Subcodes	**Lenze setting**	**Info**
C01140/1	0: High edge	L_Transient_5: Function
C01140/2	0: High edge	L_Transient_6: Function
C01140/3	0: High edge	L_Transient_7: Function
C01140/4	0: High edge	L_Transient_8: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01141

Parameter | Name: **C01141 | L_Transient 5-8 pulse duration**			Data type: UNSIGNED_16 Index: 23434_d = $5B8A_h$
Setting range (min. value | unit | max. value)			
0.000	s	60.000	
Subcodes	**Lenze setting**		**Info**
C01141/1	0.000 s		L_Transient_5: Pulse duration
C01141/2	0.000 s		L_Transient_6: Pulse duration
C01141/3	0.000 s		L_Transient_7: Pulse duration
C01141/4	0.000 s		L_Transient_8: Pulse duration
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01150

Parameter | Name: **C01150 | L_PhaseIntK: Function**		Data type: UNSIGNED_8 Index: 23425_d = $5B81_h$
Loading behaviour of the integrator		
Selection list		**Info**
0	Loading with level	Load integrator with TRUE level at the input *bLoad*
1	Loading with edge	Load integrator with FALSE/TRUE edge at the input *bLoad*
2	Loading with level + reset	Load integrator when the comparison value (C01151) is reached or with TRUE level at the *bLoad* input
Subcodes	**Lenze setting**	**Info**
C01150/1	0: Loading with level	L_PhaseIntK_1: Function
C01150/2	0: Loading with level	L_PhaseIntK_2: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01151

Parameter | Name: **C01151 | L_PhaseIntK: Compare**			Data type: INTEGER_32 Index: 23424_d = $5B80_h$
Comparison value			
Setting range (min. value | unit | max. value)			
0		2147418112	
Subcodes	**Lenze setting**		**Info**
C01151/1	0		L_PhaseIntK_1: Comparison value
C01151/2	0		L_PhaseIntK_2: Comparison value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01152

Parameter | Name: **C01152 | L_SwitchPointPar_1: Dead time**			Data type: UNSIGNED_16 Index: 23423_d = $5B7F_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Information**
C01152/1	0		L_SwitchPointPar_1: Dead time 1
C01152/2	0		L_SwitchPointPar_1: Dead time 2
C01152/3	0		L_SwitchPointPar_1: Dead time 3
C01152/4	0		L_SwitchPointPar_1: Dead time 4
C01152/5	0		L_SwitchPointPar_1: Dead time 5
C01152/6	0		L_SwitchPointPar_1: Dead time 6
C01152/7	0		L_SwitchPointPar_1: Dead time 7
C01152/8	0		L_SwitchPointPar_1: Dead time 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01153

Parameter | Name: **C01153 | L_SwitchPointPar_1: Hysteresis**			Data type: UNSIGNED_16 Index: 23422_d = $5B7E_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0	Incr.	65535	
Subcodes	**Lenze setting**		**Information**
C01153/1	0 incr.		L_SwitchPointPar_1: Hysteresis 1
C01153/2	0 incr.		L_SwitchPointPar_1: Hysteresis 2
C01153/3	0 incr.		L_SwitchPointPar_1: Hysteresis 3
C01153/4	0 incr.		L_SwitchPointPar_1: Hysteresis 4
C01153/5	0 incr.		L_SwitchPointPar_1: Hysteresis 5
C01153/6	0 incr.		L_SwitchPointPar_1: Hysteresis 6
C01153/7	0 incr.		L_SwitchPointPar_1: Hysteresis 7
C01153/8	0 incr.		L_SwitchPointPar_1: Hysteresis 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01154

Parameter | Name: **C01154 | L_SwitchPointPar_1: CenterMode**		Data type: UNSIGNED_8 Index: 23421_d = $5B7D_h$
From version 12.00.00		
Selection list		
0	False	
1	True	
Subcodes	**Lenze setting**	**Information**
C01154/1	0: FALSE	L_SwitchPointPar_1: CenterMode 1
C01154/2	0: FALSE	L_SwitchPointPar_1: CenterMode 2
C01154/3	0: FALSE	L_SwitchPointPar_1: CenterMode 3
C01154/4	0: FALSE	L_SwitchPointPar_1: CenterMode 4
C01154/5	0: FALSE	L_SwitchPointPar_1: CenterMode 5
C01154/6	0: FALSE	L_SwitchPointPar_1: CenterMode 6
C01154/7	0: FALSE	L_SwitchPointPar_1: CenterMode 7
C01154/8	0: FALSE	L_SwitchPointPar_1: CenterMode 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01155

Parameter | Name: **C01155 | L_SwitchPointPar_1: Running time**			Data type: UNSIGNED_16 Index: 23420_d = $5B7C_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0	ms	60000	
Subcodes	**Lenze setting**		**Information**
C01155/1	0 ms		L_SwitchPointPar_1: Running time 1
C01155/2	0 ms		L_SwitchPointPar_1: Running time 2
C01155/3	0 ms		L_SwitchPointPar_1: Running time 3
C01155/4	0 ms		L_SwitchPointPar_1: Running time 4
C01155/5	0 ms		L_SwitchPointPar_1: Running time 5
C01155/6	0 ms		L_SwitchPointPar_1: Running time 6
C01155/7	0 ms		L_SwitchPointPar_1: Running time 7
C01155/8	0 ms		L_SwitchPointPar_1: Running time 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01156

Parameter | Name: **C01156 | L_SwitchPointPar_1: Status**		Data type: INTEGER_16 Index: 23419_d = $5B7B_h$
From version 12.00.00		
Selection list		
0	OK	
10	FB not active	
100	Switching points not plausible	
Subcodes		**Information**
C01156/1		L_SwitchPointPar_1: Status 1
C01156/2		L_SwitchPointPar_1: Status 2
C01156/3		L_SwitchPointPar_1: Status 3
C01156/4		L_SwitchPointPar_1: Status 4
C01156/5		L_SwitchPointPar_1: Status 5
C01156/6		L_SwitchPointPar_1: Status 6
C01156/7		L_SwitchPointPar_1: Status 7
C01156/8		L_SwitchPointPar_1: Status 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01157

Parameter | Name: **C01157 | L_SwitchPointPar_1:Position**			Data type: INTEGER_32 Index: 23418_d = $5B7A_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes	**Lenze setting**		**Information**
C01157/1	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint1_p
C01157/2	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size1_p
C01157/3	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint2_p
C01157/4	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size2_p
C01157/5	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint3_p
C01157/6	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size3_p
C01157/7	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint4_p
C01157/8	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size4_p
C01157/9	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint5_p
C01157/10	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size5_p
C01157/11	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint6_p
C01157/12	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size6_p
C01157/13	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint7_p
C01157/14	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size7_p
C01157/15	0.0000 units		L_SwitchPointPar_1: dnSwitchPoint8_p
C01157/16	0.0000 units		L_SwitchPointPar_1: dn2ndPoint_Size8_p
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01158

Parameter | Name: **C01158 | L_SwitchPointPar_1: Invert**			Data type: UNSIGNED_16 Index: 23417_d = $5B79_h$
From version 12.00.00			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Invert Output1		Bit set = inversion active
Bit 1	Invert Output2		
Bit 2	Invert Output3		
Bit 3	Invert Output4		
Bit 4	Invert Output5		
Bit 5	Invert Output6		
Bit 6	Invert Output7		
Bit 7	Invert Output8		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Information**
C01158/1	0x0000		L_SwitchPointPar_1: Inversion of outputs
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01190

Parameter | Name: **C01190 | Motor thermal sensor**		Data type: UNSIGNED_8 Index: 23385_d = $5B59_h$
Selection of the thermal sensor for monitoring the motor temperature ▶ Encoder/feedback system: Motor temperature monitoring (KTY)		
Selection list		**Info**
0	KTY83-110	Lenze standard KTY83-110 (MDSKX, MCS06)
1	Spec. characteristic	Characteristic defined via C01191 and C01192
2	KTY83-110 + 2 x PTC	Lenze standard KTY83-110 + 2 x PTC 150°C (MCS09-MCS19)
4	KTY84-130	Lenze standard KTY83-130
5	PT1000	
6	PT1000 + 2 x PTC	
Subcodes	**Lenze setting**	**Information**
C01190/1	0: KTY83-110	Type of motor temperature sensor resolver
C01190/2	0: KTY83-110	Type of motor temperature sensor MultiEncoder
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C01191

Parameter | Name: **C01191 | PTC characteristic: Temperature 1/2**			Data type: UNSIGNED_8 Index: 23384_d = $5B58_h$
The spec. thermal sensor characteristic is selected through the setting C01190="1" ▸ Encoder/feedback system: Motor temperature monitoring (KTY)			
Setting range (min. value | unit | max. value)			
0	°C	255	
Subcodes	**Lenze setting**		**Information**
C01191/1	100 °C		PTC characteristic: Temperature 1
C01191/2	150 °C		PTC characteristic: Temperature 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C01192

Parameter | Name: **C01192 | PTC characteristic: Resistance 1/2**			Data type: INTEGER_32 Index: 23383_d = $5B57_h$
The spec. thermal sensor characteristic is selected through the setting C01190="1" ▸ Encoder/feedback system: Motor temperature monitoring (KTY)			
Setting range (min. value | unit | max. value)			
0	Ohm	30000	
Subcodes	**Lenze setting**		**Information**
C01192/1	1070 Ohms		PTC characteristic: Resistance 1
C01192/2	2225 ohms		PTC characteristic: Resistance 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C01193

Parameter | Name: **C01193 | Motor temp. feedback system**		Data type: UNSIGNED_8 Index: 23382_d = $5B56_h$
Selection of the feedback system for monitoring the motor temperature ▸ Encoder/feedback system: Motor temperature monitoring (KTY)		
Selection list(Lenze setting printed in bold)		
0	**Speed feedback**	
1	Resolver input	
2	Encoder input	
5	Res. and enc. parallel	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C01201

Parameter | Name: **C01201 | MCK: Cycle**			Data type: INTEGER_32 Index: 23374_d = $5B4E_h$
Cycle for Modulo measuring system • When the cycle is set to 0 units (Lenze setting), the traversing range is unlimited (classical measuring system). ▸ Activation of the Modulo measuring system			
Setting range (min. value | unit | max. value)			
0.0000	units	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01201/1	0.0000 units		MCK: Cycle
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01202

Parameter | Name: **C01202 | MCK: iM motor/process**			Data type: UNSIGNED_16 Index: 23373_d = $5B4D_h$
Gearbox factor - motor • Entry of the gearbox factor as numerator/denominator ratio (numerator = motor speed and denominator = output speed of gearbox) or from the number of teeth of the gearbox arrangement. Machine parameter			
Setting range (min. value | unit | max. value)			
1		65535	
Subcodes	**Lenze setting**		**Info**
C01202/1	1		MCK: iM: Numerator gearbox factor Z2
C01202/2	1		MCK: iM: Denominator gearbox factor Z1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01203

Parameter | Name: **C01203 | MCK: iG motor/position encoder**			Data type: UNSIGNED_16 Index: 23372_d = $5B4C_h$
Gearbox factor - position encoder • Entry of the gearbox factor as numerator/denominator ratio, with numerator = motor speed and denominator = position encoder speed. Machine parameter			
Setting range (min. value | unit | max. value)			
1		65535	
Subcodes	**Lenze setting**		**Info**
C01203/1	1		MCK: iG: Numerator (motor speed)
C01203/2	1		MCK: iG: Denominator (encoder speed)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01204

Parameter | Name: **C01204 | MCK: Feed constant**			Data type: INTEGER_32 Index: 23371_d = $5B4B_h$
The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft. • The value is entered in application units referred to one revolution. Machine parameter			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0001	units/rev.	214748.3647	**360.0000 units/rev.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01205

Parameter | Name: **C01205 | MCK: Position resolution**			Data type: INTEGER_32 Index: 23370_d = $5B4A_h$
Display of the number of increments corresponding to one unit. • 1 motor revolution ≡ 65536 increments Machine parameter			
Display range (min. value | unit | max. value)			
0.0000	Incr./unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01206

Parameter | Name: **C01206 | MCK: Mounting direction**		Data type: UNSIGNED_8 Index: 23369d = 5B49h
Inversion for mirrored motor and encoder mounting ▸ Machine parameter		
Selection list		
0	Not inverted	
1	Inverted	
Subcodes	**Lenze setting**	**Info**
C01206/1	0: Not inverted	Motor mounting direction • Setting for motor mounting turned by 180°.
C01206/2	0: Not inverted	Position encoder mounting direction • Setting for position encoder system mounting turned by 180°.
C01206/3	0: Not inverted	DFOUT: Setpoint direction
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C01210

Parameter | Name: **C01210 | MCK: Current positions**			Data type: INTEGER_32 Index: 23365d = 5B45h
Display of current position data of the Motion Control Kernel			
Display range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes			**Info**
C01210/1			MCK: Feed • Display of the current feed for positioning profiles as a relative distance.
C01210/2			MCK: Set position • Display of the current setpoint position calculated by the MCK.
C01210/3			MCK: Actual position • Display of the current actual position calculated by an optional encoder system.
C01210/4			MCK: Following error • Display of the current following error as a difference between setpoint position and actual position.
C01210/5			MCK: Positioning accuracy • Display of the current positioning accuracy referred to the actual number of increments of the position encoder.
C01210/6			MCK: Target position
C01210/7			MCK: Set position modulo
C01210/8			MCK: Actual position modulo • From version 12.00.00
C01210/9			MCK: Dist. ref. mark and Ref-TP • From version 14.00.00 • Display of the distance between pre-stop mark/limit switch and TP/MP while homing. ▸ Distance measurement
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01211

Parameter | Name: **C01211 | MCK: Speed**			Data type: INTEGER_32 Index: 23364_d = $5B44_h$
Display of the current speed data			
Display range (min. value | unit | max. value)			
-214748.3647	units/s	214748.3647	
Subcodes			**Info**
C01211/1			MCK: Max. traversing speed 100%_C11 • Display of the maximum traversing speed based on the reference speed set in C00011.
C01211/2			MCK: Set speed • From version 12.00.00
C01211/3			MCK: Actual speed • From version 12.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01213

Parameter | Name: **C01213 | MCK: Max. traversing distance**			Data type: INTEGER_32 Index: 23362_d = $5B42_h$
Display of current position limits			
Display range (min. value | unit | max. value)			
-2147480000	units	2147480000	
Subcodes			**Info**
C01213/1			MCK: Max. traversing distance • Display of the maximum distance to be traversed referred to the 32-bit display area (max. 2147483647 increments).
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01215

Parameter | Name: **C01215 | MCK: Following error**			Data type: INTEGER_32 Index: 23360_d = $5B40_h$
Limits for following error monitoring system ▶ Following error monitoring system			
Setting range (min. value | unit | max. value)			
0.0000	units	214748.0000	
Subcodes	**Lenze setting**		**Info**
C01215/1	0.0000 units		MCK: Following error limit 1 • First limit of the maximum following error for monitoring and response activation.
C01215/2	0.0000 units		MCK: Following error limit 2 • Second limit of the maximum following error for monitoring and response activation.
C01215/3	0.0000 units		MCK: Following error SetPos=ActPos • Ab Version 14.00.00 ▶ Monitoring of the following error in case of controller inhibit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01216

Parameter | Name: C01216 | MCK: Positioning setting			Data type: UNSIGNED_8 Index: 23359_d = $5B3F_h$
			▸ Positioning
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x01** (decimal: 1)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☑	PosAbort at PosInit		1 ≡ When a changeover to "positioning" mode is made, ramping down at the rate of deceleration set in C01251for normal stopping is carried out.
Bit 1 ☐	PosExecute active at PosInit		1 ≡When a changeover is made to the "positioning" mode, the specified profile is immediately executed if the "MCK PosExecute" control bit has also been set. If the MCK "PosExecute" control bit has not been set, the setpoint is continued.
Bit 2 ☐	Reserved		
Bit 3 ☐	Reserved		
Bit 4 ☐	ProfilStart at PosInit		1 ≡When it is changed to the "Positioning" operating mode, the defined profile is executed immediately without the need to set the MCK control bit "PosExecute".
Bit 5 ☐	Maximum jerk		1 ≡ The maximum jerk is calculated based on the acceleration and S-ramp time of the current profile and the sequence profile. This is then used in an accelerated drive for reducing the acceleration to 0 or for acceleration of the sequence profile. • Ab Version 14.00.00 ▸ Setting or activation of maximum jerk for traversing profiles
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01218

Parameter | Name: **C01218 | MCK: Position follower setting**			Data type: UNSIGNED_8 Index: 23357_d = $5B3D_h$
Settings for "Position follower" mode			Position follower
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x0C** (decimal: 12)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	Speed FF control.: nSpeedSetValue_a		1 ≡ speed feedforward control value comes from main setpoint *nSpeedSetValue_a*
Bit 1 ☐	Speed FF control: nSpeedAddValue_v		1 ≡ speed feedforward control value comes from additive speed value *nSpeedAddValue_v*
Bit 2 ☑	HW limit switch on		1 ≡ Travel range monitoring via hardware limit switch is active. The error response can be parameterised in C00595/1 and C00595/2.
Bit 3 ☑	SW limit switch on		1 ≡ Travel range monitoring via parameterised software limit positions is active. The error response can be parameterised in C00595/3 and C00595/4.
Bit 4 ☐	Reserved		
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Position controller off		1 ≡ Position controller is deactivated. Thus, the compensation of the following error is switched off.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01219

Parameter | Name: **C01219 | MCK: Speed follower setting**			Data type: UNSIGNED_8 Index: 23356_d = $5B3C_h$
Settings for "Speed follower" mode			Speed follower
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x0C** (decimal: 12)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	Reserved		
Bit 1 ☐	Reserved		
Bit 2 ☑	HW limit switch on		1 ≡ Travel range monitoring via hardware limit switch is active. The error response can be parameterised in C00595/1 and C00595/2.
Bit 3 ☑	SW limit switch on		1 ≡ Travel range monitoring via parameterised software limit positions is active. The error response can be parameterised in C00595/3 and C00595/4.
Bit 4 ☐	Reserved		
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Position controller off		1 ≡ Position controller is deactivated. Thus, the compensation of the following error is switched off.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01220

Parameter | Name: **C01220 | MCK: Ref. setting**			Data type: UNSIGNED_16 Index: 23355_d = $5B3B_h$
From version 14.00.00 Settings for "homing" mode Homing			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Reserved		
Bit 1	Flange-sensitive start		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Start deletes bHomePosAvailable		1 ≡ When homing is started (mode 4 ... 15), the reference information is deleted.
Bit 5	Maximum jerk		1 ≡ The maximum jerk is calculated based on the acceleration and S-ramp time of the current profile and the sequence profile. This is then used in an accelerated drive for reducing the acceleration to 0 or for acceleration of the sequence profile. Setting or activation of maximum jerk for traversing profiles
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	SW limits inactive after Ref Ok		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C01220/1	0x0000		MCK: Ref. setting
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01221

Parameter | Name: **C01221 | MCK: Ref. mode**		Data type: UNSIGNED_8 Index: 23354_d = $5B3A_h$
Selection of the homing mode.		▶ Homing
Selection list(Lenze setting printed in bold)		**Info**
4	>_Rp_<_TP	Positive direction - reversing on pre-stop mark (pos. edge) - to touch probe • From version 14.00.00
5	<_Rp_>_TP	Negative direction - reversing on pre-stop mark (pos. edge) - to touch probe • From version 14.00.00
6	>_Rn_>_TP	Positive direction - via pre-stop mark (neg. edge) - continuing in positive direction - to touch probe
7	<_Rn_<_TP	Negative direction - via pre-stop mark (neg. edge) - continuing in negativee direction - to touch probe
8	>_TP	Positive direction - to touch probe
9	<_TP	Negative direction - to touch probe
10	>_Lp_<_TP	Positive direction - reversing on pos. limit switch - to touch probe
11	<_Ln_>_TP	Negative direction - reversing on neg. limit switch - to touch probe
12	**>_Lp**	Positive direction - to pos. limit switch
13	<_Ln	Negative direction - to neg. limit switch
14	>_Mlim	Positive direction towards torque limit (C01222)
15	<_Mlim	Negative direction towards torque limit (C01222)
100	SetRef	Direct acceptance and setting of the home position • The measuring system is set based on the home position parameterised in C01227/2 when the drive is at standstill.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01222

Parameter | Name: **C01222 | MCK: Ref. M limit mode 14/15**			Data type: INTEGER_16 Index: 23353_d = $5B39_h$
Torque limit for homing modes 14 and 15 (homing towards positive stop) • 100 % ≡ maximum torque (C00057)			▶ Homing
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	199.99	**10.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01223

Parameter | Name: **C01223 | MCK: Ref. waiting time mode 14/15**			Data type: UNSIGNED_16 Index: 23352_d = $5B38_h$
Blocking time for homing modes 14 and 15 (homing towards positive stop) • The reference is set if an excess of the torque limit set in C01222 has been deteced over a time period defined here. ▶ Homing			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65000	**100 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01224

Parameter | Name: **C01224 | MCK: Ref. speeds**			Data type: INTEGER_32 Index: 23351_d = $5B37_h$
Speeds for reference search/homing ▶ Homing			
Setting range (min. value | unit | max. value)			
0.0000	unit/s	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01224/1	720.0000 unit/s		MCK: Ref. initial speed • Start speed for quick approach of the limit switch/pre-stop mark (depending on the selected homing mode).
C01224/2	180.0000 unit/s		MCK: Ref. search speed • Search speed for slower - but more precise - approaching of the touch probe sensor (sensor reference signal).
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01225

Parameter | Name: **C01225 | MCK: Ref. accelerations**			Data type: INTEGER_32 Index: 23350_d = $5B36_h$
Accelerations for reference search/homing ▶ Homing			
Setting range (min. value | unit | max. value)			
0.0000	unit/s2	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01225/1	720.0000 unit/s2		MCK: Ref. initial acceleration • Starting acceleration for the starting speed ramps.
C01225/2	720.0000 unit/s2		MCK: Ref. search acceleration • Search acceleration for the search speed ramps.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01226

Parameter | Name: C01226 | MCK: Ref. S-ramp time			Data type: UNSIGNED_16 Index: 23349_d = $5B35_h$
S-ramp time for reference search/homing • Setting "0 s" ≡ no rounding ▸ Homing			
Setting range (min. value | unit | max. value)			
0.000	s	10.000	
Subcodes	**Lenze setting**		**Info**
C01226/1	0.000 s		MCK: Ref. S-ramp time • S-ramp time for the starting and search speed ramps.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01227

Parameter | Name: C01227 | MCK: Ref. positions			Data type: INTEGER_32 Index: 23348_d = $5B34_h$
Positions for determining the zero position of the reference measuring system ▸ Homing			
Setting range (min. value | unit | max. value)			
-214748.3647	unit	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01227/1	0.0000 unit		MCK: Ref. offset reference degree • Relative traverse path by which the drive traverses correctly signed with search speed after detection of the sensor reference signal.
C01227/2	0.0000 unit		MCK: Ref. home position • Position with which setpoint and actual position are loaded after completion of homing.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01228

Parameter | Name: C01228 | MCK: Ref. sequence profile			Data type: UNSIGNED_8 Index: 23347_d = $5B33_h$
Number of the sequence profile the absolute position of which will be approached after homing. ▸ Homing			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		15	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01229

Parameter | Name: **C01229 | MCK: Position limiting values**			Data type: INTEGER_32 Index: 23346_d = $5B32_h$
Software limit position for limiting the valid traversing range • The error response to leaving the valid traversing range can be parameterised in C00595/3 and C00595/4. **Note:** For limiting the traversing range by means of software limit positions, the home position must be known and the positive software limit position must be higher than the negative software limit position! ▶ Limit position monitoring			
Setting range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01229/1	0.0000 units		MCK: Positive SW limit position (positive travel range limit)
C01229/2	0.0000 units		MCK: Negative SW limit position (negative travel range limit)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01230

Parameter | Name: **C01230 | Manual jog: Setting**			Data type: UNSIGNED_8 Index: 23345_d = $5B31_h$
Settings for "Manual jog" mode ▶ Manual jog			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x00** (decimal: 0)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	Breakpoints 1..4 on		1 ≡ approach of the parameterised breakpoint in manual jog direction
Bit 1 ☐	Time-based start of 2nd speed		1 ≡ Automatic change-over to second manual speed after the wating time set in C01235/1.
Bit 2 ☐	HW limit switch on		1 ≡ Travel range monitoring via hardware limit switch is active. The error response can be parameterised in C00595/1 and C00595/2.
Bit 3 ☐	SW limit switch on		1 ≡ Travel range monitoring via parameterised software limit positions is active. The error response can be parameterised in C00595/3 and C00595/4.
Bit 4 ☐	Reserved		
Bit 5 ☐	Maximum jerk		1 ≡ The maximum jerk is calculated based on the acceleration and S-ramp time of the current profile and the sequence profile. This is then used in an accelerated drive for reducing the acceleration to 0 or for acceleration of the sequence profile. • Ab Version 14.00.00 ▶ Setting or activation of maximum jerk for traversing profiles
Bit 6 ☐	Reserved		
Bit 7 ☐	Position controller off		1 ≡ Position controller is deactivated. Thus, the compensation of the following error is switched off.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01231

Parameter | Name: **C01231 | MCK: Manual jog speeds**			Data type: INTEGER_32 Index: 23344_d = $5B30_h$
Manual jog speeds ▶ Manual jog			
Setting range (min. value | unit | max. value)			
-214748.3647	units/s	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01231/1	360.0000 units/s		Manual jog: speed 1 • Regular manual speed
C01231/2	720.0000 units/s		Manual jog: Speed 2 • Second manual speed (can be activated via MCK control word)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01232

Parameter | Name: **C01232 | MCK: Manual jog accelerations**			Data type: INTEGER_32 Index: 23343_d = $5B2F_h$
Acceleration/deceleration for manual jog ▶ Manual jog			
Setting range (min. value | unit | max. value)			
-214748.3647	units/s2	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01232/1	720.0000 units/s2		Manual jog: Acceleration • Acceleration for ramp-up to manual speed.
C01232/2	720.0000 units/s2		Manual jog: Deceleration • Deceleration for manual speed ramp-down to standstill.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01233

Parameter | Name: **C01233 | MCK: Manual jog S-ramp time**			Data type: UNSIGNED_16 Index: 23342_d = $5B2E_h$
S-ramp time for manual jog • Setting "0 s" ≡ no rounding ▶ Manual jog			
Setting range (min. value | unit | max. value)			
0.000	s	10.000	
Subcodes	**Lenze setting**		**Info**
C01233/1	0.000 s		Manual jog: S-ramp time • S-ramp time for the manual speed ramps.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01234

Parameter | Name: **C01234 | MCK: Manual jog breakpoints**			Data type: INTEGER_32 Index: 23341_d = $5B2D_h$
Breakpoint positions for manual jog • The drive stops at the parameterised positions if these are located in the manual jog direction and the approaching of the breakpoints is switched on in C01230 via bit 0. ▸ Manual jog			
Setting range (min. value | unit | max. value)			
-214748.3647	unit	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01234/1	0.0000 unit		Manual jog: Breakpoint 1
C01234/2	0.0000 unit		Manual jog: Breakpoint 2
C01234/3	0.0000 unit		Manual jog: Breakpoint 3
C01234/4	0.0000 unit		Manual jog: Breakpoint 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01235

Parameter | Name: **C01235 | MCK: Manual jog waiting times**			Data type: UNSIGNED_16 Index: 23340_d = $5B2C_h$
Waiting times for manual jog ▸ Manual jog			
Setting range (min. value | unit | max. value)			
0.000	s	65.000	
Subcodes	**Lenze setting**		**Info**
C01235/1	5.000 s		Manual jog: Waiting time 2nd speed • Time after which the first manual speed (C01231/1) is accelerated to the second manual speed (C01231/2).
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01236

Parameter | Name: **C01236 | MCK: Speed follower**			Data type: INTEGER_32 Index: 23339_d = $5B2B_h$
Speeds for position follower ▸ Position follower			
Setting range (min. value | unit | max. value)			
-214748.3647	units/s	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01236/1	360.0000 units/s		PosFollower: Sync. speed • Synchronisation speed for approaching the setpoint position of the master.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01237

Parameter | Name: **C01237 | MCK: Acceleration follower**			Data type: INTEGER_32 Index: 23338_d = $5B2A_h$
Accelerations for position follower			▶ Position follower
Setting range (min. value | unit | max. value)			
-214748.3647	units/s2	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01237/1	720.0000 units/s2		Pos follower: Sync. accel. • Acceleration for ramp-up to synchronisation speed.
C01237/2	720.0000 units/s2		Pos follower: Sync. decel. • Deceleration for synchronisation speed ramp-down to standstill (to the setpoint position of the master).
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01238

Parameter | Name: **C01238 | MCK: S-ramp time follower**			Data type: UNSIGNED_16 Index: 23337_d = $5B29_h$
S-ramp times for position follower • Setting "0 s" ≡ no rounding			▶ Position follower
Setting range (min. value | unit | max. value)			
0.000	s	10.000	
Subcodes	**Lenze setting**		**Info**
C01238/1	0.000 s		Pos follower: Sync. S-ramp time • S-ramp time for the synchronisation speed ramps.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01239

Parameter | Name: **C01239 | Setpoint holding**			Data type: UNSIGNED_32 Index: 23336_d = $5B28_h$
From version 12.00.00			▶ Setpoint holding for bus runtime compensation
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	32	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01240

Parameter | Name: **C01240 | MCK: Control word**			Data type: UNSIGNED_32 Index: 23335_d = $5B27_h$
Bit-coded status of the MCK control word • The MCK control word is used to control the functions of the Motion Control Kernel.			▶ MCK control word
Display area (min. hex value | max. hex value)			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	OpMode_Bit0		Active operating mode - control bit 0
Bit 1	OpMode_Bit1		Active operating mode - control bit 1

Parameter | Name: **C01240 | MCK: Control word**		Data type: UNSIGNED_32 Index: 23335_d = $5B27_h$
Bit 2	OpMode_Bit2	Active operating mode - control bit 2
Bit 3	OpMode_Bit3	Active operating mode - control bit 4
Bit 4	ManJogPos	Control bit for manual jog in positive direction
Bit 5	ManJogNeg	Control bit for manual jog in negative direction
Bit 6	ManExecute2ndSpeed	Control bit for activation of second manual speed
Bit 7	ReleaseLimitSwitch	Control bit for retracting the hardware limit positions
Bit 8	HomStartStop	Control bit for start/stop homing
Bit 9	HomSetPos	Control bit for setting of home position • 0->1 ≡ sets position setpoint and actual position value to the home position at the MCK. • This function is independent of the operating modes.
Bit 10	HomResetPos	Control bit for reset of home position • 0->1 ≡ resets the information bit for the status "Home position known" (home position not known). • The position displays are not influenced by this.
Bit 11	EnableSpeedOverride	Control bit for activation of speed override • 1 ≡ activation of the override speed at the MCK for acceptance in a running speed profile.
Bit 12	EnableAccOverride	Control bit for activation of acceleration override • 1 ≡ activation of the override acceleration at the MCK for acceptance in a running speed profile.
Bit 13	EnableSRampOverride	Control bit for deactivation of S-shaping for speed profiles • 1 ≡ deactivates S-shaping for speed profiles
Bit 14	PosTeachSetPos	Control bit for accepting the selected setpoint position in the selected profile • 0->1 ≡ accepts the setpoint position at the MCK in the profile with the profile number defined via the MCK control word.
Bit 15	PosTeachActPos	Control bit for accepting the current actual position in the selected profile • 0->1 ≡ accepts the current actual position in the profile with the profile number defined via the MCK control word.
Bit 16	PosExecute	Control bit for starting a positioning profile • 0->1 ≡ starts a positioning profile selected via the control word in the "Positioning" mode.
Bit 17	PosFinishTarget	Control bit for completing processing of an interrupted positioning profile • 0->1 ≡ processing of a positioning profile which has already been started and has been interrupted because of a cancellation condition or a change of the operating mode is completed in the "Positioning" mode when the home position is known. **Note:** Completion of profile processing through this control is based on the target position detected at the profile start!
Bit 18	PosDisableFollowProfile	Control bit for suppression of the sequence profile • 1 ≡ positioning profiles whose profile numbers are entered in profile data sets for sequence profiles are not started after completion of the corresponding profile.

Parameter | Name: **C01240 | MCK: Control word**		Data type: UNSIGNED_32 Index: 23335_d = $5B27_h$
Bit 19	PosStop	Control bit for stopping the positioning profile • 0->1 ≡ ("Positioning" mode): Interrupts a running profile through deceleration along the set deceleration ramp to standstill. • 0->1 ≡ ("Stop" mode): Starts a new deceleration to standstill with the parameterised stop ramp.
Bit 20	PosModeBit0	Active positioning mode - control bit 0
Bit 21	PosModeBit1	Active positioning mode - control bit 1
Bit 22	PosModeBit2	Active positioning mode - control bit 2
Bit 23	PosModeBit3	Active positioning mode - control bit 3
Bit 24	ProfileNo_Bit0	Active positioning profile number - control bit 0
Bit 25	ProfileNo_Bit1	Active positioning profile number - control bit 1
Bit 26	ProfileNo_Bit2	Active positioning profile number - control bit 2
Bit 27	ProfileNo_Bit3	Active positioning profile number - control bit 3
Bit 28	ProfileNo_Bit4	Active positioning profile number - control bit 4
Bit 29	ProfileNo_Bit5	Active positioning profile number - control bit 5
Bit 30	ProfileNo_Bit6	Active positioning profile number - control bit 6
Bit 31	ProfileNo_Bit7	Active positioning profile number - control bit 7
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT		

C01241

Parameter | Name: **C01241 | MCK: Status word**			Data type: UNSIGNED_32 Index: 23334_d = $5B26_h$
Bit-coded status of the MCK status word • The MCK status word is used to output status messages of the MotionControlKernel. ▶ MCK status word			
Display area (min. hex value | max. hex value)			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	ActOpModeBit00		Active operating mode - status bit 0
Bit 1	ActOpModeBit01		Active operating mode - status bit 1
Bit 2	ActOpModeBit02		Active operating mode - status bit 2
Bit 3	ActOpModeBit03		Active operating mode - status bit 4
Bit 4	Busy		Status bit - profile generation active • 1 ≡ Internal profile generation is active. A speed profile is being generated.
Bit 5	Done		Status bit - profile generation completed • 1 ≡ Generation of a speed profile with the selected position has been completed.
Bit 6	AcceleratingActive		Status bit - acceleration process for profile generation active • 1 ≡ Profile generation phase is in the acceleration process.
Bit 7	ConstSpeedDuty		Status bit - constant speed for profile generation active • 1 ≡ Profile generation phase at constant speed active.
Bit 8	DeceleratingActive		Status bit - deceleration process for profile generation active • 1 ≡ Profile generation phase is in the deceleration process.

Parameter | Name: **C01241 | MCK: Status word**		Data type: UNSIGNED_32 Index: 23334_d = $5B26_h$
Bit 9	S_ShapingActive	Status bit - rounding for profile generation active • 1 ≡ Rounding during acceleration/deceleration active.
Bit 10	Pos. HW-Limit Detected	Status bit - positive hardware limit detected • 1 ≡ Positive limit switch has triggered. • Reset only possible via"Manual jog" mode!
Bit 11	Neg. HW-Limit Detected	Status bit - negative hardware limit detected • 1 ≡ Negative limit switch has triggered. • Reset only possible via"Manual jog" mode!
Bit 12	HomPosDone	Status bit - homing completed • 1 ≡ Homing has been completed.
Bit 13	HomPosAvailable	Status bit - home position known • 1 ≡ The home position has been detected and is known in the drive.
Bit 14	Pos. SW-Limit Detected	Status bit - positive software limit detected • 1 ≡ Positive software limit position overtravelled.
Bit 15	Neg. SW-Limit Detected	Status bit - negative software limit detected • 1 ≡ Negative software limit position overtravelled.
Bit 16	DwellTime	Status bit - transient effects in target position active • 1 ≡ Dwell time after reaching the setpoint position is active.
Bit 17	InTarget	Status bit - actual position is in the target window • 1 ≡ Dwell time has expired and current actual position is in the set target window.
Bit 18	PosDone	Status bit - positioning process completed • 1 ≡ Positioning profile has been completed in the "Positioning" mode. Setpoint position is in target.
Bit 19	ReadyToOperate	Status bit - readiness to process setpoint signals or setpoint commands
Bit 20	ActPosMode_Bit00	Active positioning mode - status bit 0
Bit 21	ActPosMode_Bit01	Active positioning mode - status bit 1
Bit 22	ActPosMode_Bit02	Active positioning mode - status bit 2
Bit 23	ActPosMode_Bit03	Active positioning mode - status bit 3
Bit 24	ActProfileNo_Bit00	Active positioning profile - status bit 0
Bit 25	ActProfileNo_Bit01	Active positioning profile - status bit 1
Bit 26	ActProfileNo_Bit02	Active positioning profile - status bit 2
Bit 27	ActProfileNo_Bit03	Active positioning profile - status bit 3
Bit 28	ActProfileNo_Bit04	Active positioning profile - status bit 4
Bit 29	ActProfileNo_Bit05	Active positioning profile - status bit 5
Bit 30	ActProfileNo_Bit06	Active positioning profile - status bit 6
Bit 31	ActProfileNo_Bit07	Active positioning profile - status bit 7
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT		

_ _

C01242

Parameter | Name: **C01242 | MCK: Current pos profile number**			Data type: UNSIGNED_8 Index: 23333_d = $5B25_h$
Display of the current profile number of the active profile in the "Positioning" mode ▶ Positioning			
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01243

Parameter | Name: **C01243 | MCK: Current operating mode**		Data type: UNSIGNED_8 Index: 23332_d = $5B24_h$
Display of the active operating mode of the Motion Control Kernel ▶ Basic drive functions		
Selection list (read only)		**Info**
0	Speed follower	Drive traverses according to a preselected speed setpoint
1	Homing	Drive finds its reference measuring system by setting the home position or homing
2	ManualJog	Drive can be traversed manually via initiators, e.g. for cleaning purposes or tool changes.
3	Positioning	Drive traverses according to defined travel profiles and motion processes
4	Stop	Drive is decelerated to standstill along a parameterised deceleration ramp
5	Position follower	Drive traverses according to a preselected position setpoint
15	StandBy	Internal operating mode for quick stop and pulse inhibit
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01244

Parameter | Name: **C01244 | MCK: Target detection - times**			Data type: UNSIGNED_16 Index: 23331_d = $5B23_h$
Timing for various MCK functions			
Setting range (min. value | unit | max. value)			
0	ms	60000	
Subcodes	**Lenze setting**		**Info**
C01244/1	100 ms		MCK: Dwell time - target position • Time expiring after reaching the setpoint position for positioning processes and enabling the actual position detection in the target position window. ▶ Target position monitoring
C01244/2	0 ms		MCK: Waiting time following error 1 • From version 02.00.00 ▶ Following error monitoring system
C01244/3	0 ms		MCK: Waiting time following error 2 • From version 02.00.00 ▶ Following error monitoring system
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01245

Parameter | Name: **C01245 | MCK: Target detection - positions**			Data type: INTEGER_32 Index: 23330_d = $5B22_h$
Position settings for various MCK functions			
Setting range (min. value | unit | max. value)			
0.0000	units	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01245/1	1.0000 units		MCK: Window target position • Window around the target position for comparison with the actual position to see whether the drive is in target. ▶ Target position monitoring
C01245/2	1.0000 units		MCK: Blocking zone Modulo ▶ Activation of the Modulo measuring system
C01245/3	0.0000 units		MCKI: Hysteresis position change
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01246

Parameter | Name: **C01246 | MCK: Select signal source**		Data type: UNSIGNED_8 Index: 23329_d = $5B21_h$
Selection of the touch probe signal source for reference search with touch probe detection • If the reference signal is to follow a real touch probe, the Touch-Probe-Interface must be configured accordingly. ▶ Homing		
Selection list		**Info**
0	No TP	
3	TP-DigIn3	
4	TP-DigIn4	
5	TP-DigIn5	
6	TP-DigIn6	
7	TP-DigIn7	
8	TP-Z-trace encoder	From version 12.00.00
9	TP-Z-trace resolver	From version 12.00.00
Subcodes	**Lenze setting**	**Info**
C01246/1	0: No TP	MCK: Ref. TP signal source **Note:** In case of setting "0: No TP", the digital input DI3 is used as signal source for reference search. An inversion of DI3 via C00114 is considered.
C01246/2	0: No TP	MCK: Set.Ref. signal source
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01247

Parameter | Name: **C01247 | MCK: Diagnostic word**			Data type: UNSIGNED_32 Index: 23328_d = $5B20_h$
From version 15.00.00			
Display area (min. hex value | max. hex value)			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	Traversing range: Limited		1 ≡ The traversing range is limited: • The software limit positions are set. • The reference is known. • There is no modulo measuring system available (C01201/1 = 0). **Note:** This status does not consider whether the software limit positions are activated or deactivated by the settings in the prevailing operating mode.
Bit 1	Traversing range: Modulo		1 ≡ A modulo measuring system is available as a cycle length has been set in C01201/1.
Bit 2	Pos. HW limit switch active		Evaluation of travel range limit switch active in positive direction
Bit 3	Neg. HW limit switch active		Evaluation of travel range limit switch active in negative direction
Bit 4	Pos. SW-Limit active		Evaluation of software limit positions active in positive direction
Bit 5	Neg. SW-Limit active		Evaluation of software limit positions active in negative direction
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	SpeedNormConstant: Low-Limit		1 ≡ Due to the set machine parameters, an internal calculation constant is limited to a possible minimum. Remedy: Increase reference speed in C00011.
Bit 9	SpeedNormConstant: Up-Limit		1 ≡ Due to the set machine parameters, an internal calculation constant is limited to a possible maximum. Remedy: Reduce reference speed in C00011.
Bit 10	Max. speed limit		1 ≡ Due to the set machine parameters, the maximum speed for specifications in [units/s] are limited to the factor 150 %.
Bit 11	Quick stop is active		1 ≡ Quick stop function is active.
Bit 12	Quick stop standstill is active		1 ≡ Standstill during active quick stop function (no ramp generation).
Bit 13	Max. speed is pending		1 ≡ The speed setpoint has exceeded the limit of ± 199.99 %. Possible cause: • Too high influence due to speed override • Increased speed by reduced acceleration with little jerk
Bit 14	Position resolution limit		1 ≡ The position resolution calculated by machine data entries has been limited to the maximum display area of 214748.3647 [Incr./Unit] with reference to the display in C01205/0. The real value is higher!
Bit 15	Modulo buffer overflow		1 ≡ The internal position memory for creating the saw tooth could not be emptied in one cycle. The speed provides a position difference per cycle that is larger than the cycle length.

Parameter | Name: **C01247 | MCK: Diagnostic word**		Data type: UNSIGNED_32 Index: 23328_d = $5B20_h$
Bit 16	Ck10: position > cycle length	Positioning mode: • The position selected for the positioning profile with process data interface is higher than the cycle length when the Modulo measuring system is activated. Homing mode: • The position selected for the sequence profile with process data interface is higher than the cycle length when the Modulo measuring system is activated.
Bit 17	Ck10: invalid speed	Cause 1: • When an infinite profile starts at a current speed = 0, the parameterised profile speed is specified with 0. Cause 2: • When a positioning profile is started for approaching a target position, the parameterised profile speed is specified with 0.
Bit 18	Ck10: invalid acceleration	When a profile is started, the parameterised deceleration is specified with 0.
Bit 19	Ck10: invalid deceleration	When a profile is started, the parameterised deceleration is specified with 0. While a positioning profile is executed to target position (current speed ≠ 0), the profile deceleration is written to 0, e.g. by Acc/Dec override.
Bit 20	Ck10: invalid final speed	If C02868/1 Bit 2 = FALSE, it is detected while crossing the finish line of a positioning process that the final profile speed is higher than the current speed. The final profile speed is not reached.
Bit 21	Ck10: reversal at overchange	If C02868/1 Bit 2 = FALSE, the reversing phase is requested during a positioning process to the target position with final profile speed ≠ 0.
Bit 22	Ck10: distance calculation error	During the profile start of a positioning profile, the profile data results in a longer distance than the defined distance (optimised by C02868/1 bit 6).
Bit 23	Reserved	
Bit 24	Reserved	
Bit 25	Reserved	
Bit 26	Reserved	
Bit 27	Reserved	
Bit 28	Reserved	
Bit 29	Reserved	
Bit 30	Reserved	
Bit 31	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT		

C01248

Parameter | Name: **C01248 | MCK: Ref. step**		Data type: UNSIGNED_8 Index: 23327_d = $5B1F_h$
From version 16.00.00		
Selection list		**Info**
0	No Operation	Initial step when changing to the homing mode
1	RefMarke: Search	Search travel until reference pre-stop mark
2	RefMarke: Stop	Stop when reference pre-stop mark has been detected
3	RefMarke: Reversal	Reversal of reference pre-stop mark
4	HW-Limit: Search	Search travel until limit switch
5	HW-Limit: Stop	Stop when limit switch has been detected
6	HW-Limit: Reversal	Reversal of limit switch
7	M-Limit: Search	Search travel until torque limit has been detected
8	M-Limit: Stop	Detection of torque limit, tolerance time is running
9	Reference: Search	Search travel until reference initiator
10	Reference: Stop	Stop when reference initiator has been detected
11	Reference: Offset	Travel offset path active
12	Reference: Sequence profile	Travel sequence profile after homing
13	Reference: Position	Travel to home position
14	Reference: Abort	Homing process aborted
15	Finish	Homing process executed
Subcodes		**Info**
C01248/1		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01251

Parameter | Name: **C01251 | MCK: Acceleration stop**			Data type: INTEGER_32 Index: 23324_d = $5B1C_h$
Acceleration parameter for stop ▶ Stop			
Setting range (min. value | unit | max. value)			
-214748.3647	units/s2	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01251/1	720.0000 units/s2		MCK: Stop: Decel. • Deceleration for setpoint speed ramp-down to standstill.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01252

Parameter | Name: **C01252 | MCK: S-ramp times stop**			Data type: UNSIGNED_16 Index: 23323_d = $5B1B_h$
S-ramp times for stop • Setting "0 s" ≡ no rounding ▸ Stop			
Setting range (min. value | unit | max. value)			
0.000	s	10.000	
Subcodes	**Lenze setting**		**Info**
C01252/1	0.000 s		MCK: Stop: S-ramp time • S-ramp time for setpoint speed ramp-down to standstill.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01292

Parameter | Name: **C01292 | MCK: Positioniermodus**		Data type: UNSIGNED_16 Index: 23283_d = $5AF3_h$
Selection list		**Info**
0	No action	
1	Absolute (shortest path)	
4	absolute (Cw)	
5	absolute (Ccw)	
Subcodes	**Lenze setting**	**Information**
C01292/1	1: absolute (beeline)	PosFolger: Sync. Modus
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01293

Parameter | Name: **C01293 | L_MckStateInterface_1: Status**			Data type: UNSIGNED_8 Index: 23282$_d$ = 5AF2$_h$
Setting range (min. hex value | max. hex value)			
0x00		0xFF	
Value is bit-coded:			**Info**
Bit 0	ActOpModeBit00		
Bit 1	ActOpModeBit01		
Bit 2	ActOpModeBit02		
Bit 3	ActOpModeBit03		
Bit 4	Busy		
Bit 5	Done		
Bit 6	HomPosAvailable		
Bit 7	ReadyToOperate		
Subcodes	**Lenze setting**		**Information**
C01293/1	0x00		L_MckStateInterface_1: Status 1
C01293/2	0x01		L_MckStateInterface_1: Status 2
C01293/3	0x02		L_MckStateInterface_1: Status 3
C01293/4	0x03		L_MckStateInterface_1: Status 4
C01293/5	0x04		L_MckStateInterface_1: Status 5
C01293/6	0x05		L_MckStateInterface_1: Status 6
C01293/7	0x0F		L_MckStateInterface_1: Status 7
C01293/8	0x0E		L_MckStateInterface_1: Status 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01294

Parameter | Name: **C01294 | Mode: Position calculation**		Data type: UNSIGNED_8 Index: 23281$_d$ = 5AF1$_h$
From version 12.00.00 Selection of the conversion mode		
Selection list		
0	dnOut_p=ConvWordsToDInt(HW_LW)	
1	16 bits: LW=+/-32767	
2	16 bits: HW=+/-; LW=0..65535	
3	32Bit: HW_LW=+/-214748.3647	
Subcodes	**Lenze setting**	**Info**
C01294/1	0: dnOut_p=ConvWordsToDInt(HW_LW)	L_ConvUnitsToIncr_1: PosCalcMode
C01294/2	0: dnOut_p=ConvWordsToDInt(HW_LW)	L_ConvUnitsToIncr_2: PosCalcMode
C01294/3	0: dnOut_p=ConvWordsToDInt(HW_LW)	L_ConvUnitsToIncr_3: PosCalcMode
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01295

Parameter | Name: **C01295 | L_MCKStateInterface_1: Pos. selection**		Data type: UNSIGNED_8 Index: 23280_d = $5AF0_h$
Selection of the position to be provided at the *dnPosOut_p* output of the FB L_MckStateInterface		
Selection list(Lenze setting printed in bold)		
0	dnPosIn_p	
1	Current feed	
2	dnSetPos_p	
3	**dnActPos_p**	
4	dnDeltaPos_p	
5	dnTargetPos_p	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01296

Parameter | Name: **C01296 | Mode: Position calculation**		Data type: UNSIGNED_8 Index: 23279_d = $5AEF_h$
		▸ MCKInterface
Selection list		
0	dnPosOut_p=dnPosIn_p	
1	16 bits: LW=+/-32767	
2	16 bits: HW=+/-; LW=0..65535	
3	32Bit: HW_LW=+/-214748.3647	
Subcodes	**Lenze setting**	**Info**
C01296/1	0: dnPosOut_p=dnPosIn_p	L_MckCtrlInterface_1: PosCalcMode
C01296/2	1: 16Bit: LW=+/-32767	L_MckStateInterface_1: PosDisplayMode
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01297

Parameter | Name: **C01297 | L_MckCtrlInterface_1: Alternative function**			Data type: UNSIGNED_8 Index: 23278_d = $5AEE_h$
Selection of the alternative function for bit 16 (PosExecute) in the MCK control word ▶ Alternative functions for "PosExecute" control bit			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x03** (decimal: 3)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☑	PosStop with PosExecute = FALSE		1 ≡ Abort positioning with PosExecute • PosExecute="0" stops a running positioning process by ramp-down to standstill.
Bit 1 ☑	HomingStartStop with PosExecute		1 ≡ Start/stop homing with PosExecute • PosExecute="1" starts homing, PosExecute="0" stops homing in the "Homing" mode.
Bit 2 ☐	SetProfilPosition with PosExecute		1 ≡ Teach set position with PosExecute • PosExecute="0->1" accepts the setpoint position at the MCK in the profile with the specified profile number.
Bit 3 ☐	SetProfilPosition at position change		1 ≡ Automatic acceptance of set position • The set position pending at the MCKInterface is automatically transferred to the profile with the preset profile number if a data change at the corresponding input for the setpoijnt is detected.
Bit 4 ☐	Profile start at position change		From version 02.00.00 1 ≡ Automatic profile start at position change • An automatic PosExecute takes place if the incremental position selection changes and this change is higher than set in the hysteresis for position change (C01245/3).
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01298

Parameter | Name: **C01298 | MCK: Operating mode change with profile no.**		Data type: UNSIGNED_8 Index: 23277_d = $5AED_h$
FB L_MckCtrlInterface_1: Assignment of an operating mode to a profile		
Selection list		**Info**
0	Op.Mode inputs	There will be no changed operating mode when the profile is changed. Instead, the operating mode requested "from outside" via the MCK control word applies.
1	Speed follower	Change to the "Speed follower" mode
2	Homing	Change to the "Homing" mode
3	Manual jog	Change to the "Manual jog" mode
4	Positioning	Change to the "Positioning" mode
5	Stop	Change to the "Stop" mode
6	Position follower	Change to the "Position follower" mode
Subcodes	**Lenze setting**	**Info**
C01298/1	1: Speed follower	MCK operating mode at profile no. 0 • Selection of the operating mode for profile 0 at L_MckCtrlInterface_1.
C01298/2	2: Homing	MCK operating mode at profile no. 1 • Selection of the operating mode for profile 1 at L_MckCtrlInterface_1.
C01298/3	3: Manual jog	MCK operating mode at profile no. 2 • Selection of the operating mode for profile 2 at L_MckCtrlInterface_1.
C01298/4	4: Positioning	MCK operating mode at profile no. 3...15 • Selection of the operating mode for profile 3 ... 15 at L_MckCtrlInterface_1.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01299

Parameter | Name: **C01299 | MCKI: Status MCKInterface**			Data type: UNSIGNED_8 Index: 23276_d = $5AEC_h$
FB L_MckCtrlInterface_1: Status messages			
Display area (min. hex value | max. hex value)			
0x00		0xFF	
Value is bit-coded:			**Info**
Bit 0	InvalidOperationMode		Invalid operating mode selection • 1 ≡ The selected operating mode is not defined/invalid.
Bit 1	InvalidPosMode		Invalid positioning mode selection • 1 ≡ The selected positioning mode is not defined/invalid.
Bit 2	InvalidProfileNo		Invalid profile number selection • 1 ≡ The selected profile number refers to a profile data set that does not exist.
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C01300

Parameter | Name: **C01300 | Profile data: Positioning mode**		Data type: UNSIGNED_8 Index: 23275_d = $5AEB_h$
Selection of the positioning mode for profiles 1 ... 15 ▸ Profile entry		
Selection list		**Info**
1	Absolute (shortest path)	Absolute positioning • The profile position corresponds to the target position.
2	Continuous	Continuous positioning without approaching a target position
3	Relative	Relative positioning • The profile position determines the distance to be traversed.
4	absolute (Cw)	Clockwise absolute positioning • The zero position of the axis can be exceed in this direction. • The profile position corresponds to the target position.
5	absolute (Ccw)	Counter-clockwise absolute positioning • The zero position of the axis can be exceed in this direction. • The profile position corresponds to the target position.
8	Absolute (shortest path) to TP	Absolute positioning • The profile position corresponds to the target position. • After a touch probe is detected, the TP profile (C01308/1...15) is executed.

Parameter | Name: **C01300 | Profile data: Positioning mode**		Data type: UNSIGNED_8 Index: 23275_d = $5AEB_h$
9	Continuous to TP	Continuous positioning without approaching a target position • After a touch probe is detected, the TP profile (C01308/1...15) is executed.
10	Relative to TP	Relative positioning • The profile position determines the distance to be traversed. • After a touch probe is detected, the TP profile (C01308/1...15) is executed.
11	Absolute (Cw) on TP	Clockwise absolute positioning • The zero position of the axis can be exceed in this direction. • The profile position corresponds to the target position. • After a touch probe is detected, the TP profile (C01308/1...15) is executed.
12	Absolute (Ccw) on TP	Counter-clockwise absolute positioning • The zero position of the axis can be exceed in this direction. • The profile position corresponds to the target position. • After a touch probe is detected, the TP profile (C01308/1...15) is executed.
Subcodes	**Lenze setting**	**Info**
C01300/1	1: absolute (beeline)	Profiles 1 ... 15: Positioning mode
C01300/...		
C01300/15		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01301

Parameter | Name: **C01301 | Profile data: Position**			Data type: INTEGER_32 Index: 23274_d = $5AEA_h$
Selection of the positions for profiles 1 ... 15 ▸ Profile entry			
Setting range (min. value | unit | max. value)			
-214748.3647	unit	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01301/1	360.0000 unit		Profiles 1 ... 15: Position
C01301/...			
C01301/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01302

Parameter | Name: **C01302 | Profile data: Speed**			Data type: INTEGER_32 Index: 23273_d = $5AE9_h$
Selection of the maximum speeds for profiles 1 ... 15			▸ Profile entry
Setting range (min. value | unit | max. value)			
-214748.3647	unit/s	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01302/1	360.0000 unit/s		Profiles 1 ... 15: Speed
C01302/...			
C01302/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01303

Parameter | Name: **C01303 | Profile data: Acceleration**			Data type: INTEGER_32 Index: 23272_d = $5AE8_h$
Selection of the accelerations for profiles 1 ... 15			▸ Profile entry
Setting range (min. value | unit | max. value)			
-214748.3647	unit/s2	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01303/1	720.0000 unit/s2		Profiles 1 ... 15: Acceleration
C01303/...			
C01303/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01304

Parameter | Name: **C01304 | Profile data: Deceleration**			Data type: INTEGER_32 Index: 23271_d = $5AE7_h$
Selection of the decelerations for profiles 1 ... 15			▸ Profile entry
Setting range (min. value | unit | max. value)			
-214748.3647	unit/s2	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01304/1	720.0000 unit/s2		Profiles 1 ... 15: Deceleration
C01304/...			
C01304/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01305

Parameter | Name: **C01305 | Profile data: Final speed**			Data type: INTEGER_32 Index: 23270_d = $5AE6_h$
Selection of the final speeds for profiles 1 ... 15 • For profile linkage with overchange. Profile entry			
Setting range (min. value | unit | max. value)			
-214748.3647	unit/s	214748.3647	
Subcodes	**Lenze setting**		**Info**
C01305/1	0.0000 unit/s		Profiles 1 ... 15: Final speed
C01305/...			
C01305/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01306

Parameter | Name: **C01306 | Profile data: S-ramp time**			Data type: UNSIGNED_16 Index: 23269_d = $5AE5_h$
Selection of the S-ramp times for profiles 1 ... 15 • The maximum S-ramp time is the time in which the maximum acceleration or the maximum deceleration is reached along a ramp (jerk limitation). Profile entry			
Setting range (min. value | unit | max. value)			
0.000	s	10.000	
Subcodes	**Lenze setting**		**Info**
C01306/1	0.000 s		Profiles 1 ... 15: S-ramp time
C01306/...			
C01306/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01307

Parameter | Name: **C01307 | Profile data: Sequence profile**			Data type: UNSIGNED_8 Index: 23268_d = $5AE4_h$
Selection of the sequence profile numbers for profiles 1 ... 15 • For profile linkage Profile entry			
Setting range (min. value | unit | max. value)			
0		15	
Subcodes	**Lenze setting**		**Info**
C01307/1	0		Profiles 1 ... 15: Sequence profile
C01307/...			
C01307/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01308

Parameter | Name: **C01308 | Profile data: TP profile**			Data type: UNSIGNED_8 Index: 23267$_d$ = 5AE3$_h$
Selection of the touch-probe profile numbers for profile 1 ... 15 • Only relevant for positioning modes with touch-probe. ▸ Profile entry			
Setting range (min. value | unit | max. value)			
0		15	
Subcodes	**Lenze setting**		**Info**
C01308/1	0		Profile 1 ... 15: TP profile
C01308/...			
C01308/15			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01309

Parameter | Name: **C01309 | Profile data: TP signal source**		Data type: UNSIGNED_8 Index: 23266$_d$ = 5AE2$_h$
Selection of the touch-probe signal sources for profile 1 ... 15 • Only relevant for positioning modes with touch-probe. ▸ Profile entry		
Selection list		
3	TP-DigIn3	
4	TP-DigIn4	
5	TP-DigIn5	
6	TP-DigIn6	
7	TP-DigIn7	
8	TP-Z-trace encoder	
9	TP-Z-trace resolver	
Subcodes	**Lenze setting**	**Info**
C01309/1	3: TP-DigIn3	Profiles 1 ... 15: Sequence profile
C01309/...		
C01309/15		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01310

Parameter | Name: **C01310 | Profile data: PI position**		Data type: UNSIGNED_8 Index: 23265_d = $5AE1_h$
Selection list		**Info**
0	Parameter value (C1301/n)	
1	dnProcessIn1_p	
2	dnProcessIn2_p	
3	dnProcessIn3_p	
4	dnProcessIn4_p	
Subcodes	**Lenze setting**	**Information**
C01310/1	0: Parameter value (C1301/n)	Profile 1: PI position
C01310/2	0: Parameter value (C1301/n)	Profil 2: PI-Position
C01310/3	0: Parameter value (C1301/n)	Profile 3: PI position
C01310/4	0: Parameter value (C1301/n)	Profile 4: PI position
C01310/5	0: Parameter value (C1301/n)	Profile 5: PI position
C01310/6	0: Parameter value (C1301/n)	Profile 6: PI position
C01310/7	0: Parameter value (C1301/n)	Profile 7: PI position
C01310/8	0: Parameter value (C1301/n)	Profile 8: PI position
C01310/9	0: Parameter value (C1301/n)	Profile 9: PI position
C01310/10	0: Parameter value (C1301/n)	Profile 10: PI position
C01310/11	0: Parameter value (C1301/n)	Profile 11: PI position
C01310/12	0: Parameter value (C1301/n)	Profile 12: PI position
C01310/13	0: Parameter value (C1301/n)	Profile 13: PI position
C01310/14	0: Parameter value (C1301/n)	Profile 14: PI position
C01310/15	0: Parameter value (C1301/n)	Profile 15: PI position
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01311

Parameter | Name: **C01311 | Profile data: PI speed**		Data type: UNSIGNED_8 Index: 23264_d = $5AE0_h$
Selection list		**Info**
0	Parameter value (C1302/n)	
1	nProcessIn1_a x (C1302/n)	
2	nProcessIn2_a x (C1302/n)	
3	nProcessIn3_a x (C1302/n)	
4	nProcessIn4_a x (C1302/n)	
5	nProcessIn5_a x (C1302/n)	
6	nProcessIn6_a x (C1302/n)	
7	nProcessIn7_a x (C1302/n)	
8	nProcessIn8_a x (C1302/n)	
Subcodes	**Lenze setting**	**Information**
C01311/1	0: Parameter value (C1302/n)	Profile 1: PI speed
C01311/2	0: Parameter value (C1302/n)	Profile 2: PI speed
C01311/3	0: Parameter value (C1302/n)	Profile 3: PI speed
C01311/4	0: Parameter value (C1302/n)	Profile 4: PI speed
C01311/5	0: Parameter value (C1302/n)	Profile 5: PI speed
C01311/6	0: Parameter value (C1302/n)	Profile 6: PI speed
C01311/7	0: Parameter value (C1302/n)	Profile 7: PI speed
C01311/8	0: Parameter value (C1302/n)	Profile 8: PI speed
C01311/9	0: Parameter value (C1302/n)	Profile 9: PI speed
C01311/10	0: Parameter value (C1302/n)	Profile 10: PI speed
C01311/11	0: Parameter value (C1302/n)	Profile 11: PI speed
C01311/12	0: Parameter value (C1302/n)	Profile 12: PI speed
C01311/13	0: Parameter value (C1302/n)	Profile 13: PI speed
C01311/14	0: Parameter value (C1302/n)	Profile 14: PI speed
C01311/15	0: Parameter value (C1302/n)	Profile 15: PI speed
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01312

Parameter | Name: **C01312 | Profile data: PI accel.**		Data type: UNSIGNED_8 Index: 23263_d = $5ADF_h$
Selection list		**Info**
0	Parameter value (C1303/n)	
1	| nProcessIn1_a x (C1303/n) |	
2	| nProcessIn2_a x (C1303/n) |	
3	| nProcessIn3_a x (C1303/n) |	
4	| nProcessIn4_a x (C1303/n) |	
5	| nProcessIn5_a x (C1303/n) |	
6	| nProcessIn6_a x (C1303/n) |	
7	| nProcessIn7_a x (C1303/n) |	
8	| nProcessIn8_a x (C1303/n) |	
Subcodes	**Lenze setting**	**Information**
C01312/1	0: Parameter value (C1303/n)	Profile 1: PI accel.
C01312/2	0: Parameter value (C1303/n)	Profile 2: PI accel.
C01312/3	0: Parameter value (C1303/n)	Profile 3: PI accel.
C01312/4	0: Parameter value (C1303/n)	Profile 4: PI accel.
C01312/5	0: Parameter value (C1303/n)	Profile 5: PI accel.
C01312/6	0: Parameter value (C1303/n)	Profile 6: PI accel.
C01312/7	0: Parameter value (C1303/n)	Profile 7: PI accel.
C01312/8	0: Parameter value (C1303/n)	Profile 8: PI accel.
C01312/9	0: Parameter value (C1303/n)	Profile 9: PI accel.
C01312/10	0: Parameter value (C1303/n)	Profile 10: PI accel.
C01312/11	0: Parameter value (C1303/n)	Profile 11: PI accel.
C01312/12	0: Parameter value (C1303/n)	Profile 12: PI accel.
C01312/13	0: Parameter value (C1303/n)	Profile 13: PI accel.
C01312/14	0: Parameter value (C1303/n)	Profile 14: PI accel.
C01312/15	0: Parameter value (C1303/n)	Profile 15: PI accel.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01313

Parameter | Name: **C01313 | Profile data: PI decel.**		Data type: UNSIGNED_8 Index: 23262_d = $5ADE_h$
Selection list		**Info**
0	Parameter value (C1304/n)	
1	| nProcessIn1_a x (C1304/n) |	
2	| nProcessIn2_a x (C1304/n) |	
3	| nProcessIn3_a x (C1304/n) |	
4	| nProcessIn4_a x (C1304/n) |	
5	| nProcessIn5_a x (C1304/n) |	
6	| nProcessIn6_a x (C1304/n) |	
7	| nProcessIn7_a x (C1304/n) |	
8	| nProcessIn8_a x (C1304/n) |	
Subcodes	**Lenze setting**	**Information**
C01313/1	0: Parameter value (C1304/n)	Profile 1: PI decel.
C01313/2	0: Parameter value (C1304/n)	Profile 2: PI decel.
C01313/3	0: Parameter value (C1304/n)	Profile 3: PI decel.
C01313/4	0: Parameter value (C1304/n)	Profile 4: PI decel.
C01313/5	0: Parameter value (C1304/n)	Profile 5: PI decel.
C01313/6	0: Parameter value (C1304/n)	Profile 6: PI decel.
C01313/7	0: Parameter value (C1304/n)	Profile 7: PI decel.
C01313/8	0: Parameter value (C1304/n)	Profile 8: PI decel.
C01313/9	0: Parameter value (C1304/n)	Profile 9: PI decel.
C01313/10	0: Parameter value (C1304/n)	Profile 10: PI decel.
C01313/11	0: Parameter value (C1304/n)	Profile 11: PI decel.
C01313/12	0: Parameter value (C1304/n)	Profile 12: PI decel.
C01313/13	0: Parameter value (C1304/n)	Profile 13: PI decel.
C01313/14	0: Parameter value (C1304/n)	Profile 14: PI decel.
C01313/15	0: Parameter value (C1304/n)	Profile 15: PI decel.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01314

Parameter | Name: C01314 | Profile data: PI final speed		Data type: UNSIGNED_8 Index: 23261_d = $5ADD_h$
Selection list		**Info**
0	Parameter value (C1305/n)	
1	nProcessIn1_a x (C1305/n)	
2	nProcessIn2_a x (C1305/n)	
3	nProcessIn3_a x (C1305/n)	
4	nProcessIn4_a x (C1305/n)	
5	nProcessIn5_a x (C1305/n)	
6	nProcessIn6_a x (C1305/n)	
7	nProcessIn7_a x (C1305/n)	
8	nProcessIn8_a x (C1305/n)	
Subcodes	**Lenze setting**	**Information**
C01314/1	0: Parameter value (C1305/n)	Profile 1: PI final speed
C01314/2	0: Parameter value (C1305/n)	Profile 2: PI final speed
C01314/3	0: Parameter value (C1305/n)	Profile 3: PI final speed
C01314/4	0: Parameter value (C1305/n)	Profile 4: PI final speed
C01314/5	0: Parameter value (C1305/n)	Profile 5: PI final speed
C01314/6	0: Parameter value (C1305/n)	Profile 6: PI final speed
C01314/7	0: Parameter value (C1305/n)	Profile 7: PI final speed
C01314/8	0: Parameter value (C1305/n)	Profile 8: PI final speed
C01314/9	0: Parameter value (C1305/n)	Profile 9: PI final speed
C01314/10	0: Parameter value (C1305/n)	Profile 10: PI final speed
C01314/11	0: Parameter value (C1305/n)	Profile 11: PI final speed
C01314/12	0: Parameter value (C1305/n)	Profile 12: PI final speed
C01314/13	0: Parameter value (C1305/n)	Profile 13: PI final speed
C01314/14	0: Parameter value (C1305/n)	Profile 14: PI final speed
C01314/15	0: Parameter value (C1305/n)	Profile 15: PI final speed
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01315

Parameter | Name: **C01315 | Profile data: PI S-ramp time**		Data type: UNSIGNED_8 Index: 23260_d = $5ADC_h$
Selection list		**Info**
0	Parameter value (C1306/n)	
1	| nProcessIn1_a | x (C1306/n)	
2	| nProcessIn2_a | x (C1306/n)	
3	| nProcessIn3_a | x (C1306/n)	
4	| nProcessIn4_a | x (C1306/n)	
5	| nProcessIn5_a | x (C1306/n)	
6	| nProcessIn6_a | x (C1306/n)	
7	| nProcessIn7_a | x (C1306/n)	
8	| nProcessIn8_a | x (C1306/n)	
Subcodes	**Lenze setting**	**Information**
C01315/1	0: Parameter value (C1306/n)	Profile 1: PI S-ramp time
C01315/2	0: Parameter value (C1306/n)	Profile 2: PI S-ramp time
C01315/3	0: Parameter value (C1306/n)	Profile 3: PI S-ramp time
C01315/4	0: Parameter value (C1306/n)	Profile 4: PI S-ramp time
C01315/5	0: Parameter value (C1306/n)	Profile 5: PI S-ramp time
C01315/6	0: Parameter value (C1306/n)	Profile 6: PI S-ramp time
C01315/7	0: Parameter value (C1306/n)	Profile 7: PI S-ramp time
C01315/8	0: Parameter value (C1306/n)	Profile 8: PI S-ramp time
C01315/9	0: Parameter value (C1306/n)	Profile 9: PI S-ramp time
C01315/10	0: Parameter value (C1306/n)	Profile 10: PI S-ramp time
C01315/11	0: Parameter value (C1306/n)	Profile 11: PI S-ramp time
C01315/12	0: Parameter value (C1306/n)	Profile 12: PI S-ramp time
C01315/13	0: Parameter value (C1306/n)	Profile 13: PI S-ramp time
C01315/14	0: Parameter value (C1306/n)	Profile 14: PI S-ramp time
C01315/15	0: Parameter value (C1306/n)	Profile 15: PI S-ramp time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01320

Parameter | Name: **C01320 | Profile data: Position**			Data type: INTEGER_32 Index: 23255_d = $5AD7_h$
Display range (min. value | unit | max. value)			
-214748.3647	unit	214748.3647	
Subcodes			**Information**
C01320/1			Profile 1: Position
C01320/2			Profile 2: Position
C01320/3			Profile 3: Position
C01320/4			Profile 4: Position
C01320/5			Profile 5: Position
C01320/6			Profile 6: Position
C01320/7			Profile 7: Position
C01320/8			Profile 8: Position
C01320/9			Profile 9: Position
C01320/10			Profile 10: Position
C01320/11			Profile 11: Position
C01320/12			Profile 12: Position
C01320/13			Profile 13: Position
C01320/14			Profile 14: Position
C01320/15			Profile 15: Position
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01321

Parameter | Name: **C01321 | Profile data: Speed**			Data type: INTEGER_32 Index: 23254_d = $5AD6_h$
Display range (min. value | unit | max. value)			
-214748.3647	unit/s	214748.3647	
Subcodes			**Information**
C01321/1			Profile 1: Speed
C01321/2			Profile 2: Speed
C01321/3			Profile 3: Speed
C01321/4			Profile 4: Speed
C01321/5			Profile 5: Speed
C01321/6			Profile 6: Speed
C01321/7			Profile 7: Speed
C01321/8			Profile 8: Speed
C01321/9			Profile 9: Speed
C01321/10			Profile 10: Speed
C01321/11			Profile 11: Speed
C01321/12			Profile 12: Speed
C01321/13			Profile 13: Speed
C01321/14			Profile 14: Speed
C01321/15			Profile 15: Speed
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01322

Parameter | Name: C01322 | Profile data: Acceleration			Data type: INTEGER_32 Index: 23253_d = $5AD5_h$
Display range (min. value | unit | max. value)			
-214748.3647	unit/s2	214748.3647	
Subcodes			**Information**
C01322/1			Profile 1: Accel.
C01322/2			Profile 2: Accel.
C01322/3			Profile 3: Accel.
C01322/4			Profile 4: Accel.
C01322/5			Profile 5: Accel.
C01322/6			Profile 6: Accel.
C01322/7			Profile 7: Accel.
C01322/8			Profile 8: Accel.
C01322/9			Profile 9: Accel.
C01322/10			Profile 10: Accel.
C01322/11			Profile 11: Accel.
C01322/12			Profile 12: Accel.
C01322/13			Profile 13: Accel.
C01322/14			Profile 14: Accel.
C01322/15			Profile 15: Accel.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01323

Parameter | Name: C01323 | Profile data: Deceleration			Data type: INTEGER_32 Index: 23252_d = $5AD4_h$
Display range (min. value | unit | max. value)			
-214748.3647	unit/s2	214748.3647	
Subcodes			**Information**
C01323/1			Profile 1: Decel.
C01323/2			Profile 2: Decel.
C01323/3			Profile 3: Decel.
C01323/4			Profile 4: Decel.
C01323/5			Profile 5: Decel.
C01323/6			Profile 6: Decel.
C01323/7			Profile 7: Decel.
C01323/8			Profile 8: Decel.
C01323/9			Profile 9: Decel.
C01323/10			Profile 10: Decel.
C01323/11			Profile 11: Decel.
C01323/12			Profile 12: Decel.
C01323/13			Profile 13: Decel.
C01323/14			Profile 14: Decel.
C01323/15			Profile 15: Decel.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01324

Parameter | Name: **C01324 | Profile data: Final speed**			Data type: INTEGER_32 Index: 23251_d = $5AD3_h$
Display range (min. value | unit | max. value)			
-214748.3647	unit/s	214748.3647	
Subcodes			**Information**
C01324/1			Profile 1: Final speed
C01324/2			Profile 2: Final speed
C01324/3			Profile 3: Final speed
C01324/4			Profile 4: Final speed
C01324/5			Profile 5: Final speed
C01324/6			Profile 6: Final speed
C01324/7			Profile 7: Final speed
C01324/8			Profile 8: Final speed
C01324/9			Profile 9: Final speed
C01324/10			Profile 10: Final speed
C01324/11			Profile 11: Final speed
C01324/12			Profile 12: Final speed
C01324/13			Profile 13: Final speed
C01324/14			Profile 14: Final speed
C01324/15			Profile 15: Final speed
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C01325

Parameter | Name: **C01325 | Profile data: S-ramp time**			Data type: UNSIGNED_16 Index: 23250_d = $5AD2_h$
Display range (min. value | unit | max. value)			
0.000	s	10.000	
Subcodes			**Information**
C01325/1			Profile 1: S-ramp time
C01325/2			Profile 2: S-ramp time
C01325/3			Profile 3: S-ramp time
C01325/4			Profile 4: S-ramp time
C01325/5			Profile 5: S-ramp time
C01325/6			Profile 6: S-ramp time
C01325/7			Profile 7: S-ramp time
C01325/8			Profile 8: S-ramp time
C01325/9			Profile 9: S-ramp time
C01325/10			Profile 10: S-ramp time
C01325/11			Profile 11: S-ramp time
C01325/12			Profile 12: S-ramp time
C01325/13			Profile 13: S-ramp time
C01325/14			Profile 14: S-ramp time
C01325/15			Profile 15: S-ramp time
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01350

Parameter | Name: **C01350 | ACDrive: Drive mode**		Data type: UNSIGNED_8 Index: 23225_d = $5AB9_h$
Selection list		
1	Speed mode	
3	Torque mode	
Subcodes	**Lenze setting**	**Info**
C01350/1	1: Speed mode	ACDrive: Drive mode
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01351

Parameter | Name: **C01351 | ACDrive: Control word**		Data type: UNSIGNED_16 Index: 23224_d = $5AB8_h$
From version 13.00.00 Display of the "AC Drive Profile" control word for the inverter • Detailed information on the "AC Drive Profile" can be found in the EtherNet/IP™ communication manual. ▸ Actuating drive speed (AC Drive Profile)		
Display area (min. hex value | max. hex value)		
0x0000	0xFFFF	
Value is bit-coded:		**Info**
Bit 0	Run Forward	Relationships between Run1 and Run2 and trigger events can be found in the chapter "Run/Stop Event".
Bit 1	Run Backward	
Bit 2	Fault Reset	0->1 ≡ Reset error 0 ≡ No response
Bit 3	Reserved	
Bit 4	Reserved	
Bit 5	NetCtrl	Run/Stop control: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 6	NetRef	Reference speed/reference torque: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 7	Reserved	
Bit 8	Reserved	
Bit 9	Reserved	
Bit 10	Reserved	
Bit 11	Reserved	
Bit 12	Reserved	
Bit 13	Reserved	
Bit 14	Reserved	
Bit 15	Reserved	
Subcodes		**Info**
C01351/1		ACDrive: Control word
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT		

C01352

Parameter | Name: **C01352 | ACDrive: Status word**			Data type: UNSIGNED_16 Index: 23223_d = $5AB7_h$
From version 13.00.00 Display of the "AC Drive Profile" status word of the inverter • Detailed information on the "AC Drive Profile" can be found in the EtherNet/IP™ communication manual. ▶ Actuating drive speed (AC Drive Profile)			
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Faulted		0 ≡ No errors 1 ≡ Errors have occurred
Bit 1	Warning		0 ≡ No warnings 1 ≡ Warnings have occurred
Bit 2	Running1 (Fwd)		Relationships between Run1 and Run2 and trigger events can be found in the chapter "Run/Stop Event".
Bit 3	Running2 (Rev)		
Bit 4	Ready		0 ≡ Different status than in case of "1" 1 ≡ Ready or Enabled or Stopping
Bit 5	Ctrl from Net		Run/Stop control: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 6	Ref from Net		Reference speed/reference torque: 0 ≡ via local setting in the device or terminal 1 ≡ via fieldbus (e.g. by the scanner)
Bit 7	At Reference		1 ≡ Currently, the inverter runs with the reference speed or reference torque (depending on the "drive mode" set in C01350/1).
Bit 8	DriveState_0		The "Drive State" is coded as follows: 0: Manufacturer-specific (not used with 8400) 1: Start-up (drive initialisation) 2: Not_Ready (mains voltage switched off) 3: Ready (mains voltage switched-on) 4: Enabled (drive has received "Run" command) 5: Stopping (drive has received "Stop" command and is stopped) 6: Fault_Stop (drive is stopped due to an error) 7: Faulted (errors have occurred)
Bit 9	DriveState_1		
Bit 10	DriveState_2		
Bit 11	DriveState_3		
Bit 12	DriveState_4		
Bit 13	DriveState_5		
Bit 14	DriveState_6		
Bit 15	DriveState_7		
Subcodes			**Info**
C01352/1			ACDrive: Status word
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C01353

Parameter | Name: **C01353 | ACDrive: Setpoint scaling**			Data type: INTEGER_8 Index: 23222_d = $5AB6_h$
From version 13.00.00 Actuating drive speed (AC Drive Profil): Scaling of the speed and torque values			
Setting range (min. value | unit | max. value)			
-128		127	
Subcodes	**Lenze setting**		**Info**
C01353/1	0		ACDrive: Speed scaling
C01353/2	0		ACDrive: Torque scaling
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01400

Parameter | Name: **C01400 | L_Sequencer_1: Sequence step**			Data type: UNSIGNED_16 Index: 23175_d = $5A87_h$
From version 12.00.00 In the subcodes, the calls of the actions required for the positioning program are stored. In this way, the basic sequence (with the exception of the branches) is defined. L_Sequencer_1 (sequence control)			
Setting range (min. value | unit | max. value)			
0		10005	
Subcodes	**Lenze setting**		**Information**
C01400/1	0		L_Sequencer_1: Action in sequence step 1
C01400/...			...
C01400/100			L_Sequencer_1: Action in sequence step 100
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01401

Parameter | Name: **C01401 | L_Sequencer_1: Current step**			Data type: UNSIGNED_16 Index: 23174_d = $5A86_h$
From version 12.00.00 L_Sequencer_1 (sequence control)			
Display range (min. value | unit | max. value)			
0		100	
Subcodes			**Information**
C01401/1			L_Sequencer_1: Current step
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01402

Parameter | Name: **C01402 | L_Sequencer_1: Sequence control: Status**			Data type: UNSIGNED_16 Index: 23173_d = $5A85_h$
From version 12.00.00 0 - READY: Positioning program ready to start/Program end reached. 1 - RUN: Positioning program running. 2 - PAUSE: Current step interrupted, pause active. 3 - BREAK: Positioning program interrupted. 4 - RESET: Positioning program reset. ▶ L_Sequencer_1 (sequence control)			
Display range (min. value | unit | max. value)			
0		4	
Subcodes			**Information**
C01402/1			L_Sequencer_1: Sequence control: Status
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01403

Parameter | Name: **C01403 | L_Sequencer_1: Sequence control: Control word**			Data type: UNSIGNED_16 Index: 23172_d = $5A84_h$
From version 12.00.00 Bit-coded code for controlling the sequencer with a PC or master control via the parameter channel. Setting a bit to "1" activates the corresponding function: • Bit 0: Start • Bit 1: Pause • Bit 2: Break • Bit 3: Cancel • Bit 4: Reset • Bit 5: NextStep ▶ L_Sequencer_1 (sequence control)			
Setting range (min. value | unit | max. value)			
0		64	
Subcodes	**Lenze setting**		**Information**
C01403/1	0		L_Sequencer_1: Sequence control: Control word
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01404

Parameter | Name: **C01404 | L_Sequencer_1: Step for bCancel = TRUE**			Data type: UNSIGNED_16 Index: 23171_d = $5A83_h$
From version 12.00.00 ▶ L_Sequencer_1 (sequence control)			
Setting range (min. value | unit | max. value)			
0		101	
Subcodes	**Lenze setting**		**Information**
C01404/1	0		L_Sequencer_1: Step for bCancel = TRUE
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01405

Parameter | Name: **C01405 | L_Sequencer_1: PSInput**		Data type: UNSIGNED_16 Index: 23170$_d$ = 5A82$_h$
From version 12.00.00		▸ L_Sequencer_1: "Positioning" action
Selection list		
0	Waiting function deactivated	
1	Input 1(wDigitalInp. Bit00)	
2	Input 2(wDigitalInp. Bit01)	
3	Input 3(wDigitalInp. Bit02)	
4	Input 4(wDigitalInp. Bit03)	
5	Input 5(wDigitalInp. Bit04)	
6	Input 6(wDigitalInp. Bit05)	
7	Input 7(wDigitalInp. Bit06)	
8	Input 8(wDigitalInp. Bit07)	
9	Input 9(wDigitalInp. Bit08)	
10	Input 10(wDigitalInp. Bit09)	
11	Input 11(wDigitalInp. Bit10)	
12	Input 12(wDigitalInp. Bit11)	
13	Input 13(wDigitalInp. Bit12)	
14	Input 14(wDigitalInp. Bit13)	
15	Input 15(wDigitalInp. Bit14)	
16	Input 16(wDigitalInp. Bit15)	
Subcodes	**Lenze setting**	**Information**
C01405/1	0: Waiting function deactivated	L_Sequencer_1: Pos. action 1: Start with
C01405/...		...
C01405/50		L_Sequencer_1: Pos. action 50: Start with
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01406

Parameter | Name: **C01406 | L_Sequencer_1: PSLevel**			Data type: UNSIGNED_8 Index: 23169$_d$ = 5A81$_h$
From version 12.00.00			▸ L_Sequencer_1: "Positioning" action
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Information**
C01406/1	0		L_Sequencer_1: Pos. action 1: input polarity
C01406/...			...
C01406/50			L_Sequencer_1: Pos. action 50: input polarity
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01407

Parameter | Name: **C01407 | L_Sequencer_1: PSProfil**			Data type: UNSIGNED_16 Index: 23168_d = $5A80_h$
From version 12.00.00			L_Sequencer_1: "Positioning" action
Setting range (min. value | unit | max. value)			
0		15	
Subcodes	**Lenze setting**		**Information**
C01407/1	0		L_Sequencer_1: Pos. action 1: Profile number
C01407/...			...
C01407/50			L_Sequencer_1: Pos. action 50: Profile number
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01408

Parameter | Name: **C01408 | L_Sequencer_1: PSStep**			Data type: UNSIGNED_16 Index: 23167_d = $5A7F_h$
From version 12.00.00			L_Sequencer_1: "Positioning" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01408/1	0		L_Sequencer_1: Pos. action 1: Jump destination
C01408/...			...
C01408/50			L_Sequencer_1: Pos. action 50: Jump destination
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01409

Parameter | Name: **C01409 | L_Sequencer_1: PS_WD_Time**			Data type: UNSIGNED_32 Index: 23166_d = $5A7E_h$
From version 12.00.00			L_Sequencer_1: "Positioning" action
Setting range (min. value | unit | max. value)			
0.000		2147480.000	
Subcodes	**Lenze setting**		**Information**
C01409/1	0.000		L_Sequencer_1: Pos. action 1: Monitoring time
C01409/...			...
C01409/50			L_Sequencer_1: Pos. action 50: Monitoring time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01410

Parameter | Name: **C01410 | L_Sequencer_1: PS_WD_Step**			Data type: UNSIGNED_16 Index: 23165_d = $5A7D_h$
From version 12.00.00			▸ L_Sequencer_1: "Positioning" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01410/1	0		L_Sequencer_1: Pos. action 1: Jump destination monit.
C01410/...			...
C01410/50			L_Sequencer_1: Pos. action 50: Jump destination monit.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01411

Parameter | Name: **C01411 | L_Sequencer_1: SwitchOut1**		Data type: UNSIGNED_16 Index: 23164_d = $5A7C_h$
From version 12.00.00		▸ L_Sequencer_1: "Switching" action
Selection list		
0	Deactivated	
1	Output 1 (wDigitalOutp. Bit00)	
2	Output 2 (wDigitalOutp. Bit01)	
3	Output 3 (wDigitalOutp. Bit02)	
4	Output 4 (wDigitalOutp. Bit03)	
5	Output 5 (wDigitalOutp. Bit04)	
6	Output 6 (wDigitalOutp. Bit05)	
7	Output 7 (wDigitalOutp. Bit06)	
8	Output 8 (wDigitalOutp. Bit07)	
9	Output 9 (wDigitalOutp. Bit08)	
10	Output 10 (wDigitalOutp. Bit09)	
11	Output 11 (wDigitalOutp. Bit10)	
12	Output 12 (wDigitalOutp. Bit11)	
13	Output 13 (wDigitalOutp. Bit12)	
14	Output 14 (wDigitalOutp. Bit13)	
15	Output 15 (wDigitalOutp. Bit14)	
16	Output 16 (wDigitalOutp. Bit15)	
Subcodes	**Lenze setting**	**Information**
C01411/1	0: Deactivated	L_Sequencer_1: Switch. act. 1: output switch. A
C01411/...		...
C01411/16		L_Sequencer_1: Switch. act. 16: output switch. A
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01412

Parameter | Name: **C01412 | L_Sequencer_1: SwitchLevel1**			Data type: UNSIGNED_8 Index: 23163_d = $5A7B_h$
From version 12.00.00			▸ L_Sequencer_1: "Switching" action
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Information**
C01412/1	0		L_Sequencer_1: Switch. act. 1: pol. switch. A
C01412/...			...
C01412/16			L_Sequencer_1: Switch. act. 16: pol. switch. A
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01413

Parameter | Name: **C01413 | L_Sequencer_1: SwitchOut2**		Data type: UNSIGNED_16 Index: 23162_d = $5A7A_h$
From version 12.00.00		▸ L_Sequencer_1: "Switching" action
Selection list		
0	Deactivated	
1	Output 1 (wDigitalOutp. Bit00)	
2	Output 2 (wDigitalOutp. Bit01)	
3	Output 3 (wDigitalOutp. Bit02)	
4	Output 4 (wDigitalOutp. Bit03)	
5	Output 5 (wDigitalOutp. Bit04)	
6	Output 6 (wDigitalOutp. Bit05)	
7	Output 7 (wDigitalOutp. Bit06)	
8	Output 8 (wDigitalOutp. Bit07)	
9	Output 9 (wDigitalOutp. Bit08)	
10	Output 10 (wDigitalOutp. Bit09)	
11	Output 11 (wDigitalOutp. Bit10)	
12	Output 12 (wDigitalOutp. Bit11)	
13	Output 13 (wDigitalOutp. Bit12)	
14	Output 14 (wDigitalOutp. Bit13)	
15	Output 15 (wDigitalOutp. Bit14)	
16	Output 16 (wDigitalOutp. Bit15)	
Subcodes	**Lenze setting**	**Information**
C01413/1	0: Deactivated	L_Sequencer_1: Switch. act. 1: output switch. B
C01413/...		...
C01413/16		L_Sequencer_1: Switch. act. 16: output switch. B
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01414

Parameter | Name: **C01414 | L_Sequencer_1: SwitchLevel2**			Data type: UNSIGNED_8 Index: 23161_d = $5A79_h$
From version 12.00.00			▸ L_Sequencer_1: "Switching" action
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Information**
C01414/1	0		L_Sequencer_1: Switch. act. 1: pol. switch. B
C01414/...			...
C01414/16			L_Sequencer_1: Switch. act. 16: pol. switch. B
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01415

Parameter | Name: **C01415 | L_Sequencer_1: BranchIn**		Data type: UNSIGNED_16 Index: 23160_d = $5A78_h$
From version 12.00.00		▸ L_Sequencer_1: "Branch" action
Selection list		
0	Unconditional branch	
1	Input 1(wDigitalInp. Bit00)	
2	Input 2(wDigitalInp. Bit01)	
3	Input 3(wDigitalInp. Bit02)	
4	Input 4(wDigitalInp. Bit03)	
5	Input 5(wDigitalInp. Bit04)	
6	Input 6(wDigitalInp. Bit05)	
7	Input 7(wDigitalInp. Bit06)	
8	Input 8(wDigitalInp. Bit07)	
9	Input 9(wDigitalInp. Bit08)	
10	Input 10(wDigitalInp. Bit09)	
11	Input 11(wDigitalInp. Bit10)	
12	Input 12(wDigitalInp. Bit11)	
13	Input 13(wDigitalInp. Bit12)	
14	Input 14(wDigitalInp. Bit13)	
15	Input 15(wDigitalInp. Bit14)	
16	Input 16(wDigitalInp. Bit15)	
Subcodes	**Lenze setting**	**Information**
C01415/1	0: Unconditional branch	L_Sequencer_1: Branch act. 1: Input for jump
C01415/...		...
C01415/16		L_Sequencer_1: Branch act. 16: Input for jump
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01416

Parameter | Name: **C01416 | L_Sequencer_1: BranchLevel**			Data type: UNSIGNED_8 Index: 23159_d = $5A77_h$
From version 12.00.00			L_Sequencer_1: "Branch" action
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Information**
C01416/1	0		L_Sequencer_1: Branch act. 1: Input polarity
C01416/...			...
C01416/16			L_Sequencer_1: Branch act. 16: Input polarity
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01417

Parameter | Name: **C01417 | L_Sequencer_1: BranchStep**			Data type: UNSIGNED_16 Index: 23158_d = $5A76_h$
From version 12.00.00			L_Sequencer_1: "Branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01417/1	0		L_Sequencer_1: Branch act.1: Jump destination
C01417/...			...
C01417/16			L_Sequencer_1: Branch act. 16: Jump destination
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01418

Parameter | Name: **C01418 | L_Sequencer_1: BranchStep1**			Data type: UNSIGNED_16 Index: 23157_d = $5A75_h$
From version 12.00.00			L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01418/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 1
C01418/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01419

Parameter | Name: C01419 | L_Sequencer_1: BranchStep2			Data type: UNSIGNED_16 Index: 23156_d = $5A74_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01419/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 2
C01419/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01420

Parameter | Name: C01420 | L_Sequencer_1: BranchStep3			Data type: UNSIGNED_16 Index: 23155_d = $5A73_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01420/1	0		L_Sequencer_1: var. branch1: jump destination 3
C01420/2	0		L_Sequencer_1: Var. branch2: Jump destin. 3
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01421

Parameter | Name: C01421 | L_Sequencer_1: BranchStep4			Data type: UNSIGNED_16 Index: 23154_d = $5A72_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01421/1	0		L_Sequencer_1: Var. branch1: Jump destin. 4
C01421/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01422

Parameter | Name: C01422 | L_Sequencer_1: BranchStep5			Data type: UNSIGNED_16 Index: 23153_d = $5A71_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01422/1	0		L_Sequencer_1: var. branch1: jump destination 5
C01422/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 5
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01423

Parameter | Name: C01423 | L_Sequencer_1: BranchStep6			Data type: UNSIGNED_16 Index: 23152$_d$ = 5A70$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01423/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 6
C01423/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 6
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01424

Parameter | Name: C01424 | L_Sequencer_1: BranchStep7			Data type: UNSIGNED_16 Index: 23151$_d$ = 5A6F$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01424/1	0		L_Sequencer_1: Var. branch1: Jump destin. 7
C01424/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 7
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01425

Parameter | Name: C01425 | L_Sequencer_1: BranchStep8			Data type: UNSIGNED_16 Index: 23150$_d$ = 5A6E$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01425/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 8
C01425/2	0		L_Sequencer_1: Var. Verzw.2: Sprungziel 8
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01426

Parameter | Name: C01426 | L_Sequencer_1: BranchStep9			Data type: UNSIGNED_16 Index: 23149$_d$ = 5A6D$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01426/1	0		L_Sequencer_1: var. branch1: jump destination 9
C01426/2	0		L_Sequencer_1: Var. branch2: Jump destination 9
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01427

Parameter | Name: **C01427 | L_Sequencer_1: BranchStep10**			Data type: UNSIGNED_16 Index: 23148_d = $5A6C_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01427/1	0		L_Sequencer_1: Var. branch1: Jump destin. 10
C01427/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 10
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01428

Parameter | Name: **C01428 | L_Sequencer_1: BranchStep11**			Data type: UNSIGNED_16 Index: 23147_d = $5A6B_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01428/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 11
C01428/2	0		L_Sequencer_1: var. branch2: jump destination 11
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01429

Parameter | Name: **C01429 | L_Sequencer_1: BranchStep12**			Data type: UNSIGNED_16 Index: 23146_d = $5A6A_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01429/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 12
C01429/2	0		L_Sequencer_1: Var. branch2: Jump destin. 12
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01430

Parameter | Name: **C01430 | L_Sequencer_1: BranchStep13**			Data type: UNSIGNED_16 Index: 23145_d = $5A69_h$
From version 12.00.00			▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01430/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 13
C01430/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 13
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01431

Parameter | Name: C01431 | L_Sequencer_1: BranchStep14			Data type: UNSIGNED_16 Index: 23144$_d$ = 5A68$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01431/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 14
C01431/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 14
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01432

Parameter | Name: C01432 | L_Sequencer_1: BranchStep15			Data type: UNSIGNED_16 Index: 23143$_d$ = 5A67$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01432/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 15
C01432/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 15
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01433

Parameter | Name: C01433 | L_Sequencer_1: BranchStep16			Data type: UNSIGNED_16 Index: 23142$_d$ = 5A66$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01433/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 16
C01433/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 16
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01434

Parameter | Name: C01434 | L_Sequencer_1: BranchStep17			Data type: UNSIGNED_16 Index: 23141$_d$ = 5A65$_h$
From version 12.00.00			▶ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01434/1	0		L_Sequencer_1: Var. branch 1: Jump destin. 17
C01434/2	0		L_Sequencer_1: Var. branch 2: Jump destin. 17
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01435

Parameter | Name: C01435 | L_Sequencer_1: BranchStep18		Data type: UNSIGNED_16 Index: 23140_d = $5A64_h$
From version 12.00.00		▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)		
0		100
Subcodes	**Lenze setting**	**Information**
C01435/1	0	L_Sequencer_1: Var. branch 1: Jump destin. 18
C01435/2	0	L_Sequencer_1: Var. branch 2: Jump destin. 18
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01436

Parameter | Name: C01436 | L_Sequencer_1: BranchStep19		Data type: UNSIGNED_16 Index: 23139_d = $5A63_h$
From version 12.00.00		▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)		
0		100
Subcodes	**Lenze setting**	**Information**
C01436/1	0	L_Sequencer_1: Var. branch 1: Jump destin. 19
C01436/2	0	L_Sequencer_1: var. branch2: jump destination 19
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01437

Parameter | Name: C01437 | L_Sequencer_1: BranchStep20		Data type: UNSIGNED_16 Index: 23138_d = $5A62_h$
From version 12.00.00		▸ L_Sequencer_1: "Variable branch" action
Setting range (min. value | unit | max. value)		
0		100
Subcodes	**Lenze setting**	**Information**
C01437/1	0	L_Sequencer_1: Var. branch 1: Jump destin. 20
C01437/2	0	L_Sequencer_1: Var. branch 2: Jump destin. 20
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01438

Parameter | Name: **C01438 | L_Sequencer_1: Wartezeit**			Data type: UNSIGNED_32 Index: 23137_d = $5A61_h$
From version 12.00.00			▸ L_Sequencer_1: "Waiting" action
Setting range (min. value | unit | max. value)			
0.000	s	2127480.000	
Subcodes	**Lenze setting**	**Information**	
C01438/1	0.000 s	L_Sequencer_1: Wait. act. 1: Waiting time	
C01438/...		...	
C01438/8		L_Sequencer_1: Wait. act. 8: Waiting time	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01439

Parameter | Name: **C01439 | L_Sequencer_1: Wait Eingang f. Weiter**		Data type: UNSIGNED_16 Index: 23136_d = $5A60_h$
From version 12.00.00		▸ L_Sequencer_1: "Waiting" action
Selection list		
0	Input deactivated	
1	Input 1(wDigitalInp. Bit00)	
2	Input 2(wDigitalInp. Bit01)	
3	Input 3(wDigitalInp. Bit02)	
4	Input 4(wDigitalInp. Bit03)	
5	Input 5(wDigitalInp. Bit04)	
6	Input 6(wDigitalInp. Bit05)	
7	Input 7(wDigitalInp. Bit06)	
8	Input 8(wDigitalInp. Bit07)	
9	Input 9(wDigitalInp. Bit08)	
10	Input 10(wDigitalInp. Bit09)	
11	Input 11(wDigitalInp. Bit10)	
12	Input 12(wDigitalInp. Bit11)	
13	Input 13(wDigitalInp. Bit12)	
14	Input 14(wDigitalInp. Bit13)	
15	Input 15(wDigitalInp. Bit14)	
16	Input 16(wDigitalInp. Bit15)	
Subcodes	**Lenze setting**	**Information**
C01439/1	0: Input deactivated	L_Sequencer_1: Wait. act. 1: Input for "Next"
C01439/...		...
C01439/8		L_Sequencer_1: Wait. act. 8: Input for "Next"
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01440

Parameter | Name: **C01440 | L_Sequencer_1: WaitLevel**			Data type: UNSIGNED_8 Index: 23135_d = $5A5F_h$
From version 12.00.00			▸ L_Sequencer_1: "Waiting" action
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Information**
C01440/1	0		L_Sequencer_1: Wait. act. 1: Input polarity
C01440/...			...
C01440/8			L_Sequencer_1: Wait. act. 8: Input polarity
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01441

Parameter | Name: **C01441 | L_Sequencer_1: SetZählernummer**			Data type: UNSIGNED_16 Index: 23134_d = $5A5E_h$
From version 12.00.00			▸ L_Sequencer_1: "Counter setting" action
Setting range (min. value | unit | max. value)			
0		5	
Subcodes	**Lenze setting**		**Information**
C01441/1	1		L_Sequencer_1: Counter set act. 1: Counter no.
C01441/...			...
C01441/5			L_Sequencer_1: Counter set act. 5: Counter no.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01442

Parameter | Name: **C01442 | L_Sequencer_1: SetZählerStartwert**			Data type: INTEGER_32 Index: 23133_d = $5A5D_h$
From version 12.00.00			▸ L_Sequencer_1: "Counter setting" action
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Information**
C01442/1	0		L_Sequencer_1: Counter set act. 1: Starting value
C01442/...			...
C01442/5			L_Sequencer_1: Counter set act. 5: Starting value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01443

Parameter | Name: **C01443 | L_Sequencer_1: Aktueller Zählerstand**			Data type: INTEGER_32 Index: 23132$_d$ = 5A5C$_h$
From version 12.00.00			▸ L_Sequencer_1: "Counter setting" action
Display range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes			**Information**
C01443/1			L_Sequencer_1: Counter 1: Current counter content
C01443/...			...
C01443/5			L_Sequencer_1: Counter 5: Current counter content
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01444

Parameter | Name: **C01444 | L_Sequencer_1: ZählerNummer**			Data type: UNSIGNED_16 Index: 23131$_d$ = 5A5B$_h$
From version 12.00.00			▸ L_Sequencer_1: "Counting" action
Setting range (min. value | unit | max. value)			
0		5	
Subcodes	**Lenze setting**		**Information**
C01444/1	1		L_Sequencer_1: Count. act. 1: Counter no.
C01444/...			...
C01444/8			L_Sequencer_1: Count. act. 8: Counter no.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01445

Parameter | Name: **C01445 | L_Sequencer_1: SchrittwertZähler**			Data type: INTEGER_32 Index: 23130$_d$ = 5A5A$_h$
From version 12.00.00			▸ L_Sequencer_1: "Counting" action
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Information**
C01445/1	1		L_Sequencer_1: Count. act. 1: Step value
C01445/...			...
C01445/8			L_Sequencer_1: Count. act. 8: Step value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01446

Parameter | Name: **C01446 | L_Sequencer_1: ZählerVergleichswert**			Data type: INTEGER_32 Index: 23129_d = $5A59_h$
From version 12.00.00			▸ L_Sequencer_1: "Counting" action
Setting range (min. value | unit | max. value)			
-2147483647		2147483647	
Subcodes	**Lenze setting**		**Information**
C01446/1	0		L_Sequencer_1: Count. act. 1: comparison value
C01446/...			...
C01446/8			L_Sequencer_1: Count. act. 8: comparison value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01447

Parameter | Name: **C01447 | L_Sequencer_1: ZählerSprungziel**			Data type: UNSIGNED_16 Index: 23128_d = $5A58_h$
From version 12.00.00			▸ L_Sequencer_1: "Counting" action
Setting range (min. value | unit | max. value)			
0		100	
Subcodes	**Lenze setting**		**Information**
C01447/1	0		L_Sequencer_1: Count. act. 1: Jump destination
C01447/...			...
C01447/8			L_Sequencer_1: Count. act. 8: Jump destination
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01448

Parameter | Name: **C01448 | L_Sequencer_1: ZählerVergleichsfunktion**		Data type: UNSIGNED_16 Index: 23127_d = $5A57_h$
From version 12.00.00		▸ L_Sequencer_1: "Counting" action
Selection list		
1	Counter content = comparison value	
2	Counter content > comparison value	
3	Counter content >= comparison value	
4	Counter content < comparison value	
5	Counter content <= comparison value	
Subcodes	**Lenze setting**	**Information**
C01448/1	1: Counter content = comparison value	L_Sequencer_1: Count. act. 1: Comparison op.
C01448/...		...
C01448/8		L_Sequencer_1: Count. act. 8: Comaprison op.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01449

Parameter | Name: **C01449 | L_Sequencer_1: Standby Eingang f. Ende**		Data type: UNSIGNED_16 Index: 23126_d = $5A56_h$
From version 12.00.00		▶ L_Sequencer_1: "Standby" action
Selection list		
1	Input 1(wDigitalInp. Bit00)	
2	Input 2(wDigitalInp. Bit01)	
3	Input 3(wDigitalInp. Bit02)	
4	Input 4(wDigitalInp. Bit03)	
5	Input 5(wDigitalInp. Bit04)	
6	Input 6(wDigitalInp. Bit05)	
7	Input 7(wDigitalInp. Bit06)	
8	Input 8(wDigitalInp. Bit07)	
9	Input 9(wDigitalInp. Bit08)	
10	Input 10(wDigitalInp. Bit09)	
11	Input 11(wDigitalInp. Bit10)	
12	Input 12(wDigitalInp. Bit11)	
13	Input 13(wDigitalInp. Bit12)	
14	Input 14(wDigitalInp. Bit13)	
15	Input 15(wDigitalInp. Bit14)	
16	Input 16(wDigitalInp. Bit15)	
Subcodes	**Lenze setting**	**Information**
C01449/1	1: Input 1(wDigitalInp. Bit00)	L_Sequencer_1: Standby act. 1: Input for "End"
C01449/...		...
C01449/5		L_Sequencer_1: Standby act. 5: Input for "End"
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01450

Parameter | Name: **C01450 | L_Sequencer_1: Standby PolaritätEingang**			Data type: UNSIGNED_8 Index: 23125_d = $5A55_h$
From version 12.00.00			▶ L_Sequencer_1: "Standby" action
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Information**
C01450/1	0		L_Sequencer_1: Standby act. 1: Input polarity
C01450/...			...
C01450/5			L_Sequencer_1: Standby act. 5: Input polarity
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01451

Parameter | Name: C01451 | L_Sequencer_1: Standby Modusfolger		Data type: UNSIGNED_16 Index: 23124_d = $5A54_h$
From version 12.00.00		▶ L_Sequencer_1: "Standby" action
Selection list		
0	Speed follower	
1	Position follower	
Subcodes	**Lenze setting**	**Information**
C01451/1	0: Speed follower	L_Sequencer_1: Standby act. 1: Follower mode
C01451/...		...
C01451/5		L_Sequencer_1: Standby act. 5: Follower mode
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01452

Parameter | Name: C01452 | L_Sequencer_1: StandbySollwert			Data type: INTEGER_16 Index: 23123_d = $5A53_h$
From version 12.00.00			▶ L_Sequencer_1: "Standby" action
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Information**
C01452/1	0.00 %		L_Sequencer_1: Standby act. 1: Setpoint
C01452/...			...
C01452/5			L_Sequencer_1: Standby act. 5: Setpoint
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01453

Parameter | Name: **C01453 | L_Sequencer_1: Bool signal**			Data type: UNSIGNED_8 Index: 23122_d = $5A52_h$
From version 12.00.00			▸ L_Sequencer_1 (sequence control)
Display range (min. value | unit | max. value)			
0		1	
Subcodes			**Information**
C01453/1			L_Sequencer_1: bStart
C01453/2			L_Sequencer_1: bPause
C01453/3			L_Sequencer_1: bNextStep
C01453/4			L_Sequencer_1: bCancel
C01453/5			L_Sequencer_1: bBreak
C01453/6			L_Sequencer_1: bReset
C01453/7			L_Sequencer_1: bProgramBusy
C01453/8			L_Sequencer_1: bStateReady
C01453/9			L_Sequencer_1: bStateRun
C01453/10			L_Sequencer_1: bStatePause
C01453/11			L_Sequencer_1: bStateBreak
C01453/12			L_Sequencer_1: bStateReset
C01453/13			L_Sequencer_1: bStateDone
C01453/14			L_Sequencer_1: bWatchDogAcitve
C01453/15			L_Sequencer_1: bErrBranch
C01453/16			L_Sequencer_1: bErrActive
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01454

Parameter | Name: **C01454 | L_Sequencer_1: INT signal**			Data type: INTEGER_16 Index: 23121_d = $5A51_h$
From version 12.00.00			▸ L_Sequencer_1 (sequence control)
Display range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes			**Information**
C01454/1			L_Sequencer_1: nSet_a
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01455

Parameter | Name: **C01455 | L_Sequencer_1: Word signal**			Data type: UNSIGNED_16 Index: 23120_d = $5A50_h$
From version 12.00.00			▶ L_Sequencer_1 (sequence control)
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Information**
C01455/1			L_Sequencer_1: wStartStep
C01455/2			L_Sequencer_1: wBranch1
C01455/3			L_Sequencer_1: wBranch2
C01455/4			L_Sequencer_1: wDigitalInputs
C01455/5			L_Sequencer_1: wMotionState1
C01455/6			L_Sequencer_1: wMotionState2
C01455/7			L_Sequencer_1: wMckPosCtrl1
C01455/8			L_Sequencer_1: wMckPosCtrl2
C01455/9			L_Sequencer_1: wAuxCtrl
C01455/10			L_Sequencer_1: wState
C01455/11			L_Sequencer_1: wActStep
C01455/12			L_Sequencer_1: wDigitalOutputs
C01455/13			L_Sequencer_1: wProfileNumber
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01460

Parameter | Name: **C01460 | L_ConvActPos: TConst**			Data type: UNSIGNED_16 Index: 23115_d = $5A4B_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0	ms	1000	
Subcodes	**Lenze setting**		**Information**
C01460/1	10 ms		L_ConvActPos_1: TConst
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01461

Parameter | Name: **C01461 | L_ConvActPos: MaxPos/MinPos**			Data type: INTEGER_16 Index: 23114_d = $5A4A_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Information**
C01461/1	100.00 %		L_ConvActPos_1: MaxPos
C01461/2	-100.00 %		L_ConvActPos_1: MinPos
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01462

Parameter | Name: **C01462 | L_ConvActPos: Length**			Data type: UNSIGNED_32 Index: 23113_d = $5A49_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0	mm	100000	
Subcodes	**Lenze setting**		**Information**
C01462/1	0 mm		L_ConvActPos_1: Length
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01463

Parameter | Name: **C01463 | L_ConvActPos: UpperRef/LowerRef**			Data type: INTEGER_16 Index: 23112_d = $5A48_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Information**
C01463/1	90.00 %		L_ConvActPos_1: UpperRef
C01463/2	-90.01 %		L_ConvActPos_1: LowerRef
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01464

Parameter | Name: **C01464 | L_ConvActPos: WindowSetPos**			Data type: INTEGER_16 Index: 23111_d = $5A47_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Information**
C01464/1	10.00 %		L_ConvActPos_1: WindowSetPos
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01465

Parameter | Name: **C01465 | L_ConvActPos: DelayTime**			Data type: UNSIGNED_16 Index: 23110_d = $5A46_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0.000	s	50.000	
Subcodes	**Lenze setting**		**Information**
C01465/1	0.100 s		L_ConvActPos_1: DelayTime
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01466

Parameter | Name: **C01466 | L_ConvActPos: TConstVAdd**			Data type: UNSIGNED_16 Index: 23109_d = $5A45_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0	ms	1000	
Subcodes	**Lenze setting**		**Information**
C01466/1	10 ms		L_ConvActPos_1: TConstVAdd
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01467

Parameter | Name: **C01467 | L_ConvActPos: DeadBand**			Data type: INTEGER_16 Index: 23108_d = $5A44_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Information**
C01467/1	1.00 %		L_ConvActPos_1: DeadBand
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01468

Parameter | Name: **C01468 | L_ConvActPos: Damping**			Data type: UNSIGNED_16 Index: 23107_d = $5A43_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0.00		10.00	
Subcodes	**Lenze setting**		**Information**
C01468/1	0.00		L_ConvActPos_1: Damping
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C01469

Parameter | Name: **C01469 | L_MFail: Vp**			Data type: UNSIGNED_16 Index: 23106_d = $5A42_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0.001		31.000	
Subcodes	**Lenze setting**		**Info**
C01469/1	1.000		L_MFail_1: Vp
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C01470

Parameter | Name: **C01470 | L_MFail: Tn**			Data type: UNSIGNED_16 Index: 23105_d = $5A41_h$
From version 12.00.00			
Setting range (min. value | unit | max. value)			
0	ms	60000	
Subcodes	**Lenze setting**		**Info**
C01470/1	20 ms		L_MFail_1: U DC-bus controller reset time
C01470/2	20 ms		L_MFail_1: Acceleration time
C01470/3	20 ms		L_MFail_1: Retrigger time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01501

Parameter | Name: **C01501 | Resp. to communication error with MCI**		Data type: UNSIGNED_8 Index: 23074_d = $5A22_h$
Configuration of monitoring functions for the communication module		
Selection list		
0	No Reaction	
1	Fault	
2	Trouble	
3	TroubleQuickStop	
4	WarningLocked	
5	Warning	
6	Information	
Subcodes	**Lenze setting**	**Info**
C01501/1	0: No Reaction	Resp. to MCI connection error • Response to a communication error of the attached communication module.
C01501/2	0: No Reaction	Resp. to MCI invalid module • Response to an unplugged or incompatible communication module
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01670

Parameter | Name: **C01670 | L_ComparePhi 1-5: Function**		Data type: UNSIGNED_8 Index: 22905_d = 5979_h
Selection of comparison operation • If the statement of the selected comparison operation is true, the binary *bOut* output will be set to TRUE.		
Selection list		
1	In1 = In2	
2	In1 > In2	
3	In1 < In2	
4	|In1| = |In2|	
5	|In1| > |In2|	
6	|In1| < |In2|	
Subcodes	**Lenze setting**	**Info**
C01670/1	1: In1 = In2	L_ComparePhi_1: Function
C01670/2	1: In1 = In2	L_ComparePhi_2: Function
C01670/3	1: In1 = In2	L_ComparePhi_3: Function
C01670/4	1: In1 = In2	L_ComparePhi_4: Function
C01670/5	1: In1 = In2	L_ComparePhi_5: Function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01671

Parameter | Name: **C01671 | L_ComparePhi 1-5: Hysteresis**			Data type: INTEGER_32 Index: 22904_d = 5978_h
Hysteresis for the comparison function			
Setting range (min. value | unit | max. value)			
0	Incr.	1073741824	
Subcodes	**Lenze setting**		**Info**
C01671/1	0 incr.		L_ComparePhi_1: Hysteresis
C01671/2	0 incr.		L_ComparePhi_2: Hysteresis
C01671/3	0 incr.		L_ComparePhi_3: Hysteresis
C01671/4	0 incr.		L_ComparePhi_4: Hysteresis
C01671/5	0 incr.		L_ComparePhi_5: Hysteresis
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01672

Parameter | Name: C01672 | L_ComparePhi 1-5: Window		Data type: INTEGER_32 Index: 22903d = 5977h
Window for the comparison operation		
Setting range (min. value | unit | max. value)		
0	Incr.	1073741824
Subcodes	**Lenze setting**	**Info**
C01672/1	0 incr.	L_ComparePhi_1: Window
C01672/2	0 incr.	L_ComparePhi_2: Window
C01672/3	0 incr.	L_ComparePhi_3: Window
C01672/4	0 incr.	L_ComparePhi_4: Window
C01672/5	0 incr.	L_ComparePhi_5: Window
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01700

Parameter | Name: C01700 | Energy saving mode: Mode			Data type: UNSIGNED_8 Index: 22875d = 595Bh
From version 17.00.00 onwards Display of the energy saving modes maximally provided			
Display range (min. value | unit | max. value)			
0		1	
Subcodes			**Info**
C01700/1			Number of energy saving modes maximally provided = 1
C01700/2			Current mode • 1 ≡ Energy saving mode is active • 0 ≡ Energy saving mode is not active
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01701

Parameter | Name: C01701 | Energy saving mode: toff min		Data type: UNSIGNED_16 Index: 22874d = 595Ah
From version 17.00.00 onwards Minimum time for which the inverter is to remain in the energy saving mode (TPm: Time Pause min).		
Setting range (min. value | unit | max. value)		
0	s	65535
Subcodes	**Lenze setting**	**Info**
C01701/1	0 s	Energy saving mode 1: toff min
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C01702

Parameter | Name: C01702 | Energy saving mode: toff			Data type: UNSIGNED_16 Index: 22873_d = 5959_h
From version 17.00.00 onwards Time until the energy saving mode is entered (TtP: Time to Pause) If the quick stop energy saving function is to be used, this time always has to be set to a greater value than the maximum time required for braking via the quickstop function.			
Setting range (min. value | unit | max. value)			
0	s	65535	
Subcodes	**Lenze setting**		**Info**
C01702/1	0 s		Energy saving mode 1: toff
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01703

Parameter | Name: C01703 | Energy saving mode: ton			Data type: UNSIGNED_16 Index: 22872_d = 5958_h
From version 17.00.00 onwards Time for exiting the energy saving mode (TtO: Time to Operate).			
Setting range (min. value | unit | max. value)			
0	s	65535	
Subcodes	**Lenze setting**		**Info**
C01703/1	0 s		Energy saving mode 1: ton
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01704

Parameter | Name: **C01704 | Energy saving mode: Function**			Data type: UNSIGNED_16 Index: 22871_d = 5957_h
From version 17.00.00 onwards Response of the device in energy saving mode			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	CINH		With entering the energy saving mode, controller inhibit is set. Controller inhibit is reset when the "Pause-End" command is received. • Diagnostics of active controller inhibit with C00158/0, bit 9 • Display: "Energy saving mode"
Bit 1	QSP		When the "Pause-Req" command is received, the inverter executes a quick stop. • Quick stop is cancelled when the "Pause-End" command has been accepted. • Quick stop diagnostics with C00159/0, bit 9 • Display: "Energy saving mode"
Bit 2	Dimming the LEDs		Most of the LEDs of the inverter are switched off or their lightling intensity is reduced. • Only a few LEDs remain active in order to check the vitality of the inverter, e.g. the "DRV-RDY" LED.
Bit 3	Reserved		
Bit 4	Decoupling the IOs		The digital output terminals are decoupled from the application (FB Editor). The output levels for these outputs can be defined via the decoupling values. In order to render the decoupling values effective, decoupling must be activated via the following parameters: • Digital outputs: C00447/0 and C00448/0
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Off		Deactivate energy saving mode.
Subcodes	**Lenze setting**		**Info**
C01704/1	0x0000		Energy saving mode: components to be switched off.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C01709

Parameter | Name: **C01709 | Energy saving mode: Status**			Data type: UNSIGNED_8 Index: 22866_d = 5952_h
From version 17.00.00 onwards			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C01709/1			Energy saving mode: Status
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01751

Parameter | Name: **C01751 | Service code inverter characteristic**	Data type: UNSIGNED_8 Index: 22824_d = 5928_h
This code is used device-internally and must not be written by the user side!	

C01752

Parameter | Name: **C01752 | Service par. inverter charact. function**	Data type: UNSIGNED_8 Index: 22823_d = 5927_h
This code is used device-internally and must not be written by the user side!	

C01755

Parameter | Name: **C01755 | Service par. inverter charact. factor**	Data type: INTEGER_16 Index: 22820_d = 5924_h
This code is used device-internally and must not be written by the user side!	

C01763

Parameter | Name: **C01763 | Service code -clamp threshold**	Data type: INTEGER_16 Index: 22812_d = $591C_h$
This code is used device-internally and must not be written by the user side!	

C01764

Parameter | Name: **C01764 | Service par. clamp time**	Data type: UNSIGNED_8 Index: 22811_d = $591B_h$
This code is used device-internally and must not be written by the user side!	

C01765

Parameter | Name: **C01765 | Service code - difference threshold UG**	Data type: UNSIGNED_16 Index: 22810_d = $591A_h$
This code is used device-internally and must not be written by the user side!	

C01770

Parameter | Name: **C01770 | Filter time - earth-fault detect. is running**			Data type: UNSIGNED_8 Index: 22805_d = 5915_h
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	250	**2 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C01771

Parameter | Name: **C01771 | BU-Osc: Cycle time selection**	Data type: UNSIGNED_8 Index: 22804_d = 5914_h
This code is used device-internally and must not be written by the user side!	

C01772

Parameter | Name: **C01772 | BU-Osc: Trigger channel selection**	Data type: UNSIGNED_8 Index: 22803_d = 5913_h
This code is used device-internally and must not be written by the user side!	

C01773

Parameter | Name: **C01773 | BU-Osc: Trigger command**	Data type: UNSIGNED_8 Index: 22802_d = 5912_h
This code is used device-internally and must not be written by the user side!	

C01774

Parameter | Name: **C01774 | BU-Osc: Trigger mode**	Data type: UNSIGNED_8 Index: 22801_d = 5911_h
This code is used device-internally and must not be written by the user side!	

C01775

Parameter | Name: **C01775 | BU-Osc: Trigger**	Data type: INTEGER_16 Index: 22800_d = 5910_h
This code is used device-internally and must not be written by the user side!	

C01902

Parameter | Name: **C01902 | Diagnostics X6: Max. baud rate**	Data type: UNSIGNED_16 Index: 22673_d = 5891_h
Maximally permissible baud rate in the standard device after determination of the baud rate at the diagnostic interface X6	
Selection list(Lenze setting printed in bold)	
192	19.200 Bd
384	38.400 Bd
576	57.600 Bd
750	75.000 Bd
751	75.000 Bd ext.
1152	115.200 Bd
1500	150.000 Bd
2500	250.000 Bd
3750	375.000 Bd
7500	**750.000 Bd**
7501	750.000 Bd ext.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1	

C01903

Parameter | Name: **C01903 | Diagnostics X6: Change baud rate**	Data type: UNSIGNED_8 Index: 22672_d = 5890_h
New baud rate determination at the diagnostic interface X6	
Selection list(Lenze setting printed in bold)	
0	**Ignore changes**
1	Negotiate baud rate
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1	

C01905

Parameter | Name: **C01905 | Diagnostics X6: Current baud rate**		Data type: UNSIGNED_32 Index: 22670_d = $588E_h$
Current baud rate at the diagnostic interface X6		
Display range (min. value | unit | max. value)		
0	Bd	3000000
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C02200

Parameter | Name: **C02200 | LS_WriteParamList: Function**		Data type: UNSIGNED_8 Index: 22375_d = 5767_h
From version 12.00.00 Configuration of which parameter lists are to be subject to the "Parameter change-over" function.		
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C02200/1	1: On	Definable parameter list
C02200/2	0: Off	Motor data parameter list
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C02210

Parameter | Name: **C02210 | LS_WriteParamList: Motor control**		Data type: UNSIGNED_8 Index: 22365_d = $575D_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
1	SC: Servo control PSM	
2	SC: Servo control ASM	
3	SLPSM: Sensorless PSM	
4	SLVC: Vector control	
6	VFCplus: V/f linear	
7	VFCplus: V/f linear + encoder	
8	VFCplus: V/f quadr	
9	VFCplus: V/f quadr + encoder	
10	VFCplus: V/f definable	
11	VFCplusEco: V/f energy-saving	
Subcodes	**Lenze setting**	**Info**
C02210/1	6: VFCplus: V/f linear	LS_WriteParamList: c6_val1
C02210/2	6: VFCplus: V/f linear	LS_WriteParamList: c6_val2
C02210/3	6: VFCplus: V/f linear	LS_WriteParamList: c6_val3
C02210/4	6: VFCplus: V/f linear	LS_WriteParamList: c6_val3
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02212

Parameter | Name: **C02212 | LS_WriteParamList: VFC: V/f base frequency**			Data type: UNSIGNED_16 Index: 22363_d = $575B_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
7.5	Hz	2600.0	
Subcodes	**Lenze setting**		**Info**
C02212/1	50.0 Hz		LS_WriteParamList: c15_val1
C02212/2	50.0 Hz		LS_WriteParamList: c15_val2
C02212/3	50.0 Hz		LS_WriteParamList: c15_val3
C02212/4	50.0 Hz		LS_WriteParamList: c15_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02213

Parameter | Name: **C02213 | LS_WriteParamList: VFC: Vmin boost**			Data type: UNSIGNED_16 Index: 22362_d = $575A_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C02213/1	1.60 %		LS_WriteParamList: c16_val1
C02213/2	1.60 %		LS_WriteParamList: c16_val2
C02213/3	1.60 %		LS_WriteParamList: c16_val3
C02213/4	1.60 %		LS_WriteParamList: c16_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02214

Parameter | Name: **C02214 | LS_WriteParamList: Switching frequency**		Data type: UNSIGNED_8 Index: 22361_d = 5759_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
1	4 kHz var./drive-optimised	
2	8 kHz var./drive-optimised	
3	16 kHz var./drive-optimised	
5	2 kHz constant/drive-optimised	
6	4 kHz constant/drive-optimised	
7	8 kHz constant/drive-optimised	
8	16 kHz constant/drive-optimised	
11	4 kHz var./min. Pv	
12	8 kHz var./min. Pv	
13	16 kHz var./min. Pv	
15	2 kHz constant/min. Pv	
16	4 kHz constant/min. Pv	
17	8 kHz constant/min. Pv	
18	16 kHz constant/min. Pv	
21	8 kHz var./drive-opt./4 kHz min	
22	16 kHz var./drive-opt./4 kHz min	
23	16 kHz var./drive-opt./8 kHz min	
31	8 kHz var./min. Pv/4 kHz min	
32	16 kHz var./min. Pv/4 kHz min	
33	16 kHz var./min. Pv/8 kHz min	
Subcodes	**Lenze setting**	**Info**
C02214/1	2: 8 kHz var./drive-opt.	LS_WriteParamList: c18_val1
C02214/2	2: 8 kHz var./drive-opt.	LS_WriteParamList: c18_val2
C02214/3	2: 8 kHz var./drive-opt.	LS_WriteParamList: c18_val3
C02214/4	2: 8 kHz var./drive-opt.	LS_WriteParamList: c18_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02215

Parameter | Name: **C02215 | LS_WriteParamList: Auto-DCB: Threshold**			Data type: UNSIGNED_16 Index: 22360_d = 5758_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0	rpm	60000	
Subcodes	**Lenze setting**		**Info**
C02215/1	3 rpm		LS_WriteParamList: c19_val1
C02215/2	3 rpm		LS_WriteParamList: c19_val2
C02215/3	3 rpm		LS_WriteParamList: c19_val3
C02215/4	3 rpm		LS_WriteParamList: c19_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02216

Parameter | Name: **C02216 | LS_WriteParamList: Slip compensation**		Data type: INTEGER_16 Index: 22359_d = 5757_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
-100.00	%	100.00
Subcodes	**Lenze setting**	**Info**
C02216/1	2.67 %	LS_WriteParamList: c21_val1
C02216/2	2.67 %	LS_WriteParamList: c21_val2
C02216/3	2.67 %	LS_WriteParamList: c21_val3
C02216/4	2.67 %	LS_WriteParamList: c21_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02217

Parameter | Name: **C02217 | LS_WriteParamList: Imax in motor mode**		Data type: UNSIGNED_16 Index: 22358_d = 5756_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.00	A	655.35
Subcodes	**Lenze setting**	**Info**
C02217/1	47.00 A	LS_WriteParamList: c22_val1
C02217/2	47.00 A	LS_WriteParamList: c22_val2
C02217/3	47.00 A	LS_WriteParamList: c22_val3
C02217/4	47.00 A	LS_WriteParamList: c22_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02218

Parameter | Name: **C02218 | LS_WriteParamList: Imax in generator mode**		Data type: INTEGER_16 Index: 22357_d = 5755_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.00	%	100.00
Subcodes	**Lenze setting**	**Info**
C02218/1	100.00 %	LS_WriteParamList: c23_val1
C02218/2	100.00 %	LS_WriteParamList: c23_val2
C02218/3	100.00 %	LS_WriteParamList: c23_val3
C02218/4	100.00 %	LS_WriteParamList: c23_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02219

Parameter | Name: **C02219 | LS_WriteParamList: DC braking: Current**			Data type: INTEGER_16 Index: 22356_d = 5754_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	200.00	
Subcodes	**Lenze setting**		**Info**
C02219/1	50.00 %		LS_WriteParamList: c36_val1
C02219/2	50.00 %		LS_WriteParamList: c36_val2
C02219/3	50.00 %		LS_WriteParamList: c36_val3
C02219/4	50.00 %		LS_WriteParamList: c36_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02220

Parameter | Name: **C02220 | LS_WriteParamList: Vp speed controller**			Data type: UNSIGNED_16 Index: 22355_d = 5753_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00		600.00	
Subcodes	**Lenze setting**		**Info**
C02220/1	15.00		LS_WriteParamList: c70_s1_val1
C02220/2	15.00		LS_WriteParamList: c70_s1_val2
C02220/3	15.00		LS_WriteParamList: c70_s1_val3
C02220/4	15.00		LS_WriteParamList: c70_s1_val4
C02220/5	6.00		LS_WriteParamList: c70_s2_val1
C02220/6	6.00		LS_WriteParamList: c70_s2_val2
C02220/7	6.00		LS_WriteParamList: c70_s2_val3
C02220/8	6.00		LS_WriteParamList: c70_s2_val4
C02220/9	3.00		LS_WriteParamList: c70_s3_val1
C02220/10	3.00		LS_WriteParamList: c70_s3_val2
C02220/11	3.00		LS_WriteParamList: c70_s3_val3
C02220/12	3.00		LS_WriteParamList: c70_s3_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02221

Parameter | Name: C02221 | LS_WriteParamList: Ti speed controller		Data type: UNSIGNED_16 Index: 22354_d = 5752_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.0	ms	6000.0
Subcodes	**Lenze setting**	**Info**
C02221/1	100.0 ms	LS_WriteParamList: c71_s1_val1
C02221/2	100.0 ms	LS_WriteParamList: c71_s1_val2
C02221/3	100.0 ms	LS_WriteParamList: c71_s1_val3
C02221/4	100.0 ms	LS_WriteParamList: c71_s1_val4
C02221/5	50.0 ms	LS_WriteParamList: c71_s2_val1
C02221/6	50.0 ms	LS_WriteParamList: c71_s2_val2
C02221/7	50.0 ms	LS_WriteParamList: c71_s2_val3
C02221/8	50.0 ms	LS_WriteParamList: c71_s2_val4
C02221/9	50.0 ms	LS_WriteParamList: c71_s3_val1
C02221/10	50.0 ms	LS_WriteParamList: c71_s3_val2
C02221/11	50.0 ms	LS_WriteParamList: c71_s3_val3
C02221/12	50.0 ms	LS_WriteParamList: c71_s3_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10		

C02222

Parameter | Name: C02222 | LS_WriteParamList: SC: Tdn speed controller		Data type: UNSIGNED_16 Index: 22353_d = 5751_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.00	ms	3.00
Subcodes	**Lenze setting**	**Info**
C02222/1	0.00 ms	LS_WriteParamList: c72_val1
C02222/2	0.00 ms	LS_WriteParamList: c72_val2
C02222/3	0.00 ms	LS_WriteParamList: c72_val3
C02222/4	0.00 ms	LS_WriteParamList: c72_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02223

Parameter | Name: C02223 | LS_WriteParamList: Imax/M controller gain			Data type: UNSIGNED_16 Index: 22352_d = 5750_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00		100.00	
Subcodes	**Lenze setting**		**Info**
C02223/1	0.25		LS_WriteParamList: c73_s1_val1
C02223/2	0.25		LS_WriteParamList: c73_s1_val2
C02223/3	0.25		LS_WriteParamList: c73_s1_val3
C02223/4	0.25		LS_WriteParamList: c73_s1_val4
C02223/5	1.25		LS_WriteParamList: c73_s2_val1
C02223/6	1.25		LS_WriteParamList: c73_s2_val2
C02223/7	1.25		LS_WriteParamList: c73_s2_val3
C02223/8	1.25		LS_WriteParamList: c73_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02224

Parameter | Name: C02224 | LS_WriteParamList: Imax/M controller reset time			Data type: UNSIGNED_16 Index: 22351_d = $574F_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0	ms	9990	
Subcodes	**Lenze setting**		**Info**
C02224/1	65 ms		LS_WriteParamList: c74_s1_val1
C02224/2	65 ms		LS_WriteParamList: c74_s1_val2
C02224/3	65 ms		LS_WriteParamList: c74_s1_val3
C02224/4	65 ms		LS_WriteParamList: c74_s1_val4
C02224/5	30 ms		LS_WriteParamList: c74_s2_val1
C02224/6	30 ms		LS_WriteParamList: c74_s2_val2
C02224/7	30 ms		LS_WriteParamList: c74_s2_val3
C02224/8	30 ms		LS_WriteParamList: c74_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02225

Parameter | Name: C02225 | LS_WriteParamList: Vp current controller			Data type: UNSIGNED_16 Index: 22350_d = $574E_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	V/A	500.00	
Subcodes	**Lenze setting**		**Info**
C02225/1	7.00 V/A		LS_WriteParamList: c75_val1
C02225/2	7.00 V/A		LS_WriteParamList: c75_val2
C02225/3	7.00 V/A		LS_WriteParamList: c75_val3
C02225/4	7.00 V/A		LS_WriteParamList: c75_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02226

Parameter | Name: **C02226 | LS_WriteParamList: Ti current controller**			Data type: UNSIGNED_16 Index: 22349_d = $574D_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	ms	500.00	
Subcodes	**Lenze setting**		**Info**
C02226/1	10.61 ms		LS_WriteParamList: c76_val1
C02226/2	10.61 ms		LS_WriteParamList: c76_val2
C02226/3	10.61 ms		LS_WriteParamList: c76_val3
C02226/4	10.61 ms		LS_WriteParamList: c76_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02227

Parameter | Name: **C02227 | LS_WriteParamList: SC: Vp field controller**			Data type: UNSIGNED_16 Index: 22348_d = $574C_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00		500.00	
Subcodes	**Lenze setting**		**Info**
C02227/1	12.80		LS_WriteParamList: c77_val1
C02227/2	12.80		LS_WriteParamList: c77_val2
C02227/3	12.80		LS_WriteParamList: c77_val3
C02227/4	12.80		LS_WriteParamList: c77_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02228

Parameter | Name: **C02228 | LS_WriteParamList: SC: Tn field controller**			Data type: UNSIGNED_16 Index: 22347_d = $574B_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	ms	6000.0	
Subcodes	**Lenze setting**		**Info**
C02228/1	256.0 ms		LS_WriteParamList: c78_val1
C02228/2	256.0 ms		LS_WriteParamList: c78_val2
C02228/3	256.0 ms		LS_WriteParamList: c78_val3
C02228/4	256.0 ms		LS_WriteParamList: c78_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02229

Parameter | Name: **C02229 | LS_WriteParamList: SC: Settings**		Data type: UNSIGNED_8 Index: 22346_d = $574A_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C02229/1	0: Off	LS_WriteParamList: c79_s1_val1
C02229/2	0: Off	LS_WriteParamList: c79_s1_val2
C02229/3	0: Off	LS_WriteParamList: c79_s1_val3
C02229/4	0: Off	LS_WriteParamList: c79_s1_val4
C02229/5	1: On	LS_WriteParamList: c79_s2_val1
C02229/6	1: On	LS_WriteParamList: c79_s2_val2
C02229/7	1: On	LS_WriteParamList: c79_s2_val3
C02229/8	1: On	LS_WriteParamList: c79_s2_val4
C02229/9	0: Off	LS_WriteParamList: c79_s3_val1
C02229/10	0: Off	LS_WriteParamList: c79_s3_val2
C02229/11	0: Off	LS_WriteParamList: c79_s3_val3
C02229/12	0: Off	LS_WriteParamList: c79_s3_val4
C02229/13	1: On	LS_WriteParamList: c79_s4_val1
C02229/14	1: On	LS_WriteParamList: c79_s4_val2
C02229/15	1: On	LS_WriteParamList: c79_s4_val3
C02229/16	1: On	LS_WriteParamList: c79_s4_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02230

Parameter | Name: **C02230 | LS_WriteParamList: Override point of field weakening**			Data type: INTEGER_16 Index: 22345_d = 5749_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
-500	Hz	500	
Subcodes	**Lenze setting**		**Info**
C02230/1	0 Hz		LS_WriteParamList: c80_val1
C02230/2	0 Hz		LS_WriteParamList: c80_val2
C02230/3	0 Hz		LS_WriteParamList: c80_val3
C02230/4	0 Hz		LS_WriteParamList: c80_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02231

Parameter | Name: **C02231 | LS_WriteParamList: Rated motor power**		Data type: UNSIGNED_16 Index: 22344_d = 5748_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.00	kW	500.00
Subcodes	**Lenze setting**	**Info**
C02231/1	11.00 kW	LS_WriteParamList: c81_val1
C02231/2	11.00 kW	LS_WriteParamList: c81_val2
C02231/3	11.00 kW	LS_WriteParamList: c81_val3
C02231/4	11.00 kW	LS_WriteParamList: c81_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02232

Parameter | Name: **C02232 | LS_WriteParamList: Motor rotor resistance**		Data type: UNSIGNED_32 Index: 22343_d = 5747_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0	mohm	200000
Subcodes	**Lenze setting**	**Info**
C02232/1	276 mohms	LS_WriteParamList: c82_val1
C02232/2	276 mohms	LS_WriteParamList: c82_val2
C02232/3	276 mohms	LS_WriteParamList: c82_val3
C02232/4	276 mohms	LS_WriteParamList: c82_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02233

Parameter | Name: **C02233 | LS_WriteParamList: Motor stator resistance**		Data type: UNSIGNED_32 Index: 22342_d = 5746_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0	mohm	200000
Subcodes	**Lenze setting**	**Info**
C02233/1	330 mOhm	LS_WriteParamList: c84_val1
C02233/2	330 mOhm	LS_WriteParamList: c84_val2
C02233/3	330 mOhm	LS_WriteParamList: c84_val3
C02233/4	330 mOhm	LS_WriteParamList: c84_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02234

Parameter | Name: **C02234 | LS_WriteParamList: Motor stator leakage inductance**		Data type: UNSIGNED_16 Index: 22341_d = 5745_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.00 mH 650.00		
Subcodes	**Lenze setting**	**Info**
C02234/1	3.50 mH	LS_WriteParamList: c85_val1
C02234/2	3.50 mH	LS_WriteParamList: c85_val2
C02234/3	3.50 mH	LS_WriteParamList: c85_val3
C02234/4	3.50 mH	LS_WriteParamList: c85_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02236

Parameter | Name: **C02236 | LS_WriteParamList: Rated motor speed**		Data type: UNSIGNED_16 Index: 22339_d = 5743_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
50 rpm 60000		
Subcodes	**Lenze setting**	**Info**
C02236/1	1460 rpm	LS_WriteParamList: c87_val1
C02236/2	1460 rpm	LS_WriteParamList: c87_val2
C02236/3	1460 rpm	LS_WriteParamList: c87_val3
C02236/4	1460 rpm	LS_WriteParamList: c87_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02237

Parameter | Name: **C02237 | LS_WriteParamList: Rated motor current**		Data type: UNSIGNED_16 Index: 22338_d = 5742_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.20 A 320.00		
Subcodes	**Lenze setting**	**Info**
C02237/1	21.00 A	LS_WriteParamList: c88_val1
C02237/2	21.00 A	LS_WriteParamList: c88_val2
C02237/3	21.00 A	LS_WriteParamList: c88_val3
C02237/4	21.00 A	LS_WriteParamList: c88_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02238

Parameter | Name: **C02238 | LS_WriteParamList: Rated motor frequency**		Data type: UNSIGNED_16 Index: 22337_d = 5741_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
1 | Hz | 1000		
Subcodes	**Lenze setting**	**Info**
C02238/1	50 Hz	LS_WriteParamList: c89_val1
C02238/2	50 Hz	LS_WriteParamList: c89_val2
C02238/3	50 Hz	LS_WriteParamList: c89_val3
C02238/4	50 Hz	LS_WriteParamList: c89_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02239

Parameter | Name: **C02239 | LS_WriteParamList: Rated motor voltage**		Data type: UNSIGNED_16 Index: 22336_d = 5740_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0 | V | 5000		
Subcodes	**Lenze setting**	**Info**
C02239/1	400 V	LS_WriteParamList: c90_val1
C02239/2	400 V	LS_WriteParamList: c90_val2
C02239/3	400 V	LS_WriteParamList: c90_val3
C02239/4	400 V	LS_WriteParamList: c90_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02240

Parameter | Name: **C02240 | LS_WriteParamList: Motor cosine phi**		Data type: UNSIGNED_8 Index: 22335_d = $573F_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.20 | | 1.00		
Subcodes	**Lenze setting**	**Info**
C02240/1	0.85	LS_WriteParamList: c91_val1
C02240/2	0.85	LS_WriteParamList: c91_val2
C02240/3	0.85	LS_WriteParamList: c91_val3
C02240/4	0.85	LS_WriteParamList: c91_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02241

Parameter | Name: **C02241 | LS_WriteParamList: Motor magnetizing inductance**			Data type: UNSIGNED_16 Index: 22334_d = $573E_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	mH	6500.0	
Subcodes	**Lenze setting**		**Info**
C02241/1	81.0 mH		LS_WriteParamList: c92_val1
C02241/2	81.0 mH		LS_WriteParamList: c92_val2
C02241/3	81.0 mH		LS_WriteParamList: c92_val3
C02241/4	81.0 mH		LS_WriteParamList: c92_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02242

Parameter | Name: **C02242 | LS_WriteParamList: Motor magnetising current**			Data type: UNSIGNED_16 Index: 22333_d = $573D_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	A	320.00	
Subcodes	**Lenze setting**		**Info**
C02242/1	8.50 A		LS_WriteParamList: c95_val1
C02242/2	8.50 A		LS_WriteParamList: c95_val2
C02242/3	8.50 A		LS_WriteParamList: c95_val3
C02242/4	8.50 A		LS_WriteParamList: c95_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02244

Parameter | Name: **C02244 | LS_WriteParamList: Auto-DCB: Hold time**			Data type: UNSIGNED_32 Index: 22331_d = $573B_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.000	s	999.000	
Subcodes	**Lenze setting**		**Info**
C02244/1	0.500 s		LS_WriteParamList: c106_val1
C02244/2	0.500 s		LS_WriteParamList: c106_val2
C02244/3	0.500 s		LS_WriteParamList: c106_val3
C02244/4	0.500 s		LS_WriteParamList: c106_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C02245

Parameter | Name: **C02245 | LS_WriteParamList: DC braking: Hold time**			Data type: UNSIGNED_32 Index: 22330_d = $573A_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.000	s	999.000	
Subcodes	**Lenze setting**		**Info**
C02245/1	998.999 s		LS_WriteParamList: c107_val1
C02245/2	998.999 s		LS_WriteParamList: c107_val2
C02245/3	998.999 s		LS_WriteParamList: c107_val3
C02245/4	998.999 s		LS_WriteParamList: c107_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C02246

Parameter | Name: **C02246 | LS_WriteParamList: Setting of motor overload (I²xt)**			Data type: INTEGER_16 Index: 22329_d = 5739_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	250.00	
Subcodes	**Lenze setting**		**Info**
C02246/1	100.00 %		LS_WriteParamList: c120_val1
C02246/2	100.00 %		LS_WriteParamList: c120_val2
C02246/3	100.00 %		LS_WriteParamList: c120_val3
C02246/4	100.00 %		LS_WriteParamList: c120_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02249

Parameter | Name: **C02249 | LS_WriteParamList: Oscillation damping influence**			Data type: UNSIGNED_16 Index: 22326_d = 5736_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	250.00	
Subcodes	**Lenze setting**		**Info**
C02249/1	5.00 %		LS_WriteParamList: c234_val1
C02249/2	5.00 %		LS_WriteParamList: c234_val2
C02249/3	5.00 %		LS_WriteParamList: c234_val3
C02249/4	5.00 %		LS_WriteParamList: c234_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02250

Parameter | Name: **C02250 | LS_WriteParamList: Filter time - oscill. damping**			Data type: UNSIGNED_8 Index: 22325_d = 5735_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
2	ms	250	
Subcodes	**Lenze setting**		**Info**
C02250/1	32 ms		LS_WriteParamList: c235_val1
C02250/2	32 ms		LS_WriteParamList: c235_val2
C02250/3	32 ms		LS_WriteParamList: c235_val3
C02250/4	32 ms		LS_WriteParamList: c235_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02251

Parameter | Name: **C02251 | LS_WriteParamList: Oscillation damping field weakening**			Data type: UNSIGNED_8 Index: 22324_d = 5734_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0		40	
Subcodes	**Lenze setting**		**Info**
C02251/1	14		LS_WriteParamList: c236_val1
C02251/2	14		LS_WriteParamList: c236_val2
C02251/3	14		LS_WriteParamList: c236_val3
C02251/4	14		LS_WriteParamList: c236_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02252

Parameter | Name: **C02252 | LS_WriteParamList: Kp position controller**			Data type: UNSIGNED_16 Index: 22323_d = 5733_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	1/s	500.00	
Subcodes	**Lenze setting**		**Info**
C02252/1	5.00 1/s		LS_WriteParamList: c254_val1
C02252/2	5.00 1/s		LS_WriteParamList: c254_val2
C02252/3	5.00 1/s		LS_WriteParamList: c254_val3
C02252/4	5.00 1/s		LS_WriteParamList: c254_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02256

Parameter | Name: **C02256 | LS_WriteParamList: Moment of inertia**		Data type: UNSIGNED_32 Index: 22319_d = $572F_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.00	kg cm^2	6000000.00
Subcodes	**Lenze setting**	**Info**
C02256/1	0.00 kg cm^2	LS_WriteParamList: c273_val1
C02256/2	0.00 kg cm^2	LS_WriteParamList: c273_val2
C02256/3	0.00 kg cm^2	LS_WriteParamList: c273_val3
C02256/4	0.00 kg cm^2	LS_WriteParamList: c273_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100		

C02260

Parameter | Name: **C02260 | LS_WriteParamList: Speed sensor selection**		Data type: UNSIGNED_8 Index: 22315_d = $572B_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	No sensor	
1	Sensor signal FreqIn12	
2	Encoder signal FreqIn67	
3	Multi encoder	
4	Resolver	
5	Encoder signal FreqIn1267	
Subcodes	**Lenze setting**	**Info**
C02260/1	0: No sensor	LS_WriteParamList: c495_val1
C02260/2	0: No sensor	LS_WriteParamList: c495_val2
C02260/3	0: No sensor	LS_WriteParamList: c495_val3
C02260/4	0: No sensor	LS_WriteParamList: c495_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02261

Parameter | Name: **C02261 | LS_WriteParamList: SC: Field feedforward control**		Data type: UNSIGNED_16 Index: 22314_d = $572A_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0	%	600
Subcodes	**Lenze setting**	**Info**
C02261/1	200 %	LS_WriteParamList: c576_val1
C02261/2	200 %	LS_WriteParamList: c576_val2
C02261/3	200 %	LS_WriteParamList: c576_val3
C02261/4	200 %	LS_WriteParamList: c576_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02262

Parameter | Name: C02262 | LS_WriteParamList: SC: Vp field weakening controller			Data type: UNSIGNED_16 Index: 22313_d = 5729_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0000		2.0000	
Subcodes	**Lenze setting**		**Info**
C02262/1	0.0010		LS_WriteParamList: c577_val1
C02262/2	0.0010		LS_WriteParamList: c577_val2
C02262/3	0.0010		LS_WriteParamList: c577_val3
C02262/4	0.0010		LS_WriteParamList: c577_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C02263

Parameter | Name: C02263 | LS_WriteParamList: SC: Tn field weakening controller			Data type: UNSIGNED_16 Index: 22312_d = 5728_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.1	ms	6200.0	
Subcodes	**Lenze setting**		**Info**
C02263/1	20.0 ms		LS_WriteParamList: c578_val1
C02263/2	20.0 ms		LS_WriteParamList: c578_val2
C02263/3	20.0 ms		LS_WriteParamList: c578_val3
C02263/4	20.0 ms		LS_WriteParamList: c578_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C02264

Parameter | Name: C02264 | LS_WriteParamList: Empfindlichkeit Sollwertvorsteuerung		Data type: UNSIGNED_8 Index: 22311_d = 5727_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	Inactive	
1	15 bits	
2	14 Bit	
3	13 bits	
4	12 bits	
5	11 Bit	
6	10 Bit	
7	9 Bit	
Subcodes	**Lenze setting**	**Info**
C02264/1	0: Inactive	LS_WriteParamList: C653_s1_val1
C02264/2	0: Inactive	LS_WriteParamList: C653_s1_val2
C02264/3	0: Inactive	LS_WriteParamList: C653_s1_val3
C02264/4	0: Inactive	LS_WriteParamList: C653_s1_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02272

Parameter | Name: **C02272 | LS_WriteParamList: Motor phase direction of rotation**		Data type: UNSIGNED_8 Index: 22303_d = $571F_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	Not inverted	
1	Inverted	
Subcodes	**Lenze setting**	**Info**
C02272/1	0: Not inverted	LS_WriteParamList: c905_val1
C02272/2	0: Not inverted	LS_WriteParamList: c905_val2
C02272/3	0: Not inverted	LS_WriteParamList: c905_val3
C02272/4	0: Not inverted	LS_WriteParamList: c905_val4
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C02273

Parameter | Name: **C02273 | LS_WriteParamList: Speed limitation**			Data type: INTEGER_16 Index: 22302_d = $571E_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	175.00	
Subcodes	**Lenze setting**		**Info**
C02273/1	120.00 %		LS_WriteParamList: c909_s1_val1
C02273/2	120.00 %		LS_WriteParamList: c909_s1_val2
C02273/3	120.00 %		LS_WriteParamList: c909_s1_val3
C02273/4	120.00 %		LS_WriteParamList: c909_s1_val4
C02273/5	120.00 %		LS_WriteParamList: c909_s2_val1
C02273/6	120.00 %		LS_WriteParamList: c909_s2_val2
C02273/7	120.00 %		LS_WriteParamList: c909_s2_val3
C02273/8	120.00 %		LS_WriteParamList: c909_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02274

Parameter | Name: **C02274 | LS_WriteParamList: Frequency limitation**			Data type: UNSIGNED_16 Index: 22301$_d$ = 571D$_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0	Hz	1300	
Subcodes	**Lenze setting**		**Info**
C02274/1	1000 Hz		LS_WriteParamList: c910_s1_val1
C02274/2	1000 Hz		LS_WriteParamList: c910_s1_val2
C02274/3	1000 Hz		LS_WriteParamList: c910_s1_val3
C02274/4	1000 Hz		LS_WriteParamList: c910_s1_val4
C02274/5	1000 Hz		LS_WriteParamList: c910_s2_val1
C02274/6	1000 Hz		LS_WriteParamList: c910_s2_val2
C02274/7	1000 Hz		LS_WriteParamList: c910_s2_val3
C02274/8	1000 Hz		LS_WriteParamList: c910_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02275

Parameter | Name: **C02275 | LS_WriteParamList: Motor cable length**			Data type: UNSIGNED_16 Index: 22300$_d$ = 571C$_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	m	1000.0	
Subcodes	**Lenze setting**		**Info**
C02275/1	5.0 m		LS_WriteParamList: c915_val1
C02275/2	5.0 m		LS_WriteParamList: c915_val2
C02275/3	5.0 m		LS_WriteParamList: c915_val3
C02275/4	5.0 m		LS_WriteParamList: c915_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02276

Parameter | Name: **C02276 | LS_WriteParamList: Motor cable cross-section**			Data type: UNSIGNED_16 Index: 22299$_d$ = 571B$_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.50	mm^2	100.00	
Subcodes	**Lenze setting**		**Info**
C02276/1	6.00 mm^2		LS_WriteParamList: c916_val1
C02276/2	6.00 mm^2		LS_WriteParamList: c916_val2
C02276/3	6.00 mm^2		LS_WriteParamList: c916_val3
C02276/4	6.00 mm^2		LS_WriteParamList: c916_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02278

Parameter | Name: C02278 | LS_WriteParamList: PSM: Maximum motor current field weakening			Data type: UNSIGNED_16 Index: 22297_d = 5719_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	500.00	
Subcodes	**Lenze setting**		**Info**
C02278/1	30.00 %		LS_WriteParamList: c938_val1
C02278/2	30.00 %		LS_WriteParamList: c938_val2
C02278/3	30.00 %		LS_WriteParamList: c938_val3
C02278/4	30.00 %		LS_WriteParamList: c938_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02279

Parameter | Name: C02279 | LS_WriteParamList: Ultimate motor current			Data type: UNSIGNED_16 Index: 22296_d = 5718_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	A	3000.0	
Subcodes	**Lenze setting**		**Info**
C02279/1	3000.0 A		LS_WriteParamList: c939_val1
C02279/2	3000.0 A		LS_WriteParamList: c939_val2
C02279/3	3000.0 A		LS_WriteParamList: c939_val3
C02279/4	3000.0 A		LS_WriteParamList: c939_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02280

Parameter | Name: C02280 | LS_WriteParamList: Max. motor speed			Data type: UNSIGNED_16 Index: 22295_d = 5717_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
50	rpm	65000	
Subcodes	**Lenze setting**		**Info**
C02280/1	60000 rpm		LS_WriteParamList: c965_val1
C02280/2	60000 rpm		LS_WriteParamList: c965_val2
C02280/3	60000 rpm		LS_WriteParamList: c965_val3
C02280/4	60000 rpm		LS_WriteParamList: c965_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02281

Parameter | Name: **C02281 | LS_WriteParamList: VFC: Time const. slip comp.**			Data type: UNSIGNED_16 Index: 22294_d = 5716_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
1	ms	6000	
Subcodes	**Lenze setting**		**Info**
C02281/1	100 ms		LS_WriteParamList: c966_val1
C02281/2	100 ms		LS_WriteParamList: c966_val2
C02281/3	100 ms		LS_WriteParamList: c966_val3
C02281/4	100 ms		LS_WriteParamList: c966_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02284

Parameter | Name: **C02284 | LS_WriteParamList: VFC: limitation V/f +encoder**			Data type: UNSIGNED_16 Index: 22291_d = 5713_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	Hz	100.00	
Subcodes	**Lenze setting**		**Info**
C02284/1	10.00 Hz		LS_WriteParamList: c971_s1_val1
C02284/2	10.00 Hz		LS_WriteParamList: c971_s1_val2
C02284/3	10.00 Hz		LS_WriteParamList: c971_s1_val3
C02284/4	10.00 Hz		LS_WriteParamList: c971_s1_val4
C02284/5	100.00 Hz		LS_WriteParamList: c971_s2_val1
C02284/6	100.00 Hz		LS_WriteParamList: c971_s2_val2
C02284/7	100.00 Hz		LS_WriteParamList: c971_s2_val3
C02284/8	100.00 Hz		LS_WriteParamList: c971_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02285

Parameter | Name: **C02285 | LS_WriteParamList: VFC: Vp V/f +encoder**			Data type: UNSIGNED_16 Index: 22290_d = 5712_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.000	Hz/Hz	64.000	
Subcodes	**Lenze setting**		**Info**
C02285/1	0.100 Hz/Hz		LS_WriteParamList: c972_val1
C02285/2	0.100 Hz/Hz		LS_WriteParamList: c972_val2
C02285/3	0.100 Hz/Hz		LS_WriteParamList: c972_val3
C02285/4	0.100 Hz/Hz		LS_WriteParamList: c972_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1000			

C02286

Parameter | Name: C02286 | LS_WriteParamList: VFC: Ti V/f +encoder			Data type: UNSIGNED_16 Index: 22289_d = 5711_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	ms	6000.0	
Subcodes	**Lenze setting**		**Info**
C02286/1	100.0 ms		LS_WriteParamList: c973_val1
C02286/2	100.0 ms		LS_WriteParamList: c973_val2
C02286/3	100.0 ms		LS_WriteParamList: c973_val3
C02286/4	100.0 ms		LS_WriteParamList: c973_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02287

Parameter | Name: C02287 | LS_WriteParamList: VFC-ECO: Vp CosPhi controller			Data type: UNSIGNED_16 Index: 22288_d = 5710_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.000	Hz/Hz	64.000	
Subcodes	**Lenze setting**		**Info**
C02287/1	0.500 Hz/Hz		LS_WriteParamList: c975_val1
C02287/2	0.500 Hz/Hz		LS_WriteParamList: c975_val2
C02287/3	0.500 Hz/Hz		LS_WriteParamList: c975_val3
C02287/4	0.500 Hz/Hz		LS_WriteParamList: c975_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1000			

C02288

Parameter | Name: C02288 | LS_WriteParamList: VFC-ECO: Ti CosPhi controller			Data type: UNSIGNED_16 Index: 22287_d = $570F_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	ms	6000.0	
Subcodes	**Lenze setting**		**Info**
C02288/1	200.0 ms		LS_WriteParamList: c976_val1
C02288/2	200.0 ms		LS_WriteParamList: c976_val2
C02288/3	200.0 ms		LS_WriteParamList: c976_val3
C02288/4	200.0 ms		LS_WriteParamList: c976_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02289

Parameter | Name: **C02289 | LS_WriteParamList: VFC-ECO: Minimum voltage U/f**			Data type: INTEGER_16 Index: 22286_d = $570E_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
20.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C02289/1	20.00 %		LS_WriteParamList: c977_val1
C02289/2	20.00 %		LS_WriteParamList: c977_val2
C02289/3	20.00 %		LS_WriteParamList: c977_val3
C02289/4	20.00 %		LS_WriteParamList: c977_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02290

Parameter | Name: **C02290 | LS_WriteParamList: VFC-ECO: Voltage reduction ramp**			Data type: UNSIGNED_8 Index: 22285_d = $570D_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	s	5.0	
Subcodes	**Lenze setting**		**Info**
C02290/1	0.8 s		LS_WriteParamList: c982_val1
C02290/2	0.8 s		LS_WriteParamList: c982_val2
C02290/3	0.8 s		LS_WriteParamList: c982_val3
C02290/4	0.8 s		LS_WriteParamList: c982_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02291

Parameter | Name: **C02291 | LS_WriteParamList: SLVC: Field current controller gain**			Data type: INTEGER_16 Index: 22284_d = $570C_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	20.00	
Subcodes	**Lenze setting**		**Info**
C02291/1	0.50 %		LS_WriteParamList: c985_val1
C02291/2	0.50 %		LS_WriteParamList: c985_val2
C02291/3	0.50 %		LS_WriteParamList: c985_val3
C02291/4	0.50 %		LS_WriteParamList: c985_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02292

Parameter | Name: **C02292 | LS_WriteParamList: SLVC: Cross current controller gain**			Data type: INTEGER_16 Index: 22283_d = $570B_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	20.00	
Subcodes	**Lenze setting**		**Info**
C02292/1	0.00 %		LS_WriteParamList: c986_val1
C02292/2	0.00 %		LS_WriteParamList: c986_val2
C02292/3	0.00 %		LS_WriteParamList: c986_val3
C02292/4	0.00 %		LS_WriteParamList: c986_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02293

Parameter | Name: **C02293 | LS_WriteParamList: Inverter motor brake: nAdd**			Data type: INTEGER_16 Index: 22282_d = $570A_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0	rpm	1000	
Subcodes	**Lenze setting**		**Info**
C02293/1	80 rpm		LS_WriteParamList: c987_val1
C02293/2	80 rpm		LS_WriteParamList: c987_val2
C02293/3	80 rpm		LS_WriteParamList: c987_val3
C02293/4	80 rpm		LS_WriteParamList: c987_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02294

Parameter | Name: **C02294 | LS_WriteParamList: Inverter motor brake: PT1 filter time**			Data type: INTEGER_16 Index: 22281_d = 5709_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	ms	100.0	
Subcodes	**Lenze setting**		**Info**
C02294/1	0.0 ms		LS_WriteParamList: c988_val1
C02294/2	0.0 ms		LS_WriteParamList: c988_val2
C02294/3	0.0 ms		LS_WriteParamList: c988_val3
C02294/4	0.0 ms		LS_WriteParamList: c988_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10			

C02295

Parameter | Name: **C02295 | LS_WriteParamList: Flying restart fct.: Activation**		Data type: UNSIGNED_8 Index: 22280_d = 5708_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C02295/1	0: Off	LS_WriteParamList: c990_val1
C02295/2	0: Off	LS_WriteParamList: c990_val2
C02295/3	0: Off	LS_WriteParamList: c990_val3
C02295/4	0: Off	LS_WriteParamList: c990_val4
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02296

Parameter | Name: **C02296 | LS_WriteParamList: Flying restart fct.: process**		Data type: UNSIGNED_16 Index: 22279_d = 5707_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	0...+n | Start: +10 Hz	
1	-n...0 | Start: -10 Hz	
2	-n...+n | Start: +10 Hz	
3	-n...+n | Start: -10 Hz	
4	-n...+n | Start: Cx992	
Subcodes	**Lenze setting**	**Info**
C02296/1	2: -n...+n | Start: +10 Hz	LS_WriteParamList: c991_val1
C02296/2	2: -n...+n | Start: +10 Hz	LS_WriteParamList: c991_val2
C02296/3	2: -n...+n | Start: +10 Hz	LS_WriteParamList: c991_val3
C02296/4	2: -n...+n | Start: +10 Hz	LS_WriteParamList: c991_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02297

Parameter | Name: **C02297 | LS_WriteParamList: Flying restart: Start frequency**			Data type: INTEGER_16 Index: 22278_d = 5706_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
-200	Hz	200	
Subcodes	**Lenze setting**		**Info**
C02297/1	10 Hz		LS_WriteParamList: c992_val1
C02297/2	10 Hz		LS_WriteParamList: c992_val2
C02297/3	10 Hz		LS_WriteParamList: c992_val3
C02297/4	10 Hz		LS_WriteParamList: c992_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02298

Parameter | Name: C02298 | LS_WriteParamList: Flying restart fct: Int. time			Data type: UNSIGNED_16 Index: 22277$_d$ = 5705$_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.0	ms	6000.0	
Subcodes	**Lenze setting**		**Info**
C02298/1	300.0 ms		LS_WriteParamList: c993_val1
C02298/2	300.0 ms		LS_WriteParamList: c993_val2
C02298/3	300.0 ms		LS_WriteParamList: c993_val3
C02298/4	300.0 ms		LS_WriteParamList: c993_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02299

Parameter | Name: C02299 | LS_WriteParamList: Flying restart fct.: Current			Data type: INTEGER_16 Index: 22276$_d$ = 5704$_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C02299/1	25.00 %		LS_WriteParamList: c994_val1
C02299/2	25.00 %		LS_WriteParamList: c994_val2
C02299/3	25.00 %		LS_WriteParamList: c994_val3
C02299/4	25.00 %		LS_WriteParamList: c994_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02300

Parameter | Name: C02300 | LS_WriteParamList: SLPSM: Controlled current setpoint			Data type: UNSIGNED_16 Index: 22275$_d$ = 5703$_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
5.00	%	400.00	
Subcodes	**Lenze setting**		**Info**
C02300/1	100.00 %		LS_WriteParamList: c995_s1_val1
C02300/2	100.00 %		LS_WriteParamList: c995_s1_val2
C02300/3	100.00 %		LS_WriteParamList: c995_s1_val3
C02300/4	100.00 %		LS_WriteParamList: c995_s1_val4
C02300/5	20.00 %		LS_WriteParamList: c995_s2_val1
C02300/6	20.00 %		LS_WriteParamList: c995_s2_val2
C02300/7	20.00 %		LS_WriteParamList: c995_s2_val3
C02300/8	20.00 %		LS_WriteParamList: c995_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02301

Parameter | Name: **C02301 | LS_WriteParamList: SLPSM: Switching speed**			Data type: INTEGER_16 Index: 22274_d = 5702_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C02301/1	13.00 %		LS_WriteParamList: c996_s1_val1
C02301/2	13.00 %		LS_WriteParamList: c996_s1_val2
C02301/3	13.00 %		LS_WriteParamList: c996_s1_val3
C02301/4	13.00 %		LS_WriteParamList: c996_s1_val4
C02301/5	8.00 %		LS_WriteParamList: c996_s2_val1
C02301/6	8.00 %		LS_WriteParamList: c996_s2_val2
C02301/7	8.00 %		LS_WriteParamList: c996_s2_val3
C02301/8	8.00 %		LS_WriteParamList: c996_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02302

Parameter | Name: **C02302 | LS_WriteParamList: SLPSM: Filter time - rotor position**			Data type: INTEGER_16 Index: 22273_d = 5701_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C02302/1	5.00 %		LS_WriteParamList: c997_val1
C02302/2	5.00 %		LS_WriteParamList: c997_val2
C02302/3	5.00 %		LS_WriteParamList: c997_val3
C02302/4	5.00 %		LS_WriteParamList: c997_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02303

Parameter | Name: **C02303 | LS_WriteParamList: SLPSM: Filter time rotor position**			Data type: INTEGER_16 Index: 22272_d = 5700_h
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0.5	ms	20.0	
Subcodes	**Lenze setting**		**Info**
C02303/1	3.0 ms		LS_WriteParamList: c998_s1_val1
C02303/2	3.0 ms		LS_WriteParamList: c998_s1_val2
C02303/3	3.0 ms		LS_WriteParamList: c998_s1_val3
C02303/4	3.0 ms		LS_WriteParamList: c998_s1_val4
C02303/5	5.0 ms		LS_WriteParamList: c998_s2_val1
C02303/6	5.0 ms		LS_WriteParamList: c998_s2_val2
C02303/7	5.0 ms		LS_WriteParamList: c998_s2_val3
C02303/8	5.0 ms		LS_WriteParamList: c998_s2_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02304

Parameter | Name: **C02304 | LS_WriteParamList: SLPSM: PLL gain**			Data type: INTEGER_16 Index: 22271_d = $56FF_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0	%	1000	
Subcodes	**Lenze setting**		**Info**
C02304/1	400 %		LS_WriteParamList: c999_val1
C02304/2	400 %		LS_WriteParamList: c999_val2
C02304/3	400 %		LS_WriteParamList: c999_val3
C02304/4	400 %		LS_WriteParamList: c999_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02305

Parameter | Name: **C02305 | LS_WriteParamList: PSM: Ppp saturation characteristic**			Data type: UNSIGNED_8 Index: 22270_d = $56FE_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
0	%	255	
Subcodes	**Lenze setting**		**Info**
C02305/1	100 %		LS_WriteParamList: c2853_s1_val1
C02305/2	100 %		LS_WriteParamList: c2853_s1_val2
C02305/3	100 %		LS_WriteParamList: c2853_s1_val3
C02305/4	100 %		LS_WriteParamList: c2853_s1_val4
C02305/5	100 %		LS_WriteParamList: c2853_s2_val1
C02305/6	100 %		LS_WriteParamList: c2853_s2_val2
C02305/7	100 %		LS_WriteParamList: c2853_s2_val3
C02305/8	100 %		LS_WriteParamList: c2853_s2_val4
C02305/9	100 %		LS_WriteParamList: c2853_s3_val1
C02305/10	100 %		LS_WriteParamList: c2853_s3_val2
C02305/11	100 %		LS_WriteParamList: c2853_s3_val3
C02305/12	100 %		LS_WriteParamList: c2853_s3_val4
C02305/13	100 %		LS_WriteParamList: c2853_s4_val1
C02305/14	100 %		LS_WriteParamList: c2853_s4_val2
C02305/15	100 %		LS_WriteParamList: c2853_s4_val3
C02305/16	100 %		LS_WriteParamList: c2853_s4_val4
C02305/17	100 %		LS_WriteParamList: c2853_s5_val1
C02305/18	100 %		LS_WriteParamList: c2853_s5_val2
C02305/19	100 %		LS_WriteParamList: c2853_s5_val3
C02305/20	100 %		LS_WriteParamList: c2853_s5_val4
C02305/21	100 %		LS_WriteParamList: c2853_s6_val1
C02305/22	100 %		LS_WriteParamList: c2853_s6_val2
C02305/23	100 %		LS_WriteParamList: c2853_s6_val3
C02305/24	100 %		LS_WriteParamList: c2853_s6_val4
C02305/25	100 %		LS_WriteParamList: c2853_s7_val1
C02305/26	100 %		LS_WriteParamList: c2853_s7_val2

Parameter | Name: **C02305 | LS_WriteParamList: PSM: Ppp saturation characteristic**		Data type: UNSIGNED_8 Index: 22270_d = $56FE_h$
C02305/27	100 %	LS_WriteParamList: c2853_s7_val3
C02305/28	100 %	LS_WriteParamList: c2853_s7_val4
C02305/29	100 %	LS_WriteParamList: c2853_s8_val1
C02305/30	100 %	LS_WriteParamList: c2853_s8_val2
C02305/31	100 %	LS_WriteParamList: c2853_s8_val3
C02305/32	100 %	LS_WriteParamList: c2853_s8_val4
C02305/33	100 %	LS_WriteParamList: c2853_s9_val1
C02305/34	100 %	LS_WriteParamList: c2853_s9_val2
C02305/35	100 %	LS_WriteParamList: c2853_s9_val3
C02305/36	100 %	LS_WriteParamList: c2853_s9_val4
C02305/37	100 %	LS_WriteParamList: c2853_s10_val1
C02305/38	100 %	LS_WriteParamList: c2853_s10_val2
C02305/39	100 %	LS_WriteParamList: c2853_s10_val3
C02305/40	100 %	LS_WriteParamList: c2853_s10_val4
C02305/41	100 %	LS_WriteParamList: c2853_s11_val1
C02305/42	100 %	LS_WriteParamList: c2853_s11_val2
C02305/43	100 %	LS_WriteParamList: c2853_s11_val3
C02305/44	100 %	LS_WriteParamList: c2853_s11_val4
C02305/45	100 %	LS_WriteParamList: c2853_s12_val1
C02305/46	100 %	LS_WriteParamList: c2853_s12_val2
C02305/47	100 %	LS_WriteParamList: c2853_s12_val3
C02305/48	100 %	LS_WriteParamList: c2853_s12_val4
C02305/49	100 %	LS_WriteParamList: c2853_s13_val1
C02305/50	100 %	LS_WriteParamList: c2853_s13_val2
C02305/51	100 %	LS_WriteParamList: c2853_s13_val3
C02305/52	100 %	LS_WriteParamList: c2853_s13_val4
C02305/53	100 %	LS_WriteParamList: c2853_s14_val1
C02305/54	100 %	LS_WriteParamList: c2853_s14_val2
C02305/55	100 %	LS_WriteParamList: c2853_s14_val3
C02305/56	100 %	LS_WriteParamList: c2853_s14_val4
C02305/57	100 %	LS_WriteParamList: c2853_s15_val1
C02305/58	100 %	LS_WriteParamList: c2853_s15_val2
C02305/59	100 %	LS_WriteParamList: c2853_s15_val3
C02305/60	100 %	LS_WriteParamList: c2853_s15_val4
C02305/61	100 %	LS_WriteParamList: c2853_s16_val1
C02305/62	100 %	LS_WriteParamList: c2853_s16_val2
C02305/63	100 %	LS_WriteParamList: c2853_s16_val3
C02305/64	100 %	LS_WriteParamList: c2853_s16_val4
C02305/65	100 %	LS_WriteParamList: c2853_s17_val1
C02305/66	100 %	LS_WriteParamList: c2853_s17_val2
C02305/67	100 %	LS_WriteParamList: c2853_s17_val3
C02305/68	100 %	LS_WriteParamList: c2853_s17_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02306

Parameter | Name: **C02306 | LS_WriteParamList: PSM: Imax Ppp saturation characteristic**		Data type: UNSIGNED_16 Index: 22269_d = $56FD_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
0.0	A	3000.0
Subcodes	**Lenze setting**	**Info**
C02306/1	3000.0 A	LS_WriteParamList: c2855_val1
C02306/2	3000.0 A	LS_WriteParamList: c2855_val2
C02306/3	3000.0 A	LS_WriteParamList: c2855_val3
C02306/4	3000.0 A	LS_WriteParamList: c2855_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10		

C02307

Parameter | Name: **C02307 | LS_WriteParamList: PSM: Activate Ppp saturation char.**		Data type: UNSIGNED_8 Index: 22268_d = $56FC_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C02307/1	0: Off	LS_WriteParamList: c2859_val1
C02307/2	0: Off	LS_WriteParamList: c2859_val
C02307/3	0: Off	LS_WriteParamList: c2859_val
C02307/4	0: Off	LS_WriteParamList: c2859_val
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02311

Parameter | Name: **C02311 | LS_WriteParamList: PLI without motion: Adaptation of time duration**		Data type: INTEGER_8 Index: 22264_d = $56F8_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Setting range (min. value | unit | max. value)		
-10		10
Subcodes	**Lenze setting**	**Info**
C02311/1	0	LS_WriteParamList: c2872_s1_val1
C02311/2	0	LS_WriteParamList: c2872_s1_val2
C02311/3	0	LS_WriteParamList: c2872_s1_val3
C02311/4	0	LS_WriteParamList: c2872_s1_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02312

Parameter | Name: **C02312 | LS_WriteParamList: PLI without motion**			Data type: UNSIGNED_16 Index: 22263_d = $56F7_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	for SLPSM with controller enable		
Bit 1	for SC PSM with mains on		
Bit 2	for SC PSM with controller enable		
Bit 3	for SC PSM once after fault reset		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C02312/1	0x0001		LS_WriteParamList: c2874_s1_val1
C02312/2	0x0001		LS_WriteParamList: c2874_s1_val2
C02312/3	0x0001		LS_WriteParamList: c2874_s1_val3
C02312/4	0x0001		LS_WriteParamList: c2874_s1_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C02313

Parameter | Name: **C02313 | LS_WriteParamList: PLI without motion: Adaptation of ident angle**			Data type: INTEGER_8 Index: 22262_d = $56F6_h$
From version 12.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over			
Setting range (min. value | unit | max. value)			
-100	°	100	
Subcodes	**Lenze setting**		**Info**
C02313/1	0 °		LS_WriteParamList: c2875_s1_val1
C02313/2	0 °		LS_WriteParamList: c2875_s1_val2
C02313/3	0 °		LS_WriteParamList: c2875_s1_val3
C02313/4	0 °		LS_WriteParamList: c2875_s1_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02314

Parameter | Name: C02314 | Reserved	Data type: UNSIGNED_8 Index: 22261_d = $56F5_h$
This code is used device-internally and must not be written by the user side!	

C02315

Parameter | Name: **C02315 | LS_WriteParamList: Manual entry of motor type**		Data type: UNSIGNED_8 Index: 22260_d = $56F4_h$
From version 14.00.00 onwards: Parameter values (1st value ... 4th value) for optional motor data change-over		
Selection list		
0	Automatic	
1	ASM	
2	PSM	
Subcodes	**Lenze setting**	**Info**
C02315/1	0: Automatic	LS_WriteParamList: c1001_s1_val1
C02315/2	0: Automatic	LS_WriteParamList: c1001_s1_val2
C02315/3	0: Automatic	LS_WriteParamList: c1001_s1_val3
C02315/4	0: Automatic	LS_WriteParamList: c1001_s1_val4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02430

Parameter | Name: **C02430 | Axis bus address and no. of nodes**			Data type: UNSIGNED_8 Index: 22145_d = 5681_h
From version 02.00.00 ▸ Data transfer axis bus			
Setting range (min. value | unit | max. value)			
0		62	
Subcodes	**Lenze setting**		**Information**
C02430/1	0		Axis bus address • 0 ≡ Nodes disconnected from the axis bus • 1 ≡ Master • 2 ... 62 ≡ slave 1 ... slave 61
C02430/2	2		Axis bus no. of nodes • Number of slaves connected to the axis bus. • Setting only required for the master.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C02431

Parameter | Name: **C02431 | Axis bus time settings**			Data type: UNSIGNED_16 Index: $22144_d = 5680_h$
From version 02.00.00			▶ Data transfer axis bus
Setting range (min. value | unit | max. value)			
0	ms	65000	
Subcodes	**Lenze setting**		**Information**
C02431/1	3000 ms		Axis bus decel. boot-up • Delay during status change from "Boot-up" to "Operational".
C02431/2	1000 ms		Axis bus decel. BusOff recovery • Delay after which the "BusOff" status is reset automatically. ▶ Data error monitoring.
C02431/3	1000 ms		Axis bus monitoring time ▶ Data error monitoring.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C02435

Parameter | Name: **C02435 | Axis bus status**		Data type: UNSIGNED_8 Index: $22140_d = 567C_h$
From version 02.00.00		▶ Data transfer axis bus
Selection list (read only)		
0	PDO active	
1	SDO active	
2	Reserved	
3	Reserved	
4	BootUp	
5	Stopped	
6	Reserved	
7	Off	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C02436

Parameter | Name: **C02436 | Axis bus error status**		Data type: UNSIGNED_8 Index: $22139_d = 567B_h$
From version 02.00.00		▶ Data transfer axis bus
Selection list (read only)		
0	No Error	
1	Warning ErrActive	
2	Warning ErrPassive	
3	Bus off	
4	Reserved	
5	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C02437

Parameter | Name: **C02437 | Axis bus MessageError**		Data type: UNSIGNED_8 Index: 22138_d = $567A_h$
From version 02.00.00		▸ Data transfer axis bus
Selection list (read only)		
0	No Error	
1	StuffError	
2	FormError	
3	AckError	
4	Bit1Error	
5	Bit0Error	
6	CRCError	
7	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C02438

Parameter | Name: **C02438 | CAB_Tx_Rx_diagnosis**			Data type: UNSIGNED_16 Index: 22137_d = 5679_h
From version 12.00.00			▸ Data transfer axis bus
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Information**
C02438/1			Axis bus Tx_Error
C02438/2			Axis bus Rx_Error
C02438/3			Axis bus transmitting meter
C02438/4			Axis bus receiving meter
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☑ COM ☐ MOT Scaling factor: 1			

C02440

Parameter | Name: **C02440 | AxisBusIO slave/master**		Data type: UNSIGNED_8 Index: 22135_d = 5677_h
From version 02.00.00		▸ Axis bus
Selection list		
0	Off	
1	master	
2	slave	
3	IO	
Subcodes	**Lenze setting**	**Information**
C02440/1	0: Off	Axis bus IO function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☑ COM ☐ MOT Scaling factor: 1		

C02442

Parameter | Name: **C02442 | CAB decoupling inputs**			Data type: UNSIGNED_16 Index: 22133_d = 5675_h
From version 12.00.00 Bit-coded selection of the events causing the decoupling of the receive data of the data transfer axis bus. ▸ Data transfer axis bus			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	BusOff_MsgErr		
Bit 1	Warning		
Bit 2	NodeStopped		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Trouble		
Bit 15	Fault		
Subcodes	**Lenze setting**		**Information**
C02442/1	0x0000		Decoupling inputs from the axis bus
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C02443

Parameter | Name: **C02443 | CAB decoupling value**			Data type: UNSIGNED_16 Index: 22132_d = 5674_h
From version 12.00.00 Definition of the value the receive data of the data transfer axis bus are to have in the decoupled state. Data transfer axis bus			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Information**
C02443/1	0		LS_AxisBusIn: wLine1 DiscVal
C02443/2	0		LS_AxisBusIn: wLine2 DiscVal
C02443/3	0		LS_AxisBusIn: wLine3 DiscVal
C02443/4	0		LS_AxisBusIn: wCas1 DiscVal
C02443/5	0		LS_AxisBusIn: wCas2 DiscVal
C02443/6	0		LS_AxisBusIn: wCas3 DiscVal
C02443/7	0		LS_AxisBusIn: wCas4 DiscVal
C02443/8	0		LS_AxisBusAux: wAuxIn1 DiscVal
C02443/9	0		LS_AxisBusAux: wAuxIn2 DiscVal
C02443/10	0		LS_AxisBusAux: wAuxIn3 DiscVal
C02443/11	0		LS_AxisBusAux: wAuxIn4 DiscVal
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02444

Parameter | Name: **C02444 | CAB configuration**			Data type: UNSIGNED_16 Index: 22131_d = 5673_h
From version 12.00.00 Bit-coded settings ▸ Data transfer axis bus			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Edge detection SetFail		0 ≡ input *bSetFail_DigOut* is level-sensitive. 1 ≡ input *bSetFail_DigOut* is edge-sensitive.
Bit 1	Reserved		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	same CobId for all Cas		
Bit 15	500 kbps		
Subcodes	**Lenze setting**		**Information**
C02444/1	0x0000		Axis bus configuration
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C02556

Parameter | Name: **C02556 | Position controller: Limitation**			Data type: INTEGER_32 Index: 22019_d = 5603_h
Setting range (min. value | unit | max. value)			
0	Incr.	2147483647	
Subcodes	**Lenze setting**		**Info**
C02556/1	65536 incr.		Following error limit • Setting of the maximum permissible correcting variable or the maximum permissible following error for the position controller.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02580

Parameter | Name: **C02580 | Holding brake: Operating mode**			Data type: UNSIGNED_8 Index: 21995_d = $55EB_h$
Selection of the operating mode for holding brake control Holding brake control			
Selection list (Lenze setting printed in bold)			**Info**
0	**Brake control off**		No holding brake is used. Internal control is switched off.
11	Manually controlled		The holding brake is released and closed via a control bit in the MCK control word.
12	Autom. controlled		The holding brake is automatically released and closed via speed setpoint comparisons.
13	Semi-automat. controlled		The holding brake is released and closed via a control bit in the MCK control word. • In contrast to the manual operation (mode 11) • the feedforward control is active in this mode, preventing a sagging e.g. in case of a hoist. • the brake in this mode also closes when the controller is inhibited in order to prevent the axis in a hoist from falling.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02581

Parameter | Name: **C02581 | Holding brake: Speed thresholds**			Data type: INTEGER_16 Index: 21994_d = $55EA_h$
Speed setpoint threshold and hysteresis for automatic holding brake control Holding brake control			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C02581/1	5.00 %		Holding brake: Switching threshold • Switching threshold of the speed setpoint from which on the holding brake is released/applied automatically.
C02581/2	1.00 %		Holding brake: Hyst.release • Hysteresis for holding brake release. • Release threshold = switching threshold + release hysteresis
C02581/3	1.00 %		Holding brake: Hyst. close • Hysteresis for holding brake application. • Application threshold = switching threshold - application hysteresis
C02581/4	0.00 %		Holding brake: FF control starting value 1
C02581/5	0.00 %		Holding brake: FF control starting value 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02582

Parameter | Name: **C02582 | Holding brake: Setting**			Data type: UNSIGNED_8 Index: $21993_d = 55E9_h$
Activation of functional holding brake control options			▸ Holding brake control
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x00		0xFF	**0x40** (decimal: 64)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	bMBrakeReleaseOut invert.		Activation of inverted control • 1 ≡ Inverted logic of the control signal for the holding brake control switching element.
Bit 1 ☐	Horizontal brake protection		Brake response in case of pulse inhibit • 1 ≡ In the case of a pulse inhibit, the actual speed value is monitored which must reach the "Close" threshold value to cause the holding brake to be applied. **Note:** • This function is only active if bit 3 (horizontal/winding technology) is set as well. The function is used in order that, when the controller is inhibited, the holding brake of a drive with horizontal traverse path does not wear out during rotation. • With vertical motion (bit 3 = 0), this function is not active. Especially with hoists and activated pulse inhibit of the inverter, an immediate application of the brake is essential for safety-related reasons!
Bit 2 ☐	with hoist inv. feedfwd. control		Direction of feedforward control with vertical/hoist technology: • 0 ≡ Positive direction • 1 ≡ Negative direction **Note:** Reversal (Ccw) is then considered.
Bit 3 ☐	Horizontal application		Direction of movement of the axis • 0 ≡ The axis performs vertical movements. Gravitational acceleration causes movements. • 1 ≡ The direction of the axis is horizontal or rotary. The gravitational acceleration does not cause any movement.

Parameter | Name: **C02582 | Holding brake: Setting**		Data type: UNSIGNED_8 Index: 21993_d = $55E9_h$

Bit 4 ☐	Feedforward control C2581	Selection of the feedforward control value
		• 0 ≡ Automatic selection.
		  • The torque saved at the last stop is used.
		• 1 ≡ Manual selection.
		  • *bMBrakeStartValue2* = FALSE: The feedforward control value 1 set in C02581/4 is used.
		  • *bMBrakeStartValue2* = TRUE: The feedforward control value set in C02581/5 is used.

Bit 5 ☐	Feedback monitoring	Activation of status monitoring • 1 ≡ The *bMBrakeApplied* input for status detection of the brake (via a switching contact at the brake) is monitored after the waiting time set in C02589/3 has expired.
Bit 6 ☑	SyncRampe L_NSet_1	From version 02.00.00 Selection of the ramp time for the synchronisation process to setpoint speed after the brake opening time has elapsed **Revised behaviour from version 02.00.00:** • 1 ≡ The ramp time of the effective acceleration of the ramp function generator (L_NSet_1) is used (Lenze setting). • 0 ≡ As before, the ramp time set in C02610/1 is used. **Note:** The changeover can be dynamically both via the ramp parameter and via bit 6.
Bit 7 ☐	Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT		

C02589

Parameter | Name: **C02589 | Holding brake: Time system**			Data type: UNSIGNED_16 Index: 21986_d = $55E2_h$
Operating times of the holding brake • The electromechanical delay times of the holding brake are specified in the data sheets or on the holding brake nameplate. ▸ Holding brake control			
Setting range (min. value | unit | max. value)			
0	ms	60000	
Subcodes	**Lenze setting**		**Info**
C02589/1	100 ms		Holding brake: Application time • Time in which the holding brake is completely applied from the beginning of control and in which the controller is inhibited.
C02589/2	100 ms		Holding brake: Release time • Time in which the holding brake is completely released from the beginning of control.
C02589/3	100 ms		Holding brake: Waiting time status • Time after which all transient reactions are completed and the switching status of the holding brake is stable. Beginning of monitoring the feedback signal for the switching status of the holding brake.
C02589/4	0 ms		Holding brake: Ramp time FF ctrl
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02593

Parameter | Name: **C02593 | Holding brake: Activation time**		Data type: UNSIGNED_32 Index: 21982_d = $55DE_h$
Time parameter for the delay of trigger signals of the holding brake control Holding brake control		
Setting range (min. value | unit | max. value)		
0.000	s	3600.000
Subcodes	**Lenze setting**	**Info**
C02593/1	0.000 s	Holding brake: Actual value monitoring • Time in which the actual value is supposed to reach the threshold for closing the brake if the setpoint has already reached the threshold. • Time > 0 s: If the actual speed value has not reached the threshold within the time for brake application, the holding brake is applied by control. • Time = 0 s: The brake is only applied by control when the actual speed has reached the application threshold.
C02593/2	0.000 s	Holding brake: Application delay • Time by which the control process for holding brake application is delayed. The time expires when the speed setpoint has reached the switching threshold for application. • With positioning processes, a continuous application and release of the holding brake can thus be suppressed for the set time.
C02593/3	0.000 s	Holding brake: Reserved
C02593/4	0.000 s	Holding brake: Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000		

C02607

Parameter | Name: **C02607 | Holding brake: Status**			Data type: UNSIGNED_16 Index: 21968_d = $55D0_h$
Switching status of the holding brake control			▸ Holding brake control
Display area (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Brake applied		1 ≡ Holding brake is completely applied
Bit 1	Brake released		1 ≡ Holding brake is completely released
Bit 2	Feedforward control active		1 ≡ Feedforward control for holding of the load via the motor is active before the holding brake releases.
Bit 3	Closing active		1 ≡ The brake closing time (C02589/1) expires
Bit 4	Forced release active		1 ≡ In case of automatic operation of the holding brake control, the brake is directly released via the MCK input *bMBrakeRelease* = TRUE
Bit 5	Release active		1 ≡ The brake release time (C02589/2) expires
Bit 6	Setpoint synchronisation active		1 ≡ A speed setpoint at the MCK is approached along a defined ramp after brake release
Bit 7	Signalling contact error		1 ≡ The status monitoring has tripped
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT			

C02610

Parameter | Name: **C02610 | MCK: Accel./decel. times**			Data type: UNSIGNED_32 Index: 21965_d = $55CD_h$
Ramp times for speed setpoint synchronisation			
Setting range (min. value | unit | max. value)			
0.000	s	999.999	
Subcodes	**Lenze setting**		**Info**
C02610/1	2.000 s		Holding brake: ramp time synchr. • Ramp time for the synchronisation process to setpoint speed after the brake opening time has elapsed **Revised behaviour from version 02.00.00:** • The setting made here is only effective if bit 6 "SyncRampe L_NSet_1" in C02582 is set to "0". • In the Lenze setting of C02582 (Bit 6 = "1"), the ramp time of the effective acceleration of the ramp function generator (L_NSet_1) is used. ▸ Holding brake control
C02610/2	2.000 s		MCK: Ramp time synchr. setpoint • Time for synchronisation ramps between setpoint jumps occurring through the exceedance of minimum and maximum speed setpoint limit ranges.
C02610/3	2.000 s		MCK: SM stopping ramp
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1000			

C02611

Parameter | Name: **C02611 | MCK: Limitations**			Data type: INTEGER_16 Index: 21964_d = $55CC_h$
Speed setpoint limits for the determination of limited validity ranges **Note:** Traversing with setpoints through resulting blocking zones is executed with the ramp set in C02610/2. ▸ Speed Min/Max			
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Info**
C02611/1	199.99 %		MCK: Pos. max. speed • Upper limit of the speed setpoint limitation in positive direction of rotation.
C02611/2	0.00 %		MCK: Pos. min. speed • Lower limit of the speed setpoint limitation in positive direction of rotation.
C02611/3	0.00 %		MCK: Neg. min. speed • Lower limit of the speed setpoint limitation in negative direction of rotation.
C02611/4	199.99 %		MCK: Neg. max. speed • Upper limit of the speed setpoint limitation in negative direction of rotation.
C02611/5	0.50 %		MCK: Max. speed • From version 12.00.00 • Maximum value for transferring the current speed to the speed setpoint when the internal operating mode "StandBy" is exited.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02652

Parameter | Name: **C02652 | Settings of measuring system**			Data type: UNSIGNED_16 Index: 21923_d = $55A3_h$
From version 02.00.00			
Setting range (min. hex value | max. hex value)			**Lenze setting**
0x0000		0xFFFF	**0x0000** (decimal: 0)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	Actual MCTRL position received at mains switch-off		▸ Actual "MCTRL position received at mains switch-off"
Bit 1 ☐	Delete reference when machine data is changed		From version 12.00.00 ▸ Status bit "HomPosAvailable"
Bit 2 ☐	Adapt position encoder offset with homing [C1112/1]		
Bit 3 ☐	Reference reconstruction after encoder error removal		
Bit 4 ☐	Reserved		
Bit 5 ☐	Reserved		
Bit 6 ☐	Reserved		
Bit 7 ☐	Reserved		
Bit 8 ☐	Reserved		
Bit 9 ☐	Reserved		
Bit 10 ☐	Reserved		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT			

C02810

Parameter | Name: C02810 | TP: Edge selection		Data type: UNSIGNED_8 Index: $21765_d = 5505_h$
		▸ Touch-probe detection
Selection list		
0	Off	
1	Rising	
2	Falling	
3	Rising and falling	
Subcodes	**Lenze setting**	**Info**
C02810/1	0: Off	Reserved
C02810/2	0: Off	Reserved
C02810/3	0: Off	TPDigIn3: Edge selection
C02810/4	0: Off	TPDigIn4: Edge selection
C02810/5	0: Off	TPDigIn5: Edge selection
C02810/6	0: Off	TPDigIn6: Edge selection
C02810/7	0: Off	TPDigIn7: Edge selection
C02810/8	0: Off	TPEncoderZtrack: Edge selection • From version 12.00.00
C02810/9	0: Off	TPResolverZtrack: Edge selection • From version 12.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C02811

Parameter | Name: C02811 | TP: Sensor delay			Data type: UNSIGNED_16 Index: $21764_d = 5504_h$
			▸ Touch-probe detection
Setting range (min. value | unit | max. value)			
0	µs	6999	
Subcodes	**Lenze setting**		**Info**
C02811/1	0 µs		Reserved
C02811/2	0 µs		Reserved
C02811/3	0 µs		TPDigIn3: Sensor delay
C02811/4	0 µs		TPDigIn4: Sensor delay
C02811/5	0 µs		TPDigIn5: Sensor delay
C02811/6	0 µs		TPDigIn6: Sensor delay
C02811/7	0 µs		TPDigIn7: Sensor delay
C02811/8	0 µs		TPEncoderZTrack: Sensor delay • From version 12.00.00
C02811/9	0 µs		TPResolverZTrack: Sensor delay • From version 12.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02812

Parameter | Name: C02812 | TP: Position offset			Data type: INTEGER_32 Index: 21763_d = 5503_h
			▶ Touch-probe detection
Setting range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes	**Lenze setting**		**Info**
C02812/1	0.0000 units		Reserved
C02812/2	0.0000 units		Reserved
C02812/3	0.0000 units		TPDigIn3: Pos offset
C02812/4	0.0000 units		TPDigIn4: Pos offset
C02812/5	0.0000 units		TPDigIn5: Pos offset
C02812/6	0.0000 units		TPDigIn6: Pos offset
C02812/7	0.0000 units		TPDigIn7: Pos offset
C02812/8	0.0000 units		TPEncoderZTrack: Pos offset • From version 12.00.00
C02812/9	0.0000 units		TPResolverZTrack: Pos offset • From version 12.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C02813

Parameter | Name: C02813 | TP: Pos. window start			Data type: INTEGER_32 Index: 21762_d = 5502_h
			▶ Touch-probe detection
Setting range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes	**Lenze setting**		**Info**
C02813/1	-214748.3647 units		TPDigIn3: Window start
C02813/2	-214748.3647 units		TPDigIn4: Window start
C02813/3	-214748.3647 units		TPDigIn5: Window start
C02813/4	-214748.3647 units		TPEncoderZTrack: Window start • From version 12.00.00
C02813/5	-214748.3647 units		TPResolverZTrack: Window start • From version 12.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C02814

Parameter | Name: **C02814 | TP: Pos. window end**			Data type: INTEGER_32 Index: $21761_d = 5501_h$
			▶ Touch-probe detection
Setting range (min. value | unit | max. value)			
-214748.3647	units	214748.3647	
Subcodes	**Lenze setting**		**Info**
C02814/1	214748.3647 units		TPDigIn3: Window end
C02814/2	214748.3647 units		TPDigIn4: Window end
C02814/3	214748.3647 units		TPDigIn5: Window end
C02814/4	214748.3647 units		TPEncoderZTrack: Window end • From version 12.00.00
C02814/5	214748.3647 units		TPResolverZTrack: Window end • From version 12.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000			

C02815

Parameter | Name: **C02815 | TP: Position source**		Data type: UNSIGNED_8 Index: $21760_d = 5500_h$
		▶ Touch-probe detection
Selection list		
0	Position encoder actual value	
1	Pos DigIn1/2	
2	Pos resolver	
3	Pos MultiEncoder	
4	Modulo position encoder actual value	
Subcodes	**Lenze setting**	**Info**
C02815/1	0: Position encoder actual value	Reserved
C02815/2	0: Position encoder actual value	Reserved
C02815/3	0: Position encoder actual value	TPDigIn3: Position source
C02815/4	0: Position encoder actual value	TPDigIn4: Position source
C02815/5	0: Position encoder actual value	TPDigIn5: Position source
C02815/6	0: Position encoder actual value	TPDigIn6: Position source
C02815/7	0: Position encoder actual value	TPDigIn7: Position source
C02815/8	0: Position encoder actual value	TPEncoderZTrack: Position source • From version 12.00.00
C02815/9	0: Position encoder actual value	TPResolverZTrack: Position source • From version 12.00.00
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C02816

Parameter | Name: **C02816 | TP: Signal counter**	Data type: UNSIGNED_16 Index: 21759_d = $54FF_h$
	▶ Touch-probe detection
Display range (min. value | unit | max. value)	
0 | | 65535	
Subcodes	**Info**
C02816/1	Reserved
C02816/2	Reserved
C02816/3	TPDigIn3: Signal counter
C02816/4	TPDigIn4: Signal counter
C02816/5	TPDigIn5: Signal counter
C02816/6	TPDigIn6: Signal counter
C02816/7	TPDigIn7: Signal counter
C02816/8	TPEncoderZTrack: Signal counter • From version 12.00.00
C02816/9	TPResolverZTrack: Signal counter • From version 12.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1	

C02817

Parameter | Name: **C02817 | TP: TouchProbe position**	Data type: INTEGER_32 Index: 21758_d = $54FE_h$
	▶ Touch-probe detection
Display range (min. value | unit | max. value)	
-214748.3647 | units | 214748.3647	
Subcodes	**Info**
C02817/1	Reserved
C02817/2	Reserved
C02817/3	TPDigIn3: TP position
C02817/4	TPDigIn4: TP position
C02817/5	TPDigIn5: TP position
C02817/6	TPDigIn6: TP position
C02817/7	TPDigIn7: TP position
C02817/8	TPEncoderZTrack: TP position • From version 12.00.00
C02817/9	TPResolverZTrack: TP position • From version 12.00.00
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 10000	

C02830

Parameter | Name: C02830 | DIx: Debounce time		Data type: UNSIGNED_8 Index: 21745_d = $54F1_h$
Debounce times for the digital inputs ▸ Digital input terminals		
Selection list		
0	0.00 ms	
1	0.25 ms	
2	0.50 ms	
3	0.75 ms	
4	1.00 ms	
5	1.25 ms	
6	1.50 ms	
7	1.75 ms	
8	2.00 ms	
10	2.50 ms	
12	3.00 ms	
14	3.50 ms	
16	4.00 ms	
18	4.50 ms	
20	5.00 ms	
22	5.50 ms	
24	6.00 ms	
28	7.00 ms	
32	8.00 ms	
36	9.00 ms	
40	10.0 ms	
44	11.0 ms	
48	12.0 ms	
52	13.0 ms	
56	14.0 ms	
64	16.0 ms	
72	18.0 ms	
80	20.0 ms	
88	22.0 ms	
96	24.0 ms	
104	26.0 ms	
112	28.0 ms	
120	30.0 ms	
128	32.0 ms	
Subcodes	**Lenze setting**	**Info**
C02830/1	1: 0.25 ms	DI1 ... DI7: Debounce time
C02830/...		
C02830/7		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C02840

Parameter | Name: C02840 | CountInx: Parameter			Data type: UNSIGNED_32 Index: 21735_d = $54E7_h$
Starting and comparison values for digital count inputs ▸ Use DI1(6) as counting input			
Setting range (min. value | unit | max. value)			
0	Incr.	2147483647	
Subcodes	**Lenze setting**		**Info**
C02840/1	0 incr.		CountIn1: Starting value
C02840/2	65535 incr.		CountIn1: Comparison value
C02840/3	0 incr.		CountIn6: Starting value
C02840/4	65535 incr.		CountIn6: Comparison value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02841

Parameter | Name: C02841 | CountInx: Counter content			Data type: UNSIGNED_32 Index: 21734_d = $54E6_h$
Display of the current counter content of the digital count inputs ▸ Use DI1(6) as counting input			
Display range (min. value | unit | max. value)			
0	Incr.	2147483647	
Subcodes			**Info**
C02841/1			CountIn1: Counter content
C02841/2			CountIn6: Counter content
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☑ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02842

Parameter | Name: C02842 | FreqInxx: Offset			Data type: INTEGER_16 Index: 21733_d = $54E5_h$
Offset for digital frequency inputs ▸ Using DI1(6) and DI2(7) as frequency inputs			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C02842/1	0.00 %		FreqIn12: Offset
C02842/2	0.00 %		FreqIn67: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02843

Parameter | Name: C02843 | FreqInxx: Gain			Data type: INTEGER_16 Index: 21732_d = $54E4_h$
Gain for digital frequency inputs ▸ Using DI1(6) and DI2(7) as frequency inputs			
Setting range (min. value | unit | max. value)			
-199.99	%	199.99	
Subcodes	**Lenze setting**		**Info**
C02843/1	100.00 %		FreqIn12: Gain
C02843/2	100.00 %		FreqIn67: Gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02844

Parameter | Name: C02844 | FreqIn12: Function		Data type: UNSIGNED_8 Index: 21731_d = $54E3_h$
▸ Output of the encoder position of the DI1/DI2 frequency input		
Selection list		
0	Loading with level	
1	Loading with edge	
2	Loading with level + reset	
Subcodes	**Lenze setting**	**Info**
C02844/1	0: Loading with level	FreqIn12: PosIn function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1		

C02845

Parameter | Name: C02845 | FreqIn12: PosIn comparison value			Data type: INTEGER_32 Index: 21730_d = $54E2_h$
▸ Output of the encoder position of the DI1/DI2 frequency input			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	2147418112	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02853

Parameter | Name: **C02853 | PSM: Lss saturation characteristic**			Data type: UNSIGNED_8 Index: 21722_d = $54DA_h$
			▸ Current-dependent stator leakage inductance Lss(I)
Setting range (min. value | unit | max. value)			
0	%	255	
Subcodes	**Lenze setting**		**Info**
C02853/1	100 %		PSM: Lss saturation characteristic
C02853/2	100 %		PSM: Lss saturation characteristic
C02853/3	100 %		PSM: Lss saturation characteristic
C02853/4	100 %		PSM: Lss saturation characteristic
C02853/5	100 %		PSM: Lss saturation characteristic
C02853/6	100 %		PSM: Lss saturation characteristic
C02853/7	100 %		PSM: Lss saturation characteristic
C02853/8	100 %		PSM: Lss saturation characteristic
C02853/9	100 %		PSM: Lss saturation characteristic
C02853/10	100 %		PSM: Lss saturation characteristic
C02853/11	100 %		PSM: Lss saturation characteristic
C02853/12	100 %		PSM: Lss saturation characteristic
C02853/13	100 %		PSM: Lss saturation characteristic
C02853/14	100 %		PSM: Lss saturation characteristic
C02853/15	100 %		PSM: Lss saturation characteristic
C02853/16	100 %		PSM: Lss saturation characteristic
C02853/17	100 %		PSM: Lss saturation characteristic
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02855

Parameter | Name: **C02855 | PSM: Imax Lss saturation characteristic**			Data type: UNSIGNED_16 Index: 21720_d = $54D8_h$
			▸ Current-dependent stator leakage inductance Lss(I)
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	A	3000.0	**3000.0 A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 10			

C02859

Parameter | Name: **C02859 | PSM: Activate Lss saturation char.**		Data type: UNSIGNED_8 Index: 21716_d = $54D4_h$
		▸ Current-dependent stator leakage inductance Lss(I)
Selection list(Lenze setting printed in bold)		
0	**Off**	
1	On	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02862

Parameter | Name: **C02862 | Resolver gain**			Data type: UNSIGNED_16 Index: 21713_d = $54D1_h$
			▶ Encoder/feedback system: Resolver
Setting range (min. value | unit | max. value)			
0.00	%	199.99	
Subcodes	**Lenze setting**		**Information**
C02862/1	100.00 %		Resolver: sine gain
C02862/2	100.00 %		Resolver: cos gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02863

Parameter | Name: **C02863 | Resolver: Phase error**			Data type: INTEGER_16 Index: 21712_d = $54D0_h$
			▶ Encoder/feedback system: Resolver
Setting range (min. value | unit | max. value)			**Lenze setting**
-199.99	%	199.99	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02864

Parameter | Name: **C02864 | MCTRL: Optimisiations**			Data type: UNSIGNED_16 Index: 21711_d = $54CF_h$
From version 14.00.00 Activation of special functions of the internal motor control			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Motor temp. 90° for operation without KTY		1 ≡ Warm machine. In case the temperature tracking is not active, a motor temperature of 90 °C is assumed. ▶ Temperature compensation within the motor control
Bit 1	No switch-off of the brake chopper after 4s		1 ≡ Limitation of the operating time of the brake chopper (brake transistor) is deactivated. ▶ Braking operation/brake energy management
Bit 2	Brake chopper off in case of trip		0 ≡ No response of the brake chopper in the "Fault" device status. 1 ≡ In the "Fault" device status, the brake chopper is permanently switched off without any delay. ▶ Braking operation/brake energy management
Bit 3	Reserved (do not change!)		
Bit 4	Motor ident: Switching frequency 8kHz		Switching frequency for motor parameter identification: 0 ≡ 4 kHz 1 ≡ 8 kHz ▶ Preventing a decrease of the switching frequency
Bit 5	Field/field weakening control in 500us		Cycle time for field weakening control: 0 ≡ 1 ms 1 ≡ 500 µs ▶ Optimising the behaviour of the asynchronous motor in the field weakening range

Parameter | Name: **C02864 | MCTRL: Optimisations**		Data type: UNSIGNED_16 Index: 21711_d = $54CF_h$
Bit 6	Motor ident.: No calculation C577 C578 C2864	From version 15.00.00: Following successful motor parameter identification, the field weakening controller parameters for ASM servo control are calculated as well. If these parameters are not to be calculated, bit 6 must be set to "1". 0 ≡ Calculate parameter 1 ≡ Do not calculate parameter ▶ Automatic motor data identification
Bit 7	If C2879 Bit1=0: C087 is warm	From version 15.00.00, the too high slip with the activated option "Slip calculation from motor nameplate data" can be corrected for the ASM servo control by setting bit 7 to "1". 1 ≡ Rated motor speed (C00087) is assumed for a warm machine. ▶ Slip calculation for SC
Bit 8	Suppress V/f emergency operation	From version 15.00.00, it is internally switched to the encoderless V/f characteristic control in case of a motor control type with feedback and an encoder open circuit in order to avoid impermissible motor movements. • **Exception:** There will be no change-over in case of servo control for synchronous motors (PSM). • In order that this "V/f emergency operation" works properly, the parameters relevant for the V/f characteristic control (base frequency, Vmin boost, slip compensation, etc.) have to be set correctly. As an alternative, a motor parameter identification can be executed as well. • The change-over to "V/f emergency operation" is reported via bit 4 in C01000 and via the *bWirebreakUfLinearActive* status signal at the SB LS_DeviceMonitor. • The setting in C00006 is not affected by the change-over. The "V/f emergency operation" can be suppressed by setting bit 8 to "1". 0 ≡ V/f emergency operation in case of encoder open circuit 1 ≡ No V/f emergency operation in case of encoder open circuit
Bit 9	Optimisation of field weakening control	
Bit 10	V/f+encoder without adaptation of voltage	
Bit 11	Latching of open circuit FreqIn12 or 67	
Bit 12	SLVC: No smoothing of nMotorTorqueAct_a	
Bit 13	SLVC: Setpoint feedforward control is active	
Bit 14	Wenn C173=3: OUon 810V OUoff=800V	
Bit 15	Activate nTorqueHigh and nTorqueLowLimit_a	
Subcodes	**Lenze setting**	**Info**
C02864/1	0x0000	MCTRL: Optimisations
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT		

C02865

Parameter | Name: **C02865 | MCTRL: Special settings**			Data type: UNSIGNED_16 Index: 21710_d = $54CE_h$
From version 12.00.00 Activation of special functions of the internal motor control			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	No TorqueLimit if TorquemodeOn		From version 12.00.00 onwards: If sensorless vector control (SLVC) or servo control (SC) are used, in the "torque control with speed limitation" mode the torque limitation via *nTorqueMotLimit_a* and *nTorqueGenLimit_a* is also active to limit the torque setpoint. If you want to deactivate the torque limitation to retain the former function, set bit 0 to "1". 0 ≡ Torque limitation is active 1 ≡ Torque limitation is not active
Bit 1	250ms IMP before DCB		
Bit 2	1000ms IMP before DCB		
Bit 3	DCB: Current controller and C036 app. to C088		
Bit 4	Motorident.: No calculation C73 C75 C76 C77 C78		From version 12.00.00 onwards: Following successful motor parameter identification, the current controller and field controller parameters are calculated as well. If these parameters are not to be calculated, bit 4 must be set to "1". 0 ≡ Calculate parameter 1 ≡ Do not calculate parameter ▶ Automatic motor data identification
Bit 5	Motorident.: Calculation C70 C71 C72		From version 12.00.00 onwards: Following successful motor parameter identification, the speed controller parameters can automatically be calculated as well. If these parameters are to be calculated, bit 5 must be set to "1". 0 ≡ Do not calculate parameter 1 ≡ Calculate parameter ▶ Automatic motor data identification
Bit 6	Motorident.: Calculation C11 C22 C497 C966 C982		From version 12.00.00 onwards: Following successful motor parameter identification, further controller parameters can be calculated automatically. If these parameters are to be calculated, bit 6 must be set to "1". 0 ≡ Do not calculate parameter 1 ≡ Calculate parameter ▶ Automatic motor data identification
Bit 7	Brake chopper off in case of pulse inhibit		From version 14.00.00 0 ≡ No response of the brake chopper in case of pulse inhibit. 1 ≡ In case of pulse inhibit, the brake chopper is permanently switched off without any delay. ▶ Braking operation/brake energy management
Bit 8	DC-bus supply with DC charging connection		

Parameter | Name: C02865 | MCTRL: Special settings		Data type: UNSIGNED_16 Index: 21710_d = $54CE_h$
Bit 9	Inverter characteristics inactive	From version 12.00.00 In case of very low-inductance motors (e.g. synchronous motor with a rated frequency of 1000 Hz) the inverter error characteristic needs to be switched off as otherwise the current of the device might be switched off. 0 ≡ Inverter error characteristic is active 1 ≡ Inverter error characteristic is not active
Bit 10	SCPSM: Extrapolation - actual speed value inactive	
Bit 11	TopLine: Speed encoder signal cycle 500us	
Bit 12	In case of QSP - nTorqueSetValue_a active	From version 13.00.00
Bit 13	In case of QSP - nTorquexxxLimit_a active	From version 13.00.00
Bit 14	Sensitivity of setpoint feedf. ctrl. 16 bits	From version 14.00.00
Bit 15	SC: Dead time of speed setpoint 500us	From version 14.00.00
Subcodes	**Lenze setting**	**Info**
C02865/1	0x0000	MCTRL: Special settings
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT		

C02866

Parameter | Name: C02866 | MCTRL: Special settings		Data type: UNSIGNED_8 Index: 21709_d = $54CD_h$
Activation of special functions of the internal motor control		
Selection list		
0	No	
1	Yes	
Subcodes	**Lenze setting**	**Info**
C02866/1	1: Yes	Motor ident.: Current controller par. from C075 C076 ▶ Automatic motor data identification
C02866/2	0: No	Motor phase error monitoring before operation • From version 02.00.00 ▶ Motor phase error monitoring before operation
C02866/3	0: No	Brake chopper DC-bus slave • From version 12.00.00 ▶ Control of multiple internal brake choppers in the DC-bus system
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02867

Parameter | Name: C02867 - Identification procedure		Data type: UNSIGNED_8 Index: 21708_d = $54CC_h$
Selection of the identification procedure for motor parameter identification ▸ Automatic motor parameter identification		
Selection list		**Info**
0	automatic	Automatic selection of the optimum identification procedure: • For synchronous motors, the extended identification procedure is always used. • For asynchronous motors with a rated motor power of up to 11 kW, the basic identification procedure is used. • For asynchronous motors with a rated motor power of more than 11 kW, the extended identification procedure is used.
1	standard identification	• Only for asynchronous motors. • Duration approx. 30 s
2	extended identification	• Stands out due to increased accuracy of the determined motor parameters. • Also supports synchronous motors and asynchronous motors with a power of more than 11 kW. • Duration approx. 80 s
Subcodes	**Lenze setting**	**Info**
C02867/1	0: automatic	Motor parameter identification: Process
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1		

C02868

Parameter | Name: C02868 | Setting of compatibility			Data type: UNSIGNED_32 Index: 21707_d = $54CB_h$
From version 16.00.00 This parameter serves to activate corrections and optimisations in various functions of the inverter. The activation causes a deviation of the respective functionality from the previous firmware versions! In each case, the changed functionality has to be checked after the activation took place. For a detailed explanation, the version information in the AKB has to be used.			
Setting range (min. hex value | max. hex value)			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	xx.01: MCK stop ramp opt.		0 ≡ (Maloperation): Profile stop activation with corresponding machine and profile data causes a rotation in the opposite direction and setpoint oscillations around zero speed. 1 ≡ (Correction): Corrected traversing of stop ramp may reduce the stopping times.
Bit 1	xx.02: MCK deceleration ramp opt.		0 ≡ (Maloperation): When corresponding profile parameters are entered, an impermissible jerk is caused during the braking phase when entering into the target position within a traversing profile. 1 ≡ (Correction): Depending on the profile data constellation, corrected traversing extends the respective traversing process towards the target position.

Parameter | Name: **C02868 | Setting of compatibility**		Data type: UNSIGNED_32 Index: 21707_d = $54CB_h$
Bit 2	xx.03: MCK Overchange opt.	0 ≡ (error CK10 occurs): • Traversing profile with final speed ≠ 0 needs a reversing process after reaching the target position. • Parameterised final speed in profile parameters cannot be reached at the target approach (final speed too high). • Position target cannot be approached if • the initial speed of the traversing profile ± final speed 1 ≡ (Correction): Ck10 does not occur anymore in this setting. • Travelling profiles with final speed is always executed in such a way that the target position will be reached if the parameterised final speed is neglected. • Expected Ck10 errors do not occur anymore, traversing blocks are executed.
Bit 3	xx.04: MCK PosFollower TargetPos	0 ≡ (Previous procedure): • No information on target position *LS_MotionControlKernel.dnPosTarget_p* when a synchronisation process is started in the position follower mode. Previous display of the target position remains unchanged. 1 ≡ Display of the target position and thus the rejection of the previously saved values in *LS_MotionControlKernel.dnPosTarget_p*.
Bit 4	xx.05: MCK speed, change in op. mode, opt.	From version 17.00.00 onwards 0 ≡ (Previous procedure): • The change from "StandBy" mode to a different operating mode (except for "SpeedFollower") causes the speed to jump to 0 if C02611/51 *LS_MotorInterface.nHlgSetValue_a* is set. 1 ≡ Instead of a speed jump, the "Stop" ramp is used to decelerate the motor to standstill. This applies to the state transition from "StandBy" to "Homing", "ManualJog" or "Stop" if no further actions take place in the new operating mode.
Bit 5	xx.06: MCK modulo multi, opt.	From version 17.00.00 onwards 0 ≡ (Previous procedure): • Modulo multiple clocking with one clock cycle too many. 1 ≡ Involvement of the blocking zone for a defined multiple clocking in case of positioning AbsCw and AbsCcw. ▸ Activation of the Modulo measuring system
Bit 6	xx.07: MCK suppression Ck10	From version 21.00.00 0 ≡ (Previous procedure): Feedforward control is always provided for the holding brake, even if the configuration option "horizontal motion control" is activated. • Tripping of a Ck10 error (distance calculation error). 1 ≡ Feedforward control for holding brake operation and the option "horizontal motion control" are deactivated.
Bit 7	xx.08: MCK h brake feedf. ctrl off	From version 21.00.00 0 ≡ (Previous procedure): Feedforward control is always provided for the holding brake, even if the configuration option "horizontal motion control" is activated. 1 ≡ Feedforward control for holding brake operation and the option "horizontal motion control" are deactivated.

Parameter | Name: **C02868 | Setting of compatibility**		Data type: UNSIGNED_32 Index: 21707_d = $54CB_h$
Bit 8	xx.09: Deactivate acceleration dynamics	From version 21.00.00 0 ≡ (Previous procedure): Adjusting/increasing the acceleration reduces reversal procedures for restarts during ongoing positioning procedures. 1 ≡ Reversal procedures are no longer reduced as the max. acceleration is not subject to any increase.
Bit 9	xx.10: Reserved	
Bit 10	xx.11: Reserved	
Bit 11	xx.12: Reserved	
Bit 12	xx.13: Reserved	
Bit 13	xx.14: Reserved	
Bit 14	xx.15: Reserved	
Bit 15	xx.16: Reserved	
Bit 16	xx.17: Reserved	
Bit 17	xx.18: Reserved	
Bit 18	xx.19: Reserved	
Bit 19	xx.20: Reserved	
Bit 20	xx.21: Reserved	
Bit 21	xx.22: Reserved	
Bit 22	xx.23: Reserved	
Bit 23	xx.24: Reserved	
Bit 24	xx.25: Reserved	
Bit 25	xx.26: Reserved	
Bit 26	xx.27: Reserved	
Bit 27	xx.28: Reserved	
Bit 28	xx.29: Reserved	
Bit 29	xx.30: Reserved	
Bit 30	xx.31: Reserved	
Bit 31	xx.32: Reserved	
Subcodes	**Lenze setting**	**Info**
C02868/1	0x00000000	List of modifications 01
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT		

C02869

Parameter | Name: C02869 | MCTRL: Special settings 2			Data type: UNSIGNED_16 Index: 21706_d = $54CA_h$
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Reserved		
Bit 1	Reserved		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C02869/1	0x0000		MCTRL: Special settings 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C02870

Parameter | Name: C02870 | PLI without motion: Optimisation factor			Data type: INTEGER_16 Index: 21705_d = $54C9_h$
From version 02.00.00			▸ Pole position identification without movement
Display range (min. value | unit | max. value)			
0.00	%	300.00	
Subcodes			**Info**
C02870/1			PLI without movement: degree of optimisation
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02871

Parameter | Name: **C02871 | PLI without motion: Running time**			Data type: INTEGER_16 Index: 21704_d = $54C8_h$
From version 02.00.00			Pole position identification without movement
Display range (min. value | unit | max. value)			
0.00	ms	300.00	
Subcodes			**Info**
C02871/1			PLI without movement: runtime
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 100			

C02872

Parameter | Name: **C02872 | PLI without motion: Adaptation of time duration**			Data type: INTEGER_8 Index: 21703_d = $54C7_h$
From version 02.00.00			Pole position identification without movement
Setting range (min. value | unit | max. value)			
-10		10	
Subcodes	**Lenze setting**		**Info**
C02872/1	0		PLI without movement: adaptation of time duration
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02873

Parameter | Name: **C02873 | PLI without motion: Ident. el. rotor displ. angle**			Data type: INTEGER_16 Index: 21702_d = $54C6_h$
From version 02.00.00			Pole position identification without movement
Display range (min. value | unit | max. value)			
0	°	360	
Subcodes			**Info**
C02873/1			PLI without movement: Ident. el. rotor displ. angle
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C02874

Parameter | Name: **C02874 | PLI without motion**			Data type: UNSIGNED_16 Index: 21701_d = $54C5_h$
From version 02.00.00			▸ Pole position identification without movement
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	for SLPSM with controller enable		
Bit 1	for SC PSM with mains on		
Bit 2	for SC PSM with controller enable		
Bit 3	for SC PSM once after fault reset		Ab Version 14.00.00
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C02874/1	0x0001		PLI without movement
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C02875

Parameter | Name: **C02875 | PLI without motion: Adaptation of ident angle**			Data type: INTEGER_8 Index: 21700_d = $54C4_h$
From version 02.00.00			▸ Pole position identification without movement
Setting range (min. value | unit | max. value)			
-100	°	100	
Subcodes	**Lenze setting**		**Info**
C02875/1	0 °		PLI without movement: adaptation of ident angle
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02876

Parameter | Name: **C02876 | PSM: Max. motor temperature**			Data type: UNSIGNED_8 Index: $21699_d = 54C3_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
90	°C	200	
Subcodes	**Lenze setting**		**Information**
C02876/1	150 °C		PSM: Max. motor temperature
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 1			

C02877

Parameter | Name: **C02877 | PSM temperature coefficient**			Data type: INTEGER_16 Index: $21698_d = 54C2_h$
From version 02.00.00			
Setting range (min. value | unit | max. value)			
-0.30	%/°C	0.00	
Subcodes	**Lenze setting**		**Information**
C02877/1	-0.11 %/°C		PSM temperature coefficient
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT Scaling factor: 100			

C02878

Parameter | Name: **C02878 | KTY motor temperature compensation**			Data type: UNSIGNED_16 Index: $21697_d = 54C1_h$
From version 02.00.00 A temperature compensation over the detected motor temperature (display in C00063/1) serves to compensate inaccuracies in the output torque within the motor control in case of temperature changes of the asynchronous or synchronous motor. **Note!** In the Lenze setting, the temperature compensation within the motor control is activated. The temperature compensation, however, is only active with speed encoder selection "3: Multi encoder" or "4: Resolver" in C00495 as well as error-free KTY temperature detection (display in C00063/1 ≠ 255 °C). ▶ Temperature compensation within the motor control			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	

Parameter | Name: **C02878 | KTY motor temperature compensation**		Data type: UNSIGNED_16 Index: 21697_d = $54C1_h$
Value is bit-coded:		
Bit 0	for SC PSM	
Bit 1	for SC ASM	
Bit 2	Reserved	
Bit 3	Reserved	
Bit 4	Reserved	
Bit 5	Reserved	
Bit 6	Reserved	
Bit 7	Reserved	
Bit 8	Reserved	
Bit 9	Reserved	
Bit 10	Reserved	
Bit 11	Reserved	
Bit 12	Reserved	
Bit 13	Reserved	
Bit 14	Reserved	
Bit 15	Reserved	
Subcodes	**Lenze setting**	**Information**
C02878/1	0x0003	KTY motor temperature compensation
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT		

C02879

Parameter | Name: C02879 | Slip calculation from equivalent circuit diagram			Data type: UNSIGNED_16 Index: 21696_d = $54C0_h$
From version 02.00.00 In order to achieve a better speed stability and torque accuracy, the slip calculation can be either derived from the motor nameplate data (e.g. rated motor speed) or the motor equivalent circuit diagram data (stator resistance, rotor resistance etc.).			
Setting range (min. hex value | max. hex value)			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	SLVC		Slip calculation for SLVC
Bit 1	SC ASM		Slip calculation for SC
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Subcodes	**Lenze setting**		**Info**
C02879/1	0x0002		Slip calculation from equivalent circuit diagram
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☑ MOT			

C02993

Parameter | Name: C02993 | FB xy position	Data type: UNSIGNED_32 Index: 21582_d = $544E_h$
This code is used device-internally and must not be written by the user side!	

C02994

Parameter | Name: C02994 | FB xy position	Data type: UNSIGNED_32 Index: 21581_d = $544D_h$
This code is used device-internally and must not be written by the user side!	

C02995

Parameter | Name: C02995 | FB display InputOutput	Data type: UNSIGNED_32 Index: 21580_d = $544C_h$
This code is used device-internally and must not be written by the user side!	

C02996

Parameter | Name: **C02996 | FB display InputOutput2**	Data type: UNSIGNED_32 Index: 21579_d = $544B_h$
This code is used device-internally and must not be written by the user side!	

C02998

Parameter | Name: **C02998 | FB displayinputoutput3**	Data type: UNSIGNED_32 Index: 21577_d = 5449_h
This code is used device-internally and must not be written by the user side!	

17.3 Overview of all parameters with power-related Lenze setting

			Mains voltage / device power											
			1 * 230 V							3 * 400 V				
Parameters	Name	Unit	0.25 kW	0.37 kW	0.55 kW	0.75 kW	1.1 kW	1.5 kW	2.2 kW	0.37 kW	0.55 kW	0.75 kW	1.1 kW	1.5 kW
C00016	VFC: Vmin boost	%	5.51	4.32	4.02	3.79	4.32	2.93	2.25	4.32	4.02	3.79	4.32	2.93
C00021	Slip comp.	%	8.67	6	6.33	6	7.33	7	4	6	6.33	6	7.33	7
C00022	Imax in motor mode	A	2.97	4.2	5.25	7	9.62	12.25	16.62	2.27	3.15	4.2	5.6	6.82
C00070/1	SLVC: Vp speed controller		7.62	15.76	12.59	18.9	12.52	12.45	15.76	15.76	12.59	18.9	12.52	12.45
C00070/2	SC: Vp speed controller		4.35	9.01	7.19	10.8	7.15	7.11	9.01	9.01	7.19	10.8	7.15	7.11
C00070/3	SLPSM: Vp speed controller		1.09	2.25	1.79	2.7	1.79	1.78	2.25	2.25	1.8	2.7	1.79	1.78
C00073/1	VFC: Vp Imax controller		0.46	0.32	0.33	0.27	0.33	0.32	0.2	0.28	0.3	0.27	0.33	0.32
C00074/1	VFC: Ti Imax controller	ms	65	65	65	65	65	65	65	65	65	65	65	65
C00075	Vp current controller	V/A	41.88	34.59	23.43	14.97	9.53	7.23	4.73	103.76	70.29	44.9	28.54	21.7
C00076	Ti current controller	ms	1.99	2.42	2.6	2.39	1.97	2.83	3.55	2.42	2.6	2.39	1.97	2.83
C00077	SC: Vp field controller		1.68	2.83	3.32	4.92	3.96	4.71	5.43	2.83	3.32	4.92	3.96	4.71
C00078	SC: Tn field controller	ms	33.6	56.6	66.3	98.3	79.1	94.2	108.6	56.6	66.3	98.3	79.1	94.2
C00081	Rated motor power	kW	0.25	0.37	0.55	0.75	1.1	1.5	2.2	0.37	0.55	0.75	1.1	1.5
C00082	Motor rotor resistance	mohm	11460	5480	3860	2820	2060	1438	740	16442	11580	8454	6150	4320
C00084	Motor stator resistance	mohm	10533	7133	4500	3127	2420	1277	667	21400	13500	9380	7260	3830
C00085	Motor stator leakage inductance	mH	20.94	17.29	11.72	7.48	4.77	3.62	2.37	51.88	35.15	22.45	14.27	10.85
C00087	Rated motor speed	rpm	1370	1410	1405	1410	1390	1395	1440	1410	1405	1410	1390	1395
C00088	Rated motor current	A	1.4	1.6	2.4	3.3	4.8	6.3	9.2	0.95	1.4	1.9	2.8	3.6
C00090	Rated motor voltage	V	230	230	230	230	230	230	230	400	400	400	400	400
C00091	Motor cosine phi		0.63	0.72	0.72	0.8	0.8	0.79	0.71	0.77	0.77	0.8	0.8	0.79
C00092	Motor magnetising inductance	mH	333.3	386.7	286.7	300	186.7	116.7	70	1160	860	900	560	350
C00093	Power section ID		2512	3712	5512	7512	1122	1522	2222	3714	5514	7514	1124	1524
C00095	Motor magnetising current	A	0.94	0.88	1.35	1.28	2.25	3.46	5.54	0.51	0.78	0.74	1.3	2
C00098	Rated device current	A	1.7	2.4	3	4	5.5	7	9.5	1.3	1.8	2.4	3.2	3.9
C00129	Brake resistance value	Ohm	180	180	100	100	33	33	33	390	390	390	180	180
C00130	Rated brake resistor power	W	50	50	100	100	200	200	300	100	100	100	200	200
C00131	Rated heat amount of brake resist.	kWs	7.5	7.5	15	15	30	30	45	15	15	15	30	30
C00497	Filter time constant	ms	1	1	1	1	1	1	1	1	1	1	1	1
C00916	Motor cable cross-section	mm^2	1	1	1	1	1.5	1.5	1.5	1	1	1	1.5	1.5
C00966	VFC: Time const. slip comp.	ms	100	100	100	100	100	100	100	100	100	109	100	100
C00982	VFC-ECO: Voltage reduction ramp	s	0.1	0.2	0.2	0.3	0.3	0.3	0.3	0.2	0.2	0.3	0.3	0.3
C00987	Inverter motor brake: nAdd	rpm	260	180	190	180	220	210	120	180	190	180	220	210
C00993	Flying restart: Integration time	ms	300	300	300	300	300	300	300	300	300	300	300	300

Parameters	Name	Unit	Mains voltage / device power											
			3 * 400 V											
			2.2 kW	3.0 kW	4.0 kW	5.5 kW	7.5 kW	11 kW	15 kW	18 kW	22 kW	30 kW	37 kW	45 kW
C00016	VFC: Vmin boost	%	2.25	2.48	2.06	1.94	1.7	1.47	1.36	1.32	1.24	0.97	0.7	0.59
C00021	Slip comp.	%	4	4.67	3.33	3.67	3	2.67	2.67	2	2.93	2.33	1.13	1.33
C00022	Imax in motor mode	A	9.8	12.77	16.62	22.75	28.87	41.12	56	68.25	82.25	103.25	126	155.75
C00070/1	SLVC: Vp speed controller		15.76	11.4	15.42	14.05	26	32.41	29.22	43.23	40.95	36.28	75.12	72.06
C00070/2	SC: Vp speed controller		9.01	6.51	8.81	8.03	14.86	18.52	16.7	24.7	23.4	20.73	42.93	41.18
C00070/3	SLPSM: Vp speed controller		2.25	1.63	2.2	2.01	3.72	4.63	4.18	6.18	5.85	5.18	10.73	10.3
C00073/1	VFC: Vp Imax controller		0.2	0.23	0.17	0.17	0.14	0.11	0.11	0.08	0.12	0.1	0.09	0.09
C00074/1	VFC: Ti Imax controller	ms	65	65	65	65	65	750	750	750	750	750	750	750
C00075	Vp current controller	V/A	17	14.2	11.1	6.97	8.1	7	5.5	4	3.8	1.8	1.49	1.06
C00076	Ti current controller	ms	4.25	4.38	5.34	4.77	8.62	10.61	11.96	10.53	12.67	10.59	14.57	15.17
C00077	SC: Vp field controller		5.46	5.4	8.44	7.29	7.67	12.8	14.84	19.74	21.47	31.45	35.17	35.59
C00078	SC: Tn field controller	ms	109.2	108.1	168.8	145.9	153.3	256	296.7	394.7	429.3	500	997	859
C00081	Rated motor power	kW	2.2	3	4	5.5	7.5	11	15	18.5	22	30	37	45
C00082	Motor rotor resistance	mohm	2220	1938	1262	842	642	276	186	168	130	111	36	29
C00084	Motor stator resistance	mohm	2000	1620	1040	730	470	330	230	190	150	85	51	35
C00085	Motor stator leakage inductance	mH	8.5	7.1	5.55	3.49	4.05	3.5	2.75	2	1.9	0.9	0.74	0.53
C00087	Rated motor speed	rpm	1440	1430	1450	1445	1455	1460	1460	1470	1456	1465	1483	1480
C00088	Rated motor current	A	5.3	7.2	9.3	12.5	17	21	27.8	32.8	38.8	53.9	65	79
C00090	Rated motor voltage	V	400	400	400	400	400	400	400	400	400	400	400	400
C00091	Motor cosine phi		0.73	0.75	0.73	0.77	0.76	0.85	0.87	0.9	0.9	0.87	0.87	0.88
C00092	Motor magnetising inductance	mH	210	168	170	103	68	81	65.5	73	62.5	52.6	35.1	24.4
C00093	Power section ID		2224	3024	4024	5524	7524	1134	1534	1834	2234	3034	3734	4534
C00095	Motor magnetising current	A	3.2	4	4.1	6.7	10.2	8.5	10.5	9.5	11.1	13.2	20.5	29.5
C00098	Rated device current	A	5.6	7.3	9.5	13	16.5	23.5	32	39	47	59	72	89
C00129	Brake resistance value	Ohm	180	82	47	47	27	27	18	15	15	7.5	7.5	7.5
C00130	Rated brake resistor power	W	300	200	400	800	600	1200	1400	1200	1200	1900	1900	1900
C00131	Rated heat amount of brake resist.	kWs	45	30	60	120	120	90	210	180	180	285	285	285
C00497	Filter time constant	ms	1	1	1	1	2	2	2	2	2	2	2	2
C00916	Motor cable cross-section	mm^2	1.5	2.5	2.5	2.5	4	6	6	6	6	10	10	10
C00966	VFC: Time const. slip comp.	ms	100	100	139	126	112	306	367	446	495	489	1016	877
C00982	VFC-ECO: Voltage reduction ramp	s	0.3	0.3	0.4	0.8	0.8	0.8	1.1	1.3	1.5	1.5	1.5	1.5
C00987	Inverter motor brake: nAdd	rpm	120	140	100	110	90	80	80	60	88	70	60	60
C00993	Flying restart: Integration time	ms	300	300	417.3	379.4	336.7	1159.4	1237.3	1387.3	1265.1	1445	2989.5	2577.3

17.4 Selection list - analog signals

This selection list is relevant for the following configuration parameters:

Parameters	
C00410	L_SignalMonitor_a: Signal sources
C00620	System connection list: 16-bit
C00700	LA_NCtrl: Analog connection list
C00710	LA_TabPos: Analog connection list
C00760	LA_SwitchPos: Analog connection list

Selection list - analog signals	
0	Not connected
1000	LA_NCtrl: wDriveControlStatus
1001	LA_NCtrl: wStateDetermFailNoLow
1002	LA_NCtrl: nMotorCurrent_a
1003	LA_NCtrl: nMotorSpeedAct_a
1006	LA_NCtrl: nGPAnalogSwitchOut_a
1007	LA_NCtrl: nGPArithmetikOut_a
1008	LA_NCtrl: nGPMulDivOut_a
1009	LA_NCtrl: nGPSignalOut1_a
1010	LA_NCtrl: nGPSignalOut2_a
1011	LA_NCtrl: nGPSignalOut3_a
1012	LA_NCtrl: nGPSignalOut4_a
1013	LA_NCtrl: nMotorTorqueAct_a
1014	LA_NCtrl: nDCVoltage_a
1015	LA_NCtrl: nMotorVoltage_a
1016	LA_NCtrl: nMotorSpeedSet_a
1017	LA_NCtrl: wStateDetermFailNoHigh
1023	LA_NCtrl: wFreeOut1
1024	LA_NCtrl: wFreeOut2
1025	LA_NCtrl: wFreeOut3
1026	LA_NCtrl: wFreeOut4
1100	LA_TabPos: wDriveControlStatus
1101	LA_TabPos: wStateDetermFailNoLow
1102	LA_TabPos: wStateDetermFailNoHigh
1103	LA_TabPos: nMotorCurrent_a
1104	LA_TabPos: nMotorSpeedSet_a
1105	LA_TabPos: nMotorSpeedAct_a
1106	LA_TabPos: nMotorTorqueAct_a
1107	LA_TabPos: nDCVoltage_a
1108	LA_TabPos: nMotorVoltage_a
1109	LA_TabPos: wMckState1
1110	LA_TabPos: wMckState2
1111	LA_TabPos: wMckActOperationMode
1112	LA_TabPos: wActProfileNo
1113	LA_TabPos: wActPosMode
1114	LA_TabPos: nGPAnalogSwitchOut_a
1115	LA_TabPos: nGPArithmetikOut_a
1116	LA_TabPos: nGPMulDivOut_a
1117	LA_TabPos: nGPSignalOut1_a
1118	LA_TabPos: nGPSignalOut2_a
1119	LA_TabPos: nGPSignalOut3_a
1120	LA_TabPos: nGPSignalOut4_a
1121	LA_TabPos: wGPCounter1Out
1122	LA_TabPos: wFreeOut1
1123	LA_TabPos: wFreeOut2
1124	LA_TabPos: wFreeOut3
1125	LA_TabPos: wFreeOut4
1126	LA_TabPos_In: nPosCtrlOutLimit_a
1127	LA_TabPos_In: nPosCtrlPAdapt_a
1128	LA_TabPos: wPosOutUnitsLW
1129	LA_TabPos: wPosOutUnitsHW
1200	LA_SwitchPos: wDriveControlStatus
1201	LA_SwitchPos: wStateDetermFailNoLow
1202	LA_SwitchPos: wStateDetermFailNoHigh
1203	LA_SwitchPos: nMotorCurrent_a
1204	LA_SwitchPos: nMotorSpeedSet_a
1205	LA_SwitchPos: nMotorSpeedAct_a
1206	LA_SwitchPos: nMotorTorqueAct_a
1207	LA_SwitchPos: nDCVoltage_a
1208	LA_SwitchPos: nMotorVoltage_a
1209	LA_SwitchPos: nGPAnalogSwitchOut_a
1210	LA_SwitchPos: nGPArithmetikOut_a
1211	LA_SwitchPos: nGPMulDivOut_a
1212	LA_SwitchPos: nGPSignalOut1_a
1213	LA_SwitchPos: nGPSignalOut2_a
1214	LA_SwitchPos: nGPSignalOut3_a
1215	LA_SwitchPos: nGPSignalOut4_a
1221	LA_SwitchPos: wFreeOut1
1222	LA_SwitchPos: wFreeOut2
1223	LA_SwitchPos: wFreeOut3
1224	LA_SwitchPos: wFreeOut4
16000	LS_AnalogInput: nIn1_a
16001	LS_AnalogInput: nIn2_a
16002	LP_CanIn1: wCtrl
16003	LP_CanIn1: wIn2
16004	LP_CanIn1: wIn3
16005	LP_CanIn1: wIn4
16006	LP_CanIn2: wIn1
16007	LP_CanIn2: wIn2
16008	LP_CanIn2: wIn3
16009	LP_CanIn2: wIn4
16010	LP_CanIn3: wIn1
16011	LP_CanIn3: wIn2
16012	LP_CanIn3: wIn3
16013	LP_CanIn3: wIn4
16014	LS_DigitalInput: wCountIn1_LW
16015	LS_DigitalInput: wCountIn1_HW
16016	LS_DigitalInput: nFreqIn12_a
16017	LS_DigitalInput: nFreqIn12_v
16018	LS_DigitalInput: wCountIn6_LW

Selection list - analog signals	
16019	LS_DigitalInput: wCountIn6_HW
16020	LS_DigitalInput: nFreqIn67_a
16021	LS_DigitalInput: nFreqIn67_v
16100	LS_DataAccess: wOut1
16101	LS_DataAccess: wOut2
16102	LS_DataAccess: wOut3
16103	LS_DataAccess: wOut4
16104	LP_MciIn:wCtrl
16105	LP_MciIn: wIn2
16106	LP_MciIn: wIn3
16107	LP_MciIn: wIn4
16108	LP_MciIn: wIn5
16109	LP_MciIn: wIn6
16110	LP_MciIn: wIn7
16111	LP_MciIn: wIn8
16112	LP_MciIn: wIn9
16113	LP_MciIn: wIn10
16114	LP_MciIn: wIn11
16115	LP_MciIn: wIn12
16116	LP_MciIn: wIn13
16117	LP_MciIn: wIn14
16118	LP_MciIn: wIn15
16119	LP_MciIn: wIn16
16120	LS_Keypad: nTorqueMotLim_a
16121	LS_Keypad: nTorqueGenLim_a
16122	LS_Keypad: nMainSetValue_a
16123	LS_CANManagement: wNodeID
16130	LS_ParReadWrite_1: wOutHWord
16131	LS_ParReadWrite_1: wOutLWord
16132	LS_ParReadWrite_2: wOutHWord
16133	LS_ParReadWrite_2: wOutLWord
16134	LS_ParReadWrite_3: wOutHWord
16135	LS_ParReadWrite_3: wOutLWord
16136	LS_ParReadWrite_4: wOutHWord
16137	LS_ParReadWrite_4: wOutLWord
16138	LS_ParReadWrite_5: wOutHWord
16139	LS_ParReadWrite_5: wOutLWord
16140	LS_ParReadWrite_6: wOutHWord
16141	LS_ParReadWrite_6: wOutLWord
16170	LS_AxisBusIn: wLine1
16171	LS_AxisBusIn: wLine2
16172	LS_AxisBusIn: wLine3
16173	LS_AxisBusIn: wCas1
16174	LS_AxisBusIn: wCas2
16175	LS_AxisBusIn: wCas3
16176	LS_AxisBusIn: wCas4
16177	LS_AxisBusAux: wAuxIn1
16178	LS_AxisBusAux: wAuxIn2
16179	LS_AxisBusAux: wAuxIn3
16180	LS_AxisBusAux: wAuxIn4
16181	LS_AxisBusAux: wSlaveNo
16300	LS_MultiEncoder: nActSpeed_v

Selection list - analog signals	
16301	LS_MultiEncoder: nActSpeed_a
16302	LS_MultiEncoder: wHighWord
16303	LS_MultiEncoder: wLowWord
16320	LS_BusEncoder: nActSpeed_v
16321	LS_BusEncoder: nActSpeed_a
16322	LS_BusEncoder: wHighWord
16323	LS_BusEncoder: wLowWord
16340	LS_Resolver: nActSpeed_v
16341	LS_Resolver: nActSpeed_a
16350	LS_RetainData: wOut1
16351	LS_RetainData: wOut2
16352	LS_RetainData: wOut3
16353	LS_RetainData: wOut4
16360	LP_CanIn4: wIn1
16361	LP_CanIn4: wIn2
16362	LP_CanIn4: wIn3
16363	LP_CanIn4: wIn4
20000	LS_ParFix: nPos100_a
20001	LS_ParFix: nNeg100_a
20002	LS_ParFix: nPos199_99_a
20003	LS_ParFix: nNeg199_99_a
20004	LS_ParFix: w65535
20005	LS_ParFix: wDriveCtrl
20010	LS_ParFree_a: nC472_1_a
20011	LS_ParFree_a: nC472_2_a
20012	LS_ParFree_a: nC472_3_a
20013	LS_ParFree_a: nC472_4_a
20014	LS_ParFree_a: nC472_5_a
20015	LS_ParFree_a: nC472_6_a
20016	LS_ParFree_a: nC472_7_a
20017	LS_ParFree_a: nC472_8_a
20018	LS_ParFree_v: nC473_1_v
20019	LS_ParFree_v: nC473_2_v
20020	LS_ParFree_v: nC473_3_v
20021	LS_ParFree_v: nC473_4_v
20022	LS_ParFree_v: nC473_5_v
20023	LS_ParFree_v: nC473_6_v
20024	LS_ParFree_v: nC473_7_v
20025	LS_ParFree_v: nC473_8_v
20026	LS_ParFree: wC471_1
20027	LS_ParFree: wC471_2
20028	LS_ParFree: wC471_3
20029	LS_ParFree: wC471_4
20030	LS_ParFree: wC471_5
20031	LS_ParFree: wC471_6
20032	LS_ParFree: wC471_7
20033	LS_ParFree: wC471_8
20034	LS_ParFree: wC471_9
20035	LS_ParFree: wC471_10
20036	LS_ParFree: wC471_11
20037	LS_ParFree: wC471_12
20038	LS_ParFree: wC471_13

Selection list - analog signals	
20039	LS_ParFree: wC471_14
20040	LS_ParFree: wC471_15
20041	LS_ParFree: wC471_16
20042	LS_ParFree: wC471_17
20043	LS_ParFree: wC471_18
20044	LS_ParFree: wC471_19
20045	LS_ParFree: wC471_20
20046	LS_ParFree: wC471_21
20047	LS_ParFree: wC471_22
20048	LS_ParFree: wC471_23
20049	LS_ParFree: wC471_24
20050	LS_ParFree: wC471_25
20051	LS_ParFree: wC471_26
20052	LS_ParFree: wC471_27
20053	LS_ParFree: wC471_28
20054	LS_ParFree: wC471_29
20055	LS_ParFree: wC471_30
20056	LS_ParFree: wC471_31
20057	LS_ParFree: wC471_32
20058	LS_ParFree_a: nC472_9_a
20059	LS_ParFree_a: nC472_10_a
20060	LS_ParFree_a: nC472_11_a
20061	LS_ParFree_a: nC472_12_a
20062	LS_ParFree_a: nC472_13_a
20063	LS_ParFree_a: nC472_14_a
20064	LS_ParFree_a: nC472_15_a
20065	LS_ParFree_a: nC472_16_a
20066	LS_ParFree_a_2: nC476_1_a
20067	LS_ParFree_a_2: nC476_2_a
20068	LS_ParFree_a_2: nC476_3_a
20069	LS_ParFree_a_2: nC476_4_a
20070	LS_ParFree_a_2: nC476_5_a
20071	LS_ParFree_a_2: nC476_6_a
20072	LS_ParFree_a_2: nC476_7_a
20073	LS_ParFree_a_2: nC476_8_a
20074	LS_ParFree_a_2: nC476_9_a
20075	LS_ParFree_a_2: nC476_10_a
20076	LS_ParFree_a_2: nC476_11_a
20077	LS_ParFree_a_2: nC476_12_a
20078	LS_ParFree_a_2: nC476_13_a
20079	LS_ParFree_a_2: nC476_14_a
20080	LS_ParFree_a_2: nC476_15_a
20081	LS_ParFree_a_2: nC476_16_a
20082	LS_ParFix_2: nPos100_a
20083	LS_ParFix_2: nNeg100_a
20084	LS_ParFix_2: nPos199_99_a
20085	LS_ParFix_2: nNeg199_99_a
20086	LS_ParFix_2: w65535
20087	LS_ParFix_2: wDriveCtrl
20088	LS_ParFree_2: wC477_1
20089	LS_ParFree_2: wC477_2
20090	LS_ParFree_2: wC477_3

Selection list - analog signals	
20091	LS_ParFree_2: wC477_4
20092	LS_ParFree_2: wC477_5
20093	LS_ParFree_2: wC477_6
20094	LS_ParFree_2: wC477_7
20095	LS_ParFree_2: wC477_8
20096	LS_ParFree_2: wC477_9
20097	LS_ParFree_2: wC477_10
20098	LS_ParFree_2: wC477_11
20099	LS_ParFree_2: wC477_12
20100	LS_ParFree_2: wC477_13
20101	LS_ParFree_2: wC477_14
20102	LS_ParFree_2: wC477_15
20103	LS_ParFree_2: wC477_16
20104	LS_ParFree_2: wC477_17
20105	LS_ParFree_2: wC477_18
20106	LS_ParFree_2: wC477_19
20107	LS_ParFree_2: wC477_20
20108	LS_ParFree_2: wC477_21
20109	LS_ParFree_2: wC477_22
20110	LS_ParFree_2: wC477_23
20111	LS_ParFree_2: wC477_24
20112	LS_ParFree_2: wC477_25
20113	LS_ParFree_2: wC477_26
20114	LS_ParFree_2: wC477_27
20115	LS_ParFree_2: wC477_28
20116	LS_ParFree_2: wC477_29
20117	LS_ParFree_2: wC477_30
20118	LS_ParFree_2: wC477_31
20119	LS_ParFree_2: wC477_32
20120	LS_ParFree_v_2: nC478_1_v
20121	LS_ParFree_v_2: nC478_2_v
20122	LS_ParFree_v_2: nC478_3_v
20123	LS_ParFree_v_2: nC478_4_v
20124	LS_ParFree_v_2: nC478_5_v
20125	LS_ParFree_v_2: nC478_6_v
20126	LS_ParFree_v_2: nC478_7_v
20127	LS_ParFree_v_2: nC478_8_v
20128	LS_DFOut: nOutAct_v
32000	LS_MotorInterface: nMotorSpeedAct_a
32001	LS_MotorInterface: nOutputSpeedCtrl_a
32002	LS_MotorInterface: nInputJerkCtrl_a
32003	LS_MotorInterface: nInputTorqueCtrl_a
32004	LS_MotorInterface: nMotorTorqueAct_a
32005	LS_MotorInterface: nActualFluxx_a
32006	LS_MotorInterface: nDCVoltage_a
32007	LS_MotorInterface: nStatorCurrentIS_a
32008	LS_MotorInterface: nEffCurrentIq_a
32009	LS_MotorInterface: nReaktCurrentId_a
32010	LS_MotorInterface: wMaxMotorSpeed
32011	LS_MotorInterface: wMaxMotorTorque
32012	LS_MotorInterface: nMotorVoltage_a
32013	LS_MotorInterface: nMotorFreqAct_a

Selection list - analog signals	
32014	LS_MotorInterface: nEffSpeedSetValue_a
32015	LS_DeviceMonitor: nMctrlIxtRate_a
32016	LS_DeviceMonitor: nMctrlI2xtRate_a
32017	LS_MotorInterface: nOutputPosCtrl_a
32018	LS_MotorInterface: nHlgSetValue_a
32019	LS_MotorInterface: nMotorSpeedAct_v
32020	LS_MotorInterface: nSpeedCtrlIAct_a
32021	LS_MotorInterface: nVoltageAngleAct_a
32022	LS_DeviceMonitor: nMctrlActiveOuputPower_a
32023	LS_DeviceMonitor: nMctrlApparentOuputPower_a
32024	LS_DeviceMonitor: nMctrlCosinePhiAct_a
32100	LS_DriveInterface: wDeviceStatusWord
32101	LS_DriveInterface: wDeviceAuxStateWord
32102	LS_DriveInterface: wStateDetermFailNoLow
32103	LS_DriveInterface: wStateDetermFailNoHigh
32104	LS_DriveInterface: wStateDetermFailNoShort
32200	LS_MotionControlKernel: nSpeedSet_v
32201	LS_MotionControlKernel: nSpeedCtrlI_a_
32202	LS_MotionControlKernel: nSpeedSetValue_a_
32203	LS_MotionControlKernel: nTorqueSetValue_a_
32204	LS_MotionControlKernel: wActProfileNo
32205	LS_MotionControlKernel: wFollowProfileNo
32206	LS_MotionControlKernel: wMotionState1
32207	LS_MotionControlKernel: wMotionState2
32208	LS_MotionControlKernel: wAuxState
32209	LS_MotionControlKernel: nPWMAngleOffset_
32210	LS_MotionControlKernel: nTorqueLimitAdapt_a_
32211	Reserved: a32211
34900	MCTRL: OszCh1
34901	MCTRL: OszCh2
34902	MCTRL: OszCh3
34903	MCTRL: OszCh4
34904	MCTRL: Status1
34905	MCTRL: Status2
34906	MCTRL: Status3
34907	LS_DeviceMonitor: wUB24V
36000	L_Absolut_1: nOut_a
36001	L_AddSub_1: nOut_a
36002	L_OffsetGain_1: nOut_a
36003	L_OffsetGain_2: nOut_a
36004	L_OffsetGainP_1: nOut_a
36005	L_OffsetGainP_2: nOut_a
36006	L_GainOffset_1: nOut_a
36007	L_GainOffset_2: nOut_a
36008	L_GainOffsetP_1: nOut_a
36009	L_GainOffsetP_2: nOut_a
36010	L_Negation_1: nOut_a
36011	L_Arithmetik_1: nOut_a
36012	L_Arithmetik_2: nOut_a
36013	L_AnalogSwitch_1: nOut_a
36014	L_AnalogSwitch_2: nOut_a
36015	L_AnalogSwitch_3: nOut_a

Selection list - analog signals	
36016	L_Limit_1: nOut_a
36017	L_Limit_2: nOut_a
36018	L_NSet_1: nOut_a
36019	L_MPot_1: nOut_a
36020	L_PCTRL_1: nOut_a
36021	L_SignalMonitor_a: nOut1_a
36022	L_SignalMonitor_a: nOut2_a
36023	L_NLim_1: nOut_a
36024	L_Counter_1: wOut
36025	L_OffsetGainP_3: nOut_a
36026	L_GainOffsetP_3: nOut_a
36027	L_SignalMonitor_a: nOut3_a
36028	L_SignalMonitor_a: nOut4_a
36029	L_MulDiv_1: nOut_a
36030	L_NLim_1: wState
36031	L_NSet_1: wState
36032	L_NSet_1: nSetValue_a
36033	L_PT1_1: nOut_a
36034	L_Absolut_2: nOut_a
36035	L_AnalogSwitch_4: nOut_a
36036	L_AnalogSwitch_5: nOut_a
36037	L_Arithmetik_3: nOut_a
36038	L_Arithmetik_4: nOut_a
36039	L_Arithmetik_5: nOut_a
36045	L_GainOffset_3: nOut_a
36053	L_MulDiv_2: nOut_a
36054	L_Negation_2: nOut_a
36055	L_NLim_2: nOut_a
36056	L_NLim_2: wState
36057	L_OffsetGain_3: nOut_a
36058	L_PT1_2: nOut_a
36059	L_PT1_3: nOut_a
36064	L_SampleHold_1: nOut_a
36065	L_SampleHold_2: nOut_a
36068	L_Counter_2: wOut
36069	L_Counter_3: wOut
36073	L_DT1_1: nOut_a
36074	L_ConvBitsToWord_1: wOut
36075	L_ConvBitsToWord_2: wOut
36076	L_ConvBitsToWord_3: wOut
36077	L_ConvDIntToWords_1: wOutLWord
36078	L_ConvDIntToWords_1: wOutHWord
36079	L_ConvDIntToWords_2: wOutLWord
36080	L_ConvDIntToWords_2: wOutHWord
36081	L_ConvDIntToWords_3: wOutLWord
36082	L_ConvDIntToWords_3: wOutHWord
36083	L_MckCtrlInterface_1: wOutMckPosCtrl_1
36084	L_MckCtrlInterface_1: wOutMckPosCtrl_2
36085	L_MckCtrlInterface_1: wFailState
36086	L_MckStateInterface_1: wOperationMode
36087	L_MckStateInterface_1: wActProfileNo
36088	L_MckStateInterface_1: wActPosMode

Selection list - analog signals	
36089	L_PosiShaftCtrlInterface_1: wOutMckPosCtrl_1
36090	L_PosiShaftCtrlInterface_1: wOutMckPosCtrl_2
36091	L_PCTRL_1: nPIDOut1_a
36092	L_PCTRL_1: nPIDOut2_a
36093	L_PCTRL_1: nInfluenceOut_a
36094	L_Curve_1: nOut_a
36095	L_Interpolator_1: nPhdOut_v
36096	L_Interpolator_1: nNOut_a
36097	L_ConvW_1: wOut
36098	L_ConvW_2: wOut
36099	L_ConvW_3: wOut
36100	L_ConvW_4: wOut
36101	L_SRFG_1: nOut_a
36102	L_SRFG_2: nOut_a
36103	L_MckStateInterface_1: wPosUnitsLW
36104	L_MckStateInterface_1: wPosUnitsHW
36105	L_SignalSwitch_1: wOut
36106	L_SignalSwitch_2: wOut
36107	L_SignalSwitch_3: wOut
36108	L_SignalSwitch_4: wOut
36109	L_SRFG_1: nDeltaOut_a
36110	L_SRFG_2: nDeltaOut_a
36111	L_Odometer_1: wLastMeasure
36112	L_FixSet_a_1: nOut_a
36113	L_FixSet_a_1: wSelect
36114	L_FixSet_w_1: wOut
36115	L_FixSet_w_1: wSelect
36116	L_FixSet_w_2: wOut
36117	L_FixSet_w_2: wSelect
36120	L_CalcDiameter_1: nDiameter_a
36121	L_CalcDiameter_1: nReziprDiameter_a
36122	L_CalcDiameter_1: nNReel_v
36123	L_CalcDiameter_1: nDMin_a
36124	L_ProcessCtrl_1: nOut_a
36125	L_ProcessCtrl_1: nOutRed_a
36126	L_ProcessCtrl_1: nDeviation_a
36127	L_ProcessCtrl_1: nDComponent
36128	L_PhiIntegrator_1: nOut16
36129	L_PhiIntegrator_1: nSpeedGearAdd_v
36130	L_PosCtrlLin_1: nNOut_v
36131	L_PosCtrlLin_2: nNOut_v
36132	L_PhiIntegrator_1: nSpeedGear_v
36133	L_DFSET_1: nSetGain_v
36134	L_DFSET_1: nSetGearTrim_v
36135	L_DFSET_1: nSetGearTrim_a
36136	L_DFRFG_1: nOut_v
36137	L_ConvPA_1: nOut_a
36138	L_ConvPA_2: nOut_a
36139	L_ConvPA_3: nOut_a
36140	L_ConvX_1: nOut
36141	L_ConvX_2: nOut
36142	L_ConvX_3: nOut

Selection list - analog signals	
36143	LS_MotionControlKernel: wGearNum
36144	LS_MotionControlKernel: wGearDenom
36145	L_DFSET_1: nSpeedSetOut_v
36146	L_Curve_2: nOut_a
36147	L_Curve_3: nOut_a
36148	L_Curve_3: nCurveValue_a
36150	L_Sequencer_1: wMCKPosCtrl1
36151	L_Sequencer_1: wMCKPosCtrl2
36152	L_Sequencer_1: wAuxCtrl
36153	L_Sequencer_1: wState
36154	L_Sequencer_1: wActStep
36155	L_Sequencer_1: nSet_a
36156	L_Sequencer_1: wProfileNumber
36157	L_Sequencer_1: wDigitalOutputs
36158	L_ConvActPos_1: nPosOut_a
36159	L_ConvActPos_1: nVAdditive_a
36160	L_ConvActPos_1: nVOut_a
36161	L_ConvActPos_1: nMAdditive_a
36162	L_MFail_1: nNOut_a
42000	LA_NCtrl_In: wCANDriveControl
42001	LA_NCtrl_In: wMCIDriveControl
42002	LA_NCtrl_In: nTorqueMotLim_a
42003	LA_NCtrl_In: nTorqueGenLim_a
42004	LA_NCtrl_In: nPIDVpAdapt_a
42005	LA_NCtrl_In: nPIDActValue_a
42006	LA_NCtrl_In: nMainSetValue_a
42007	LA_NCtrl_In: nAuxSetValue_a
42008	LA_NCtrl_In: nGPAnalogSwitchIn1_a
42009	LA_NCtrl_In: nGPAnalogSwitchIn2_a
42010	LA_NCtrl_In: nGPArithmetikIn1_a
42011	LA_NCtrl_In: nGPArithmetikIn2_a
42012	LA_NCtrl_In: nGPMulDivIn_a
42013	LA_NCtrl_In: nGPCompareIn1_a
42014	LA_NCtrl_In: nGPCompareIn2_a
42015	LA_NCtrl_In: nVoltageAdd_a
42016	LA_NCtrl_In: nPIDInfluence_a
42017	LA_NCtrl_In: nPIDSetValue_a
42018	LA_NCtrl_In: nPWMAngleOffset
42019	LA_NCtrl_In: nBoost_a
42020	LA_NCtrl_In: wSMControl
42025	LA_NCtrl_In: wFreeIn1
42026	LA_NCtrl_In: wFreeIn2
42027	LA_NCtrl_In: wFreeIn3
42028	LA_NCtrl_In: wFreeIn4
42100	LA_TabPos_In: wCanDriveControl
42101	LA_TabPos_In: wMciDriveControl
42102	LA_TabPos_In: nTorqueMotLim_a
42103	LA_TabPos_In: nTorqueGenLim_a
42104	LA_TabPos_In: nMainSetValue_a
42105	LA_TabPos_In: nAuxSetValue_a
42106	LA_TabPos_In: wMckCtrl1
42107	LA_TabPos_In: wMckCtrl2

Selection list - analog signals	
42108	LA_TabPos_In: wMckOperationMode
42109	LA_TabPos_In: wPosProfileMode
42110	LA_TabPos_In: wPosProfileNo
42111	LA_TabPos_In: nGPAnalogSwitchIn1_a
42112	LA_TabPos_In: nGPAnalogSwitchIn2_a
42113	LA_TabPos_In: nGPArithmetikIn1_a
42114	LA_TabPos_In: nGPArithmetikIn2_a
42115	LA_TabPos_In: nGPMulDivIn_a
42116	LA_TabPos_In: nGPCompareIn1_a
42117	LA_TabPos_In: nGPCompareIn2_a
42118	LA_TabPos_In: wGPCounter1LdVal
42119	LA_TabPos_In: wGPCounter1CmpVal
42120	LA_TabPos_In: nSpeedOverride_a
42121	LA_TabPos_In: nAccOverride_a
42122	LA_TabPos_In: wFreeIn1
42123	LA_TabPos_In: wFreeIn2
42124	LA_TabPos_In: wFreeIn3
42125	LA_TabPos_In: wFreeIn4
42126	LA_TabPos_In: wSMControl
42127	LA_TabPos_In: wPosProfileUnitsLW
42128	LA_TabPos_In: wPosProfileUnitsHW
42200	LA_SwitchPos_In: wCANDriveControl
42201	LA_SwitchPos_In: wMCIDriveControl
42202	LA_SwitchPos_In: nVoltageAdd_a
42203	LA_SwitchPos_In: nBoost_a
42204	LA_SwitchPos_In: nPWMAngleOffset
42205	LA_SwitchPos_In: nTorqueMotLim_a
42206	LA_SwitchPos_In: nTorqueGenLim_a
42207	LA_SwitchPos_In: nMainSetValue_a
42208	LA_SwitchPos_In: nAuxSetValue_a
42209	LA_SwitchPos_In: nGPAnalogSwitchIn1_a
42210	LA_SwitchPos_In: nGPAnalogSwitchIn2_a
42211	LA_SwitchPos_In: nGPArithmetikIn1_a
42212	LA_SwitchPos_In: nGPArithmetikIn2_a
42213	LA_SwitchPos_In: nGPMulDivIn_a
42214	LA_SwitchPos_In: nGPCompareIn1_a
42215	LA_SwitchPos_In: nGPCompareIn2_a
42216	LA_SwitchPos_In: wSMControl
42221	LA_SwitchPos_In: wFreeIn1
42222	LA_SwitchPos_In: wFreeIn2
42223	LA_SwitchPos_In: wFreeIn3
42224	LA_SwitchPos_In: wFreeIn4

17.5 Selection list - digital signals

This selection list is relevant for the following configuration parameters:

Parameters	
C00411	L_SignalMonitor_b: Signal sources
C00621	System connection list: Bool
C00701	LA_NCtrl: Digital connection list
C00711	LA_TabPos: Digital connection list
C00761	LA_SwitchPos: Digital connection list

Selection list - digital signals	
0	Not connected
1000	LA_NCtrl: bDriveReady
1001	LA_NCtrl: bDriveFail
1002	LA_NCtrl: bCInhActive
1003	LA_NCtrl: bQSPIsActive
1004	LA_NCtrl: bSpeedCcw
1005	LA_NCtrl: bSpeedActCompare
1008	LA_NCtrl: bGPDigitalDelayOut
1009	LA_NCtrl: bGPLogicOut
1010	LA_NCtrl: bGPSignalOut1
1011	LA_NCtrl: bGPSignalOut2
1012	LA_NCtrl: bGPSignalOut3
1013	LA_NCtrl: bGPSignalOut4
1014	LA_NCtrl: bOverLoadActive
1015	LA_NCtrl: bMBrakeReleaseOut
1016	LA_NCtrl: bMBrakeReleased
1017	LA_NCtrl: bGPCompareOut
1018	LA_NCtrl: bUnderLoadActive
1019	LA_NCtrl: bImaxActive
1020	LA_NCtrl: bSpeedSetReached
1021	LA_NCtrl: bSpeedActEqSet
1022	LA_NCtrl: bGPDFlipFlop_Out
1023	LA_NCtrl: bGPDFlipFlop_NegOut
1029	LA_NCtrl: bFreeOut1
1030	LA_NCtrl: bFreeOut2
1031	LA_NCtrl: bFreeOut3
1032	LA_NCtrl: bFreeOut4
1033	LA_NCtrl: bFreeOut5
1034	LA_NCtrl: bFreeOut6
1035	LA_NCtrl: bFreeOut7
1036	LA_NCtrl: bFreeOut8
1100	LA_TabPos: bDriveFail
1101	LA_TabPos: bDriveReady
1102	LA_TabPos: bCInhActive
1103	LA_TabPos: bQSPIsActive
1104	LA_TabPos: bSpeedCcw
1105	LA_TabPos: bSpeedActCompare
1106	LA_TabPos: bImaxActive
1107	LA_TabPos: bSpeedSetReached
1108	LA_TabPos: bMBrakeReleaseOut
1109	LA_TabPos: bMBrakeReleased
1110	LA_TabPos: bHomeDone
1111	LA_TabPos: bHomePosAvailable
1112	LA_TabPos: bProfileDone
1113	LA_TabPos: bProfileBusy
1114	LA_TabPos: bAccelerating
1115	LA_TabPos: bConstantDuty
1116	LA_TabPos: bDecelerating
1117	LA_TabPos: bDwellTime
1118	LA_TabPos: bInTarget
1119	LA_TabPos: bGPDigitalDelayOut
1120	LA_TabPos: bGPLogicOut
1121	LA_TabPos: bGPCompareOut
1122	LA_TabPos: bGPSignalOut1
1123	LA_TabPos: bGPSignalOut2
1124	LA_TabPos: bGPSignalOut3
1125	LA_TabPos: bGPSignalOut4
1126	LA_TabPos: bGPDFlipFlop_Out
1127	LA_TabPos: bGPDFlipFlop_NegOut
1128	LA_TabPos: bGPCounter1Equal
1129	LA_TabPos: bFreeOut1
1130	LA_TabPos: bFreeOut2
1131	LA_TabPos: bFreeOut3
1132	LA_TabPos: bFreeOut4
1133	LA_TabPos: bFreeOut5
1134	LA_TabPos: bFreeOut6
1135	LA_TabPos: bFreeOut7
1136	LA_TabPos: bFreeOut8
1200	LA_SwitchPos: bDriveFail
1201	LA_SwitchPos: bWarningActive
1202	LA_SwitchPos: bSafeTorqueOff
1203	LA_SwitchPos: bDriveReady
1204	LA_SwitchPos: bCInhActive
1205	LA_SwitchPos: bImpIsActive
1206	LA_SwitchPos: bQSPIsActive
1207	LA_SwitchPos: bSpeedCcw
1208	LA_SwitchPos: bSpeedActCompare
1209	LA_SwitchPos: bImaxActive
1210	LA_SwitchPos: bSpeedSetReached
1211	LA_SwitchPos: bSpeedActEqSet
1212	LA_SwitchPos: bMBrakeReleaseOut
1213	LA_SwitchPos: bMBrakeReleased
1214	LA_SwitchPos: bGPDigitalDelayOut
1215	LA_SwitchPos: bGPLogicOut
1216	LA_SwitchPos: bGPCompareOut
1217	LA_SwitchPos: bGPDFlipFlop_Out
1218	LA_SwitchPos: bGPDFlipFlop_NegOut
1219	LA_SwitchPos: bGPSignalOut1
1220	LA_SwitchPos: bGPSignalOut2
1221	LA_SwitchPos: bGPSignalOut3

Selection list - digital signals	
1222	LA_SwitchPos: bGPSignalOut4
1228	LA_SwitchPos: bFreeOut1
1229	LA_SwitchPos: bFreeOut2
1230	LA_SwitchPos: bFreeOut3
1231	LA_SwitchPos: bFreeOut4
1232	LA_SwitchPos: bFreeOut5
1233	LA_SwitchPos: bFreeOut6
1234	LA_SwitchPos: bFreeOut7
1235	LA_SwitchPos: bFreeOut8
16000	LS_DigitalInput: bIn1
16001	LS_DigitalInput: bIn2
16002	LS_DigitalInput: bIn3
16003	LS_DigitalInput: bIn4
16004	LS_DigitalInput: bIn5
16005	LS_DigitalInput: bIn6
16006	LS_DigitalInput: bIn7
16008	LS_DigitalInput: bCInh
16009	LS_DigitalInput: bCountIn1_Compare
16010	LS_DigitalInput: bCountIn6_Compare
16011	LS_AnalogInput: bCurrentErrorIn1
16012	LS_AnalogInput: bCurrentErrorIn2
16013	LP_CanIn1: bCtrl1_B0
16014	LP_CanIn1: bCtrl1_B1
16015	LP_CanIn1: bCtrl1_B2
16016	LP_CanIn1: bCtrl1_B3
16017	LP_CanIn1: bCtrl1_B4
16018	LP_CanIn1: bCtrl1_B5
16019	LP_CanIn1: bCtrl1_B6
16020	LP_CanIn1: bCtrl1_B7
16021	LP_CanIn1: bCtrl1_B8
16022	LP_CanIn1: bCtrl1_B9
16023	LP_CanIn1: bCtrl1_B10
16024	LP_CanIn1: bCtrl1_B11
16025	LP_CanIn1: bCtrl1_B12
16026	LP_CanIn1: bCtrl1_B13
16027	LP_CanIn1: bCtrl1_B14
16028	LP_CanIn1: bCtrl1_B15
16029	LP_CanIn2: bIn1_B0
16030	LP_CanIn2: bIn1_B1
16031	LP_CanIn2: bIn1_B2
16032	LP_CanIn2: bIn1_B3
16033	LP_CanIn2: bIn1_B4
16034	LP_CanIn2: bIn1_B5
16035	LP_CanIn2: bIn1_B6
16036	LP_CanIn2: bIn1_B7
16037	LP_CanIn2: bIn1_B8
16038	LP_CanIn2: bIn1_B9
16039	LP_CanIn2: bIn1_B10
16040	LP_CanIn2: bIn1_B11
16041	LP_CanIn2: bIn1_B12
16042	LP_CanIn2: bIn1_B13
16043	LP_CanIn2: bIn1_B14

Selection list - digital signals	
16044	LP_CanIn2: bIn1_B15
16045	LP_CanIn3: bIn1_B0
16046	LP_CanIn3: bIn1_B1
16047	LP_CanIn3: bIn1_B2
16048	LP_CanIn3: bIn1_B3
16049	LP_CanIn3: bIn1_B4
16050	LP_CanIn3: bIn1_B5
16051	LP_CanIn3: bIn1_B6
16052	LP_CanIn3: bIn1_B7
16053	LP_CanIn3: bIn1_B8
16054	LP_CanIn3: bIn1_B9
16055	LP_CanIn3: bIn1_B10
16056	LP_CanIn3: bIn1_B11
16057	LP_CanIn3: bIn1_B12
16058	LP_CanIn3: bIn1_B13
16059	LP_CanIn3: bIn1_B14
16060	LP_CanIn3: bIn1_B15
16061	LP_MciIn: bCtrl_B0
16062	LP_MciIn: bCtrl_B1
16063	LP_MciIn: bCtrl_B2
16064	LP_MciIn: bCtrl_B3
16065	LP_MciIn: bCtrl_B4
16066	LP_MciIn: bCtrl_B5
16067	LP_MciIn: bCtrl_B6
16068	LP_MciIn: bCtrl_B7
16069	LP_MciIn: bCtrl_B8
16070	LP_MciIn: bCtrl_B9
16071	LP_MciIn: bCtrl_B10
16072	LP_MciIn: bCtrl_B11
16073	LP_MciIn: bCtrl_B12
16074	LP_MciIn: bCtrl_B13
16075	LP_MciIn: bCtrl_B14
16076	LP_MciIn: bCtrl_B15
16077	LP_MciIn: bIn2_B0
16078	LP_MciIn: bIn2_B1
16079	LP_MciIn: bIn2_B2
16080	LP_MciIn: bIn2_B3
16081	LP_MciIn: bIn2_B4
16082	LP_MciIn: bIn2_B5
16083	LP_MciIn: bIn2_B6
16084	LP_MciIn: bIn2_B7
16085	LP_MciIn: bIn2_B8
16086	LP_MciIn: bIn2_B9
16087	LP_MciIn: bIn2_B10
16088	LP_MciIn: bIn2_B11
16089	LP_MciIn: bIn2_B12
16090	LP_MciIn: bIn2_B13
16091	LP_MciIn: bIn2_B14
16092	LP_MciIn: bIn2_B15
16093	LS_Keypad: bSetQuickstop
16094	LS_Keypad: bSetDCBrake
16095	LS_Keypad: bSetSpeedCcw

Selection list - digital signals	
16096	LS_Keypad: bJogSpeed1
16097	LS_Keypad: bJogSpeed2
16098	LS_Keypad: bMPotEnable
16099	LS_Keypad: bMPotUp
16100	LS_Keypad: bMPotDown
16101	LS_DigitalInput: bPosIn12_State
16102	LS_DigitalInput: bOutHC_BrakeApplied
16110	LS_ParReadWrite_1: bDone
16111	LS_ParReadWrite_1: bFail
16112	LS_ParReadWrite_2: bDone
16113	LS_ParReadWrite_2: bFail
16114	LS_ParReadWrite_3: bDone
16115	LS_ParReadWrite_3: bFail
16116	LS_ParReadWrite_4: bDone
16117	LS_ParReadWrite_4: bFail
16118	LS_ParReadWrite_5: bDone
16119	LS_ParReadWrite_5: bFail
16120	LS_ParReadWrite_6: bDone
16121	LS_ParReadWrite_6: bFail
16122	LS_WriteParamList: bDone
16123	LS_WriteParamList: bFail
16161	LS_CANManagement: bFail
16162	LS_CANManagement: bOperational
16165	LS_MultiEncoder: bFail
16166	LS_Resolver: bFail
16200	LS_SyncManagement: bSyncSignalOK
16201	LS_SyncManagement: bSyncPhaseOK
16202	LS_MultiEncoder: bState
16205	LS_MultiEncoder: bState2
16206	LS_MultiEncoder: bState3
16303	LS_TouchProbe: bTPDigIn3Received
16304	LS_TouchProbe: bTPDigIn4Received
16305	LS_TouchProbe: bTPDigIn5Received
16306	LS_TouchProbe: bTPDigIn6Received
16307	LS_TouchProbe: bTPDigIn7Received
16308	LS_AxisBusIO: bFail_DigIn
16309	LS_AxisBusIO: bResetFail_In
16310	LS_RetainData: bOut1
16311	LS_RetainData: bOut2
16312	LS_RetainData: bOut3
16313	LS_RetainData: bOut4
16314	LS_TouchProbe: bTPEncoderReceived
16315	LS_TouchProbe: bTPResolverReceived
16320	LS_AxisBusAux: bDone
16321	LS_AxisBusAux: bFail
16400	LP_CanIn4: bIn1_B0
16401	LP_CanIn4: bIn1_B1
16402	LP_CanIn4: bIn1_B2
16403	LP_CanIn4: bIn1_B3
16404	LP_CanIn4: bIn1_B4
16405	LP_CanIn4: bIn1_B5
16406	LP_CanIn4: bIn1_B6

Selection list - digital signals	
16407	LP_CanIn4: bIn1_B7
16408	LP_CanIn4: bIn1_B8
16409	LP_CanIn4: bIn1_B9
16410	LP_CanIn4: bIn1_B10
16411	LP_CanIn4: bIn1_B11
16412	LP_CanIn4: bIn1_B12
16413	LP_CanIn4: bIn1_B13
16414	LP_CanIn4: bIn1_B14
16415	LP_CanIn4: bIn1_B15
20000	LS_ParFix: bTrue
20001	LS_ParFree_b: bC470_1
20002	LS_ParFree_b: bC470_2
20003	LS_ParFree_b: bC470_3
20004	LS_ParFree_b: bC470_4
20005	LS_ParFree_b: bC470_5
20006	LS_ParFree_b: bC470_6
20007	LS_ParFree_b: bC470_7
20008	LS_ParFree_b: bC470_8
20009	LS_ParFree_b: bC470_9
20010	LS_ParFree_b: bC470_10
20011	LS_ParFree_b: bC470_11
20012	LS_ParFree_b: bC470_12
20013	LS_ParFree_b: bC470_13
20014	LS_ParFree_b: bC470_14
20015	LS_ParFree_b: bC470_15
20016	LS_ParFree_b: bC470_16
20017	LS_ParFree_b: bC470_17
20018	LS_ParFree_b: bC470_18
20019	LS_ParFree_b: bC470_19
20020	LS_ParFree_b: bC470_20
20021	LS_ParFree_b: bC470_21
20022	LS_ParFree_b: bC470_22
20023	LS_ParFree_b: bC470_23
20024	LS_ParFree_b: bC470_24
20025	LS_ParFree_b: bC470_25
20026	LS_ParFree_b: bC470_26
20027	LS_ParFree_b: bC470_27
20028	LS_ParFree_b: bC470_28
20029	LS_ParFree_b: bC470_29
20030	LS_ParFree_b: bC470_30
20031	LS_ParFree_b: bC470_31
20032	LS_ParFree_b: bC470_32
20033	LS_PulseGenerator: b100Hz
20034	LS_PulseGenerator: b10Hz
20035	LS_PulseGenerator: b2Hz
20036	LS_PulseGenerator: b1Hz
20037	LS_PulseGenerator: b1HzFlash
20038	LS_PulseGenerator: b2HzFlash
20039	LS_PulseGenerator: bSingleFlash1
20040	LS_PulseGenerator: bSingleFlash2
20041	LS_PulseGenerator: bDoubleFlash
20042	LS_PulseGenerator: bSquareWave

Selection list - digital signals	
20043	LS_PulseGenerator: bFirstCycleDone
20044	LS_ParFix_2: bTrue
20045	LS_ParFreeUnit_1: bDataValid
20046	LS_ParFreeUnit_2: bDataValid
32000	LS_DeviceMonitor: bMctrlFanFault
32001	LS_DeviceMonitor: bMctrlHeatSinkTemp
32002	LS_MotorInterface: bLimPosCtrlOut
32003	LS_MotorInterface: bLimSpeedCtrlOut
32004	LS_MotorInterface: bLimSpeedSetVal
32005	LS_MotorInterface: bLimTorqueSetVal
32006	LS_MotorInterface: bLimCurrentSetVal
32007	LS_DeviceMonitor: bMctrlUVDetected
32008	LS_DeviceMonitor: bMctrlOVDetected
32009	LS_DeviceMonitor: bMctrlMotorPhaseFault
32010	LS_DeviceMonitor: bMctrlEncoderComFault
32011	LS_DeviceMonitor: bMctrlIxtOverload
32012	LS_DeviceMonitor: bMctrlI2xtOverload
32013	LS_MotorInterface: bIdentificationActive
32014	LS_MotorInterface: bFlyingSyncActive
32015	LS_DeviceMonitor: bMctrlTorqueMax
32016	LS_DeviceMonitor: bMctrlNMax
32017	LS_DeviceMonitor: bMctrlFChopReduced
32018	LS_DeviceMonitor: bMctrlMotorPtc
32019	LS_DeviceMonitor: bMctrlMotorTemp
32020	LS_MotorInterface: bDcbActive
32021	LS_DeviceMonitor: bMctrlBrakeChopperFault
32022	LS_MotorInterface: bQspIsActive
32023	LS_MotorInterface: bHlgLoad
32024	LS_MotorInterface: bHlgStop
32025	LS_DeviceMonitor: bMctrlImpActive
32026	LS_DeviceMonitor: bMctrlClampActive
32027	LS_DeviceMonitor: bMctrlMainsFault
32028	LS_DeviceMonitor: bMctrlNmaxForFChop
32029	LS_DeviceMonitor: bMctrlShortCircuit
32030	LS_DeviceMonitor: bMctrlEarthFault
32031	LS_DeviceMonitor: bWirebreakUfLinearActive
32032	LS_DeviceMonitor: bCurrentMonitoringOverload
32033	LS_DeviceMonitor: bSlpsmSpeedOpenLoopControl
32100	LS_DriveInterface: bInit
32101	LS_DriveInterface: bReady
32102	LS_DriveInterface: bReadyToSwitchOn
32103	LS_DriveInterface: bOperationEnable
32104	LS_DriveInterface: bWarning
32105	LS_DriveInterface: bTrouble
32106	LS_DriveInterface: bFail
32107	LS_DriveInterface: bCollectedFail
32108	LS_DriveInterface: bSafeTorqueOff
32109	LS_DriveInterface: bImpIsActive
32110	LS_DriveInterface: bCInhIsActive
32111	LS_DriveInterface: bSafetyIsActive
32112	LS_DriveInterface: bCwCcw
32113	LS_DriveInterface: bNActCompare

Selection list - digital signals	
32200	LS_MotionControlKernel: bPosCtrlOn_
32201	LS_MotionControlKernel: bSpeedCtrlIOn_
32202	LS_MotionControlKernel: bTorquemodeOn_
32203	LS_MotionControlKernel: bDcBrakeOn_
32204	LS_MotionControlKernel: bMBrakeReleaseOut
32205	LS_MotionControlKernel: bMBrakeReleased
32206	LS_MotionControlKernel: bDeltaPosOn_
32207	LS_MotionControlKernel: bPosDerivativeOn_
32208	LS_MotionControlKernel: bMotorRefOffsetOn_
32209	LS_MotionControlKernel: bQspOn_
32210	LS_MotionControlKernel: bPosBusy
32211	LS_MotionControlKernel: bPosDone
32212	LS_MotionControlKernel: bHomDone
32213	LS_MotionControlKernel: bHomAvailable
32214	LS_MotionControlKernel: bTorqueLimitAdaptOn_
32215	Reserved: b32215
32216	Reserved: b32216
32217	Reserved: b32217
32218	Reserved: b32218
32219	LS_MotionControlKernel: bFollowErrLim1
32220	LS_MotionControlKernel: bFollowErrLim2
32221	LS_MotionControlKernel: bReadyToOperate
36000	L_And_1: bOut
36001	L_And_2: bOut
36002	L_And_3: bOut
36003	L_Or_1: bOut
36004	L_Or_2: bOut
36005	L_Or_3: bOut
36006	L_Not_1: bOut
36007	L_Not_2: bOut
36008	L_Not_3: bOut
36009	L_DFlipFlop_1: bOut
36010	L_RLQ_1: bQsp
36011	L_RLQ_1: bCwCcw
36012	L_DigitalDelay_1: bOut
36013	L_Compare_1: bOut
36014	L_Compare_2: bOut
36015	L_Compare_3: bOut
36016	L_NSet_1: bRfgEqO
36017	L_DigitalLogic_1: bOut
36018	L_Counter_1: bEqual
36019	L_SignalMonitor_b: bOut1
36020	L_SignalMonitor_b: bOut2
36021	L_SignalMonitor_b: bOut3
36022	L_SignalMonitor_b: bOut4
36023	L_PCTRL_1: bActEqSet
36024	L_NLim_1: bLimitActive
36025	L_DFlipFlop_1: bNegOut
36026	L_And5_1: bOut
36027	L_And5_2: bOut
36028	L_Compare_4: bOut
36029	L_Compare_5: bOut

Selection list - digital signals	
36030	L_ComparePhi_1: bOut
36031	L_ComparePhi_2: bOut
36032	L_ComparePhi_3: bOut
36033	L_ComparePhi_4: bOut
36034	L_ComparePhi_5: bOut
36035	L_DFlipFlop_2: bOut
36036	L_DFlipFlop_2: bNegOut
36037	L_DigitalDelay_2: bOut
36038	L_DigitalDelay_3: bOut
36039	L_Or_4: bOut
36040	L_DigitalLogic_3: bOut
36041	L_DigitalLogic_2: bOut
36042	L_DigitalLogic5_1: bOut
36043	L_DigitalLogic5_2: bOut
36044	L_NLim_2: bLimitActive
36045	L_Or5_1: bOut
36046	L_Or5_2: bOut
36047	L_Not_4: bOut
36048	L_Not_5: bOut
36049	L_Not_6: bOut
36050	L_Not_7: bOut
36055	L_PhaseIntK_1: bState
36056	L_PhaseIntK_2: bState
36057	L_Counter_2: bEqual
36058	L_Counter_3: bEqual
36059	L_RSFlipFlop_1: bOut
36060	L_RSFlipFlop_1: bNegOut
36061	L_RSFlipFlop_2: bOut
36062	L_RSFlipFlop_2: bNegOut
36063	L_Transient_5: bOut
36064	L_Transient_6: bOut
36065	L_Transient_7: bOut
36066	L_Transient_8: bOut
36067	L_ConvWordToBits_1: bBit0
36068	L_ConvWordToBits_1: bBit1
36069	L_ConvWordToBits_1: bBit2
36070	L_ConvWordToBits_1: bBit3
36071	L_ConvWordToBits_1: bBit4
36072	L_ConvWordToBits_1: bBit5
36073	L_ConvWordToBits_1: bBit6
36074	L_ConvWordToBits_1: bBit7
36075	L_ConvWordToBits_1: bBit8
36076	L_ConvWordToBits_1: bBit9
36077	L_ConvWordToBits_1: bBit10
36078	L_ConvWordToBits_1: bBit11
36079	L_ConvWordToBits_1: bBit12
36080	L_ConvWordToBits_1: bBit13
36081	L_ConvWordToBits_1: bBit14
36082	L_ConvWordToBits_1: bBit15
36083	L_ConvWordToBits_2: bBit0
36084	L_ConvWordToBits_2: bBit1
36085	L_ConvWordToBits_2: bBit2

Selection list - digital signals	
36086	L_ConvWordToBits_2: bBit3
36087	L_ConvWordToBits_2: bBit4
36088	L_ConvWordToBits_2: bBit5
36089	L_ConvWordToBits_2: bBit6
36090	L_ConvWordToBits_2: bBit7
36091	L_ConvWordToBits_2: bBit8
36092	L_ConvWordToBits_2: bBit9
36093	L_ConvWordToBits_2: bBit10
36094	L_ConvWordToBits_2: bBit11
36095	L_ConvWordToBits_2: bBit12
36096	L_ConvWordToBits_2: bBit13
36097	L_ConvWordToBits_2: bBit14
36098	L_ConvWordToBits_2: bBit15
36099	L_ConvWordToBits_3: bBit0
36100	L_ConvWordToBits_3: bBit1
36101	L_ConvWordToBits_3: bBit2
36102	L_ConvWordToBits_3: bBit3
36103	L_ConvWordToBits_3: bBit4
36104	L_ConvWordToBits_3: bBit5
36105	L_ConvWordToBits_3: bBit6
36106	L_ConvWordToBits_3: bBit7
36107	L_ConvWordToBits_3: bBit8
36108	L_ConvWordToBits_3: bBit9
36109	L_ConvWordToBits_3: bBit10
36110	L_ConvWordToBits_3: bBit11
36111	L_ConvWordToBits_3: bBit12
36112	L_ConvWordToBits_3: bBit13
36113	L_ConvWordToBits_3: bBit14
36114	L_ConvWordToBits_3: bBit15
36115	L_MckCtrlInterface_1: bFail
36116	L_MckStateInterface_1: bAccelerating
36117	L_MckStateInterface_1: bDecelerating
36118	L_MckStateInterface_1: bConstantSpeedDuty
36119	L_MckStateInterface_1: bSShaping
36120	L_MckStateInterface_1: bBusy
36121	L_MckStateInterface_1: bDone
36122	L_MckStateInterface_1: bHomingPosAvailable
36123	L_MckStateInterface_1: bHomingDone
36124	L_MckStateInterface_1: bDwellTime
36125	L_MckStateInterface_1: bInTarget
36126	L_MckStateInterface_1: bPosDone
36127	L_PosiShaftCtrlInterface_1: bQsp
36128	L_PosiShaftCtrlInterface_1: bRsp
36129	L_PosiShaftCtrlInterface_1: bTripSet
36130	L_PosiShaftCtrlInterface_1: bTripReset
36131	L_JogCtrlExtension_1: bRfgOut
36132	L_JogCtrlExtension_1: bJog1Out
36133	L_JogCtrlExtension_1: bJog2Out
36134	L_Curve_1: bLimit
36135	L_Interpolator_1: bIPulse
36136	L_Interpolator_1: bSignalError
36137	L_MckCtrlInterface_1: bPosSetDataValid

Selection list - digital signals	
36138	L_Transient_1: bOut
36139	L_Transient_2: bOut
36140	L_Transient_3: bOut
36141	L_Transient_4: bOut
36142	L_CalcDiameter_1: bDMaxLimit
36143	L_CalcDiameter_1: bDMinLimit
36144	L_CalcDiameter_1: bWebBreak
36145	L_PhiIntegrator_1: bOvfl32
36146	L_PhiIntegrator_1: bOvfl16
36147	L_PhiIntegrator_1: bInitDone
36148	L_PosCtrlLin_1: bInTarget
36149	L_PosCtrlLin_2: bInTarget
36150	L_SwitchPoint_1: bOut1
36151	L_SwitchPoint_1: bOut2
36152	L_SwitchPoint_1: bOut3
36153	L_SwitchPoint_1: bOut4
36154	L_DFSET_1: bAck
36155	L_DFSET_1: bFollowingErr
36156	L_DFSET_1: bPosOverflow
36157	L_DFRFG_1: bSync
36158	L_DFRFG_1: bFail
36159	L_Curve_2: bLimit
36160	L_Curve_3: bLimit
36161	L_Sequencer_1: bProgramBusy
36162	L_Sequencer_1: bStateReady
36163	L_Sequencer_1: bStateRun
36164	L_Sequencer_1: bStatePause
36165	L_Sequencer_1: bStateBreak
36166	L_Sequencer_1: bStateReset
36167	L_Sequencer_1: bStateDone
36168	L_Sequencer_1: bErrActive
36169	L_Sequencer_1: bWatchdogActive
36170	L_Sequencer_1: bErrBranch
36171	L_ConvUnitsToIncr_1: bDataValid
36172	L_ConvUnitsToIncr_2: bDataValid
36173	L_ConvUnitsToIncr_3: bDataValid
36174	L_ConvActPos_1: bPosOverrange
36175	L_ConvActPos_1: bInSetPosition
36176	L_ConvActPos_1: bNegOverrange
36177	L_MFail_1: bActive
36178	L_MFail_1: bNCtrlIReset
36179	L_Curve_3: bLimit
36180	L_SwitchPoint_1: bOut5
36181	L_SwitchPoint_1: bOut6
36182	L_SwitchPoint_1: bOut7
36183	L_SwitchPoint_1: bOut8
36184	L_SwitchPointPar_1: bOut1
36185	L_SwitchPointPar_1: bOut2
36186	L_SwitchPointPar_1: bOut3
36187	L_SwitchPointPar_1: bOut4
36188	L_SwitchPointPar_1: bOut5
36189	L_SwitchPointPar_1: bOut6

Selection list - digital signals	
36190	L_SwitchPointPar_1: bOut7
36191	L_SwitchPointPar_1: bOut8
36284	L_MckStateInterface_1: bReadyToOperate
36285	L_MckStateInterface_1: bOperationState1
36286	L_MckStateInterface_1: bOperationState2
36287	L_MckStateInterface_1: bOperationState3
36288	L_MckStateInterface_1: bOperationState4
36289	L_MckStateInterface_1: bOperationState5
36290	L_MckStateInterface_1: bOperationState6
36291	L_MckStateInterface_1: bOperationState7
36292	L_MckStateInterface_1: bOperationState8
36293	L_DigitalLogic_1: bOut2
36294	L_DigitalLogic_1: bOut3
36295	L_DigitalLogic_2: bOut2
36296	L_DigitalLogic_2: bOut3
36297	L_DigitalLogic_3: bOut2
36298	L_DigitalLogic_3: bOut3
42000	LA_NCtrl_In: bCInh
42001	LA_NCtrl_In: bFailReset
42002	LA_NCtrl_In: bSetQuickstop
42003	LA_NCtrl_In: bSetDCBrake
42004	LA_NCtrl_In: bRFG_Stop
42005	LA_NCtrl_In: bRFG_0
42007	LA_NCtrl_In: bSetSpeedCcw
42008	LA_NCtrl_In: bJogSpeed1
42009	LA_NCtrl_In: bJogSpeed2
42010	LA_NCtrl_In: bJogSpeed4
42011	LA_NCtrl_In: bJogSpeed8
42012	LA_NCtrl_In: bJogRamp1
42013	LA_NCtrl_In: bJogRamp2
42014	LA_NCtrl_In: bJogRamp4
42015	LA_NCtrl_In: bJogRamp8
42017	LA_NCtrl_In: bMPotInAct
42018	LA_NCtrl_In: bMPotUp
42019	LA_NCtrl_In: bMPotDown
42020	LA_NCtrl_In: bMBrakeRelease
42021	LA_NCtrl_In: bGPFree1
42022	LA_NCtrl_In: bGPFree2
42023	LA_NCtrl_In: bGPAnalogSwitchSet
42024	LA_NCtrl_In: bGPDigitalDelayIn
42025	LA_NCtrl_In: bGPLogicIn1
42026	LA_NCtrl_In: bGPLogicIn2
42027	LA_NCtrl_In: bGPLogicIn3
42028	LA_NCtrl_In: bGPDFlipFlop_InD
42029	LA_NCtrl_In: bGPDFlipFlop_InClk
42030	LA_NCtrl_In: bGPDFlipFlop_InClr
42031	LA_NCtrl_In: bMPotEnable
42032	LA_NCtrl_In: bPIDEnableInfluenceRamp
42033	LA_NCtrl_In: bPIDIOff
42034	LA_NCtrl_In: bRLQCw
42035	LA_NCtrl_In: bRLQCcw
42041	LA_NCtrl_In: bFreeIn1

Selection list - digital signals	
42042	LA_NCtrl_In: bFreeIn2
42043	LA_NCtrl_In: bFreeIn3
42044	LA_NCtrl_In: bFreeIn4
42045	LA_NCtrl_In: bFreeIn5
42046	LA_NCtrl_In: bFreeIn6
42047	LA_NCtrl_In: bFreeIn7
42048	LA_NCtrl_In: bFreeIn8
42100	LA_TabPos_In: bCInh
42101	LA_TabPos_In: bFailReset
42102	LA_TabPos_In: bSetQuickstop
42103	LA_TabPos_In: bSetSpeedCcw
42104	LA_TabPos_In: bJogSpeed1
42105	LA_TabPos_In: bJogSpeed2
42106	LA_TabPos_In: bMPotEnable
42107	LA_TabPos_In: bMPotUp
42108	LA_TabPos_In: bMPotDown
42109	LA_TabPos_In: bMBrakeRelease
42110	LA_TabPos_In: bPosCtrlOn
42111	LA_TabPos_In: bLimitSwitchPos
42112	LA_TabPos_In: bLimitSwitchNeg
42113	LA_TabPos_In: bReleaseLimitSwitch
42114	LA_TabPos_In: bManJogPos
42115	LA_TabPos_In: bManJogNeg
42116	LA_TabPos_In: bManEnable2ndSpeed
42117	LA_TabPos_In: bEnableSpeedOverride
42118	LA_TabPos_In: bEnableAccOverride
42119	LA_TabPos_In: bHomeStartStop
42120	LA_TabPos_In: bHomeSetPosition
42121	LA_TabPos_In: bHomeResetPosition
42122	LA_TabPos_In: bHomeMark
42123	LA_TabPos_In: bPosSetProfilePosition
42124	LA_TabPos_In: bPosSetActualPosition
42125	LA_TabPos_In: bPosExecute
42126	LA_TabPos_In: bPosFinishTarget
42127	LA_TabPos_In: bPosDisableFollowProfile
42128	LA_TabPos_In: bPosStop
42129	LA_TabPos_In: bGPAnalogSwitchSet
42130	LA_TabPos_In: bGPDigitalDelayIn
42131	LA_TabPos_In: bGPLogicIn1
42132	LA_TabPos_In: bGPLogicIn2
42133	LA_TabPos_In: bGPLogicIn3
42134	LA_TabPos_In: bGPDFlipFlop_InD
42135	LA_TabPos_In: bGPDFlipFlop_InClk
42136	LA_TabPos_In: bGPDFlipFlop_InClr
42137	LA_TabPos_In: bGPCounter1ClkUp
42138	LA_TabPos_In: bGPCounter1ClkDown
42139	LA_TabPos_In: bGPCounter1Load
42140	LA_TabPos_In: bMckOperationMode_1
42141	LA_TabPos_In: bMckOperationMode_2
42142	LA_TabPos_In: bMckOperationMode_4
42143	LA_TabPos_In: bMckOperationMode_8
42144	LA_TabPos_In: bPosProfileNo_1

Selection list - digital signals	
42145	LA_TabPos_In: bPosProfileNo_2
42146	LA_TabPos_In: bPosProfileNo_4
42147	LA_TabPos_In: bPosProfileNo_8
42148	LA_TabPos_In: bFreeIn1
42149	LA_TabPos_In: bFreeIn2
42150	LA_TabPos_In: bFreeIn3
42151	LA_TabPos_In: bFreeIn4
42152	LA_TabPos_In: bFreeIn5
42153	LA_TabPos_In: bFreeIn6
42154	LA_TabPos_In: bFreeIn7
42155	LA_TabPos_In: bFreeIn8
42200	LA_SwitchPos_In: bCInh
42201	LA_SwitchPos_In: bFailReset
42202	LA_SwitchPos_In: bSetQuickstop
42203	LA_SwitchPos_In: bSetDCBrake
42204	LA_SwitchPos_In: bRFG_Stop
42205	LA_SwitchPos_In: bSetSpeedCcw
42206	LA_SwitchPos_In: bRLQCw
42207	LA_SwitchPos_In: bRLQCcw
42208	LA_SwitchPos_In: bJogCtrlInputSel1
42209	LA_SwitchPos_In: bJogCtrlInputSel2
42210	LA_SwitchPos_In: bJogCtrlRfgIn
42211	LA_SwitchPos_In: bJogCtrlJog1
42212	LA_SwitchPos_In: bJogCtrlJog2
42213	LA_SwitchPos_In: bJogCtrlSlowDown1
42214	LA_SwitchPos_In: bJogCtrlStop1
42215	LA_SwitchPos_In: bJogCtrlSlowDown2
42216	LA_SwitchPos_In: bJogCtrlStop2
42217	LA_SwitchPos_In: bJogCtrlSlowDown3
42218	LA_SwitchPos_In: bJogCtrlStop3
42219	LA_SwitchPos_In: bJogSpeed4
42220	LA_SwitchPos_In: bJogSpeed8
42221	LA_SwitchPos_In: bJogRamp1
42222	LA_SwitchPos_In: bJogRamp2
42223	LA_SwitchPos_In: bJogRamp4
42224	LA_SwitchPos_In: bJogRamp8
42225	LA_SwitchPos_In: bMBrakeRelease
42226	LA_SwitchPos_In: bGPAnalogSwitchSet
42227	LA_SwitchPos_In: bGPDigitalDelayIn
42228	LA_SwitchPos_In: bGPLogicIn1
42229	LA_SwitchPos_In: bGPLogicIn2
42230	LA_SwitchPos_In: bGPLogicIn3
42231	LA_SwitchPos_In: bGPDFlipFlop_InD
42232	LA_SwitchPos_In: bGPDFlipFlop_InClk
42233	LA_SwitchPos_In: bGPDFlipFlop_InClr
42239	LA_SwitchPos_In: bFreeIn1
42240	LA_SwitchPos_In: bFreeIn2
42241	LA_SwitchPos_In: bFreeIn3
42242	LA_SwitchPos_In: bFreeIn4
42243	LA_SwitchPos_In: bFreeIn5
42244	LA_SwitchPos_In: bFreeIn6
42245	LA_SwitchPos_In: bFreeIn7

Selection list - digital signals	
42246	LA_SwitchPos_In: bFreeIn8
42247	LS_MotorInterface: bBrakeChopperActive

17.6 Selection list - angle signals

This selection list is relevant for the following configuration parameters:

Parameters	
C00622	System connection list: Angle
C00712	LA_TabPos: phi connection list

Selection list - angle signals	
0	Not connected
1005	LA_NCtrl: dnFreeOut1_p
1006	LA_NCtrl: dnFreeOut2_p
1100	LA_TabPos: dnTargetPos_p
1101	LA_TabPos: dnSetPos_p
1102	LA_TabPos: dnFreeOut1_p
1103	LA_TabPos: dnFreeOut2_p
1104	LA_TabPos: dnPosAct_p
1105	LA_TabPos: dnDeltaPosAct_p
1205	LA_SwitchPos: dnFreeOut1_p
1206	LA_SwitchPos: dnFreeOut2_p
16000	LP_CanIn1: dnIn34_p
16001	LP_CanIn2: dnIn34_p
16002	LP_CanIn3: dnIn34_p
16003	LP_MciIn: dnIn34_p
16006	LP_CanIn1: dnIn12_p
16007	LP_CanIn2: dnIn12_p
16008	LP_CanIn3: dnIn12_p
16009	LP_MciIn: dnIn56_p
16010	LP_MciIn: dnIn78_p
16011	LP_CanIn4: dnIn12_p
16012	LP_CanIn4: dnIn34_p
17000	LS_MultiEncoder: dnPosition_p
17001	LS_Resolver: dnPosition_p
17010	LS_BusEncoder: dnPosition_p
17020	LS_DigitalInput: dnPosIn12_p
17101	LS_TouchProbe: dnTPDigIn1_Position_p
17102	LS_TouchProbe: dnTPDigIn2_Position_p
17103	LS_TouchProbe: dnTPDigIn3_Position_p
17104	LS_TouchProbe: dnTPDigIn4_Position_p
17105	LS_TouchProbe: dnTPDigIn5_Position_p
17106	LS_TouchProbe: dnTPDigIn6_Position_p
17107	LS_TouchProbe: dnTPDigIn7_Position_p
17108	LS_TouchProbe: dnTPEncoder_Position_p
17109	LS_TouchProbe: dnTPResolver_Position_p
17120	LS_RetainData: dnOut1
17121	LS_RetainData: dnOut2
17122	LS_RetainData: dnOut3
17123	LS_RetainData: dnOut4
17130	LS_AxisBusIn: dnLine12
17131	LS_AxisBusIn: dnCas12
17132	LS_AxisBusIn: dnCas34
17133	LS_AxisBusAux: dnAuxIn12

Selection list - angle signals	
17134	LS_AxisBusAux: dnAuxIn34
20000	LS_ParFree_p: dnC474_1_p
20001	LS_ParFree_p: dnC474_2_p
20002	LS_ParFree_p: dnC474_3_p
20003	LS_ParFree_p: dnC474_4_p
20004	LS_ParFree_p: dnC474_5_p
20005	LS_ParFree_p: dnC474_6_p
20006	LS_ParFree_p: dnC474_7_p
20007	LS_ParFree_p: dnC474_8_p
20008	LS_ParFree32: dnC479_1
20009	LS_ParFree32: dnC479_2
20010	LS_ParFree32: dnC479_3
20011	LS_ParFree32: dnC479_4
20012	LS_ParFree32: dnC479_5
20013	LS_ParFree32: dnC479_6
20014	LS_ParFree32: dnC479_7
20015	LS_ParFree32: dnC479_8
20016	LS_ParFreeUnit_1: dnC475_1
20017	LS_ParFreeUnit_1: dnC475_2
20018	LS_ParFreeUnit_1: dnC475_3
20019	LS_ParFreeUnit_1: dnC475_4
20020	LS_ParFreeUnit_1: dnC475_5
20021	LS_ParFreeUnit_1: dnC475_6
20022	LS_ParFreeUnit_1: dnC475_7
20023	LS_ParFreeUnit_1: dnC475_8
20024	LS_ParFreeUnit_1: dnC475_9
20025	LS_ParFreeUnit_1: dnC475_10
20026	LS_ParFreeUnit_1: dnC475_11
20027	LS_ParFreeUnit_1: dnC475_12
20028	LS_ParFreeUnit_1: dnC475_13
20029	LS_ParFreeUnit_1: dnC475_14
20030	LS_ParFreeUnit_1: dnC475_15
20031	LS_ParFreeUnit_1: dnC475_16
20032	LS_ParFreeUnit_2: dnC475_17
20033	LS_ParFreeUnit_2: dnC475_18
20034	LS_ParFreeUnit_2: dnC475_19
20035	LS_ParFreeUnit_2: dnC475_20
20036	LS_ParFreeUnit_2: dnC475_21
20037	LS_ParFreeUnit_2: dnC475_22
20038	LS_ParFreeUnit_2: dnC475_23
20039	LS_ParFreeUnit_2: dnC475_24
20040	LS_ParFreeUnit_2: dnC475_25
20041	LS_ParFreeUnit_2: dnC475_26
20042	LS_ParFreeUnit_2: dnC475_27
20043	LS_ParFreeUnit_2: dnC475_28
20044	LS_ParFreeUnit_2: dnC475_29
20045	LS_ParFreeUnit_2: dnC475_30
20046	LS_ParFreeUnit_2: dnC475_31
20047	LS_ParFreeUnit_2: dnC475_32
32000	LS_MotorInterface: dnMotorPosAct_p

Selection list - angle signals	
32001	LS_MotorInterface: dnMotorDeltaPosAct_p
32200	LS_MotionControlKernel: dnPosTarget_p
32201	LS_MotionControlKernel: dnPosSetValue_p_
32202	LS_MotionControlKernel: dnDeltaPos_p_
32203	LS_MotionControlKernel: dnMotorRefOffset_p_
32204	LS_MotionControlKernel: dnPosSet_p
32205	LS_MotionControlKernel: dnPosSetRelative_p
32206	LS_MotionControlKernel: dnPosCycle_p
36040	L_ArithmetikPhi_1: dnOut_p
36046	L_GainOffsetPhiP_1: dnOut_p
36047	L_GainOffsetPhiP_2: dnOut_p
36050	L_LimitPhi_1: dnOut_p
36051	L_LimitPhi_2: dnOut_p
36052	L_LimitPhi_3: dnOut_p
36060	L_OffsetGainPhiP_1: dnOut_p
36061	L_OffsetGainPhiP_2: dnOut_p
36066	L_PhaseIntK_1: dnOut_p
36067	L_PhaseIntK_2: dnOut_p
36070	L_ArithmetikPhi_2: dnOut_p
36071	L_ArithmetikPhi_3: dnOut_p
36072	L_SQrt_1: dnOut_p
36073	L_Mux_1: dnOut_p
36074	L_ConvWordsToDInt_1: dnOut_p
36075	L_ConvWordsToDInt_2: dnOut_p
36076	L_ConvWordsToDInt_3: dnOut_p
36077	L_ConvUnitsToIncr_1: dnOut_p
36078	L_ConvUnitsToIncr_2: dnOut_p
36079	L_ConvUnitsToIncr_3: dnOut_p
36080	L_Interpolator_1: dnPhiOut_p
36081	L_MckCtrlInterface_1: dnPosSetOut_p
36082	L_PhaseDiff_1: dnOut_p
36083	L_PhaseDiff_2: dnOut_p
36086	L_MckStateInterface_1: dnPosOut_p
36087	L_Odometer_1: dnPosOut_1_p
36088	L_Odometer_1: dnPosOut_2_p
36089	L_Odometer_1: dnPosOut_3_p
36090	L_Odometer_1: dnPosOut_4_p
36091	L_Odometer_1: dnPosOut_5_p
36092	L_Odometer_1: dnPosOut_6_p
36093	L_Odometer_1: dnPosOut_7_p
36094	L_Odometer_1: dnPosOut_8_p
36095	L_Odometer_1: dnDeltaPos_12_p
36096	L_Odometer_1: dnDeltaPos_23_p
36097	L_Odometer_1: dnDeltaPos_34_p
36098	L_Odometer_1: dnDeltaPos_45_p
36099	L_Odometer_1: dnDeltaPos_56_p
36100	L_Odometer_1: dnDeltaPos_67_p
36101	L_Odometer_1: dnDeltaPos_78_p
36102	L_Odometer_1: dnDeltaPos_18_p
36103	L_CalcDiameter_1: dnPos_p
36104	L_PhiIntegrator_1: dnOut32_p
36105	L_PosCtrlLin_1: dnPosOut_p

Selection list - angle signals	
36106	L_PosCtrlLin_2: dnPosOut_p
36107	L_DFSET_1: dnPosDiffOut_p
36108	L_DFSET_1: dnPosSetOut_p
36109	L_GearComp_1: dnOut_p
36110	L_ConvAP_1: dnOut_p
36111	L_ConvAP_2: dnOut_p
36112	L_ConvAP_3: dnOut_p
36113	L_ConvPP_1: dnOut_p
36114	L_ConvPP_2: dnOut_p
36115	L_ConvPP_3: dnOut_p
36116	L_SignalSwitch32_1: dnOut
36117	L_SignalSwitch32_2: dnOut
36118	L_SignalSwitch32_3: dnOut
36119	L_CalcDiameter_1: dwOutDiameter
36120	L_ArithmetikPhi_4: dnOut_p
36121	L_ArithmetikPhi_5: dnOut_p
36122	L_ArithmetikPhi_6: dnOut_p
36123	L_PhiDiv_1: dnOut_p
36124	L_PhiAdd_1: dnOut_p
36125	L_PhiAdd_1: dnOut2_p
36126	L_DFSET_1: dnDeltaPos_p
42005	LA_NCtrl_In: dnFreeIn1_p
42006	LA_NCtrl_In: dnFreeIn2_p
42100	LA_TabPos_In: dnPosProfilePosition_p
42101	LA_TabPos_In: dnFreeIn1_p
42102	LA_TabPos_In: dnFreeIn2_p
42205	LA_SwitchPos_In: dnFreeIn1_p

17.7 Table of attributes

The table of attributes contains information that is required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Type		Parameter type	Selection list	Value from selection list
			Bit coded	Bit coded value
			Linear value	Value with setting range
			String	String
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes without sign
			VISIBLE_STRING [xx]	ASCII string (with character length xx)
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ No decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions 10000 ≡ 4 decimal positions
	CINH	Writing is only possible if the controller is inhibited	CINH	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00001	Keypad UserLevel	Selection list	24574	5FFE	A	1	UNSIGNED_8	1	
C00002	Device commands	Selection list	24573	5FFD	A	34	UNSIGNED_8	1	
C00003	Status of the last device command	Selection list	24572	5FFC	E	1	UNSIGNED_8	1	
C00005	Application	Selection list	24570	5FFA	E	1	UNSIGNED_16	1	
C00006	Motor control	Selection list	24569	5FF9	E	1	UNSIGNED_8	1	
C00007	Control mode	Selection list	24568	5FF8	E	1	UNSIGNED_16	1	
C00008	Original application|control source	Selection list	24567	5FF7	E	1	UNSIGNED_16	1	
C00010	AIN1: Characteristic	Linear value	24565	5FF5	A	8	INTEGER_16	100	
C00011	Appl.: Reference speed	Linear value	24564	5FF4	E	1	UNSIGNED_16	1	
C00012	Accel. time - main setpoint	Linear value	24563	5FF3	E	1	UNSIGNED_32	1000	
C00013	Decel. time - main setpoint	Linear value	24562	5FF2	E	1	UNSIGNED_32	1000	
C00015	VFC: V/f base frequency	Linear value	24560	5FF0	E	1	UNSIGNED_16	10	
C00016	VFC: Vmin boost	Linear value	24559	5FEF	E	1	UNSIGNED_16	100	
C00018	Switching frequency	Selection list	24557	5FED	E	1	UNSIGNED_8	1	
C00019	Auto-DCB: Threshold	Linear value	24556	5FEC	E	1	UNSIGNED_16	1	
C00020	AIN2: Characteristic	Linear value	24555	5FEB	A	8	INTEGER_16	100	
C00021	Slip comp.	Linear value	24554	5FEA	E	1	INTEGER_16	100	
C00022	Imax in motor mode	Linear value	24553	5FE9	E	1	UNSIGNED_16	100	
C00023	Imax in generator mode	Linear value	24552	5FE8	E	1	INTEGER_16	100	
C00024	LS_DriveInterface: bNActCompare	Linear value	24551	5FE7	E	1	INTEGER_16	100	
C00025	LS_DriveInterface: bNActHysterese	Linear value	24550	5FE6	A	1	INTEGER_16	100	
C00026	AINx: Offset	Linear value	24549	5FE5	A	2	INTEGER_16	100	
C00027	AINx: Gain	Linear value	24548	5FE4	A	2	INTEGER_32	10000	
C00028	AINx: Input voltage	Linear value	24547	5FE3	A	2	INTEGER_16	100	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00029	AINx: Input current	Linear value	24546	5FE2	A	2	INTEGER_16	100	
C00030	LS_DFOut: Const.	Linear value	24545	5FE1	A	1	INTEGER_16	1	
C00033	AINx: Output value	Linear value	24542	5FDE	A	2	INTEGER_16	100	
C00034	AINx: Configuration	Selection list	24541	5FDD	A	2	UNSIGNED_8	1	
C00036	DC braking: Current	Linear value	24539	5FDB	E	1	INTEGER_16	100	
C00039	Fixed setpoint x (L_NSet_1 n-Fix)	Linear value	24536	5FD8	A	15	INTEGER_16	100	
C00050	MCTRL: Speed setpoint	Linear value	24525	5FCD	E	1	INTEGER_32	1	
C00051	MCTRL: Actual speed value	Linear value	24524	5FCC	E	1	INTEGER_32	1	
C00052	Motor voltage	Linear value	24523	5FCB	E	1	UNSIGNED_16	1	
C00053	DC-bus voltage	Linear value	24522	5FCA	E	1	UNSIGNED_16	1	
C00054	Motor current	Linear value	24521	5FC9	E	1	UNSIGNED_16	100	
C00055	Actual values	Linear value	24520	5FC8	A	4	INTEGER_16	1	
C00056	Torque	Linear value	24519	5FC7	A	2	INTEGER_32	100	
C00057	Maximum torque	Linear value	24518	5FC6	E	1	UNSIGNED_32	100	
C00058	Output frequency	Linear value	24517	5FC5	E	1	INTEGER_32	100	
C00059	Appl.: Reference frequency C11	Linear value	24516	5FC4	E	1	UNSIGNED_32	100	
C00060	Motor rotor position	Linear value	24515	5FC3	E	1	UNSIGNED_16	1	
C00061	Heatsink temperature	Linear value	24514	5FC2	E	1	INTEGER_16	1	
C00062	Interior temperature	Linear value	24513	5FC1	A	1	INTEGER_16	1	
C00063	Motor temperature	Linear value	24512	5FC0	A	3	INTEGER_16	1	
C00064	Device utilisation (Ixt)	Linear value	24511	5FBF	A	3	INTEGER_16	100	
C00065	Supply voltage 24V	Linear value	24510	5FBE	E	1	INTEGER_16	10	
C00066	Thermal motor load (I²xt)	Linear value	24509	5FBD	E	1	INTEGER_16	100	
C00070	Vp speed controller	Linear value	24505	5FB9	A	3	UNSIGNED_16	100	
C00071	Ti speed controller	Linear value	24504	5FB8	A	3	UNSIGNED_16	10	
C00072	SC: Tdn speed controller	Linear value	24503	5FB7	E	1	UNSIGNED_16	100	
C00073	Imax/M controller gain	Linear value	24502	5FB6	A	2	UNSIGNED_16	100	
C00074	Reset time Imax/M controller	Linear value	24501	5FB5	A	2	UNSIGNED_16	1	
C00075	Vp current controller	Linear value	24500	5FB4	E	1	UNSIGNED_16	100	
C00076	Ti current controller	Linear value	24499	5FB3	E	1	UNSIGNED_16	100	
C00077	SC: Vp field controller	Linear value	24498	5FB2	E	1	UNSIGNED_16	100	
C00078	SC: Tn field controller	Linear value	24497	5FB1	E	1	UNSIGNED_16	10	
C00079	SC: Settings	Selection list	24496	5FB0	A	4	UNSIGNED_8	1	
C00080	Override point of field weakening	Linear value	24495	5FAF	E	1	INTEGER_16	1	
C00081	Rated motor power	Linear value	24494	5FAE	E	1	UNSIGNED_16	100	
C00082	Motor rotor resistance	Linear value	24493	5FAD	E	1	UNSIGNED_32	1	
C00083	Motor rotor time constant	Linear value	24492	5FAC	E	1	UNSIGNED_16	1	
C00084	Motor stator resistance	Linear value	24491	5FAB	E	1	UNSIGNED_32	1	
C00085	Motor stator leakage inductance	Linear value	24490	5FAA	E	1	UNSIGNED_16	100	
C00087	Rated motor speed	Linear value	24488	5FA8	E	1	UNSIGNED_16	1	
C00088	Rated motor current	Linear value	24487	5FA7	E	1	UNSIGNED_16	100	
C00089	Rated motor frequency	Linear value	24486	5FA6	E	1	UNSIGNED_16	1	
C00090	Rated motor voltage	Linear value	24485	5FA5	E	1	UNSIGNED_16	1	
C00091	Motor cosine phi	Linear value	24484	5FA4	E	1	UNSIGNED_8	100	
C00092	Motor magnetising inductance	Linear value	24483	5FA3	E	1	UNSIGNED_16	10	
C00093	Power section ID	Linear value	24482	5FA2	E	1	UNSIGNED_16	1	
C00095	Motor magnetising current	Linear value	24480	5FA0	E	1	UNSIGNED_16	100	
C00097	Rated motor torque	Linear value	24478	5F9E	E	1	UNSIGNED_32	100	
C00098	Rated device current	Linear value	24477	5F9D	E	1	UNSIGNED_16	10	
C00099	Firmware version	String	24476	5F9C	E	1	VISIBLE_STRING [12]		
C00100	Firmware version	Linear value	24475	5F9B	A	4	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00101	Add. acceleration time x	Linear value	24474	5F9A	A	15	UNSIGNED_32	1000	
C00103	Add. acceleration time x	Linear value	24472	5F98	A	15	UNSIGNED_32	1000	
C00104	Quick stop setting	Bit coded	24471	5F97	A	1	UNSIGNED_16	1	
C00105	Decel. time - quick stop	Linear value	24470	5F96	E	1	UNSIGNED_32	1000	
C00106	Auto-DCB: Hold time	Linear value	24469	5F95	E	1	UNSIGNED_32	1000	
C00107	DC braking: Hold time	Linear value	24468	5F94	E	1	UNSIGNED_32	1000	
C00114	DigInX: Inversion	Bit coded	24461	5F8D	E	1	UNSIGNED_16	1	
C00115	DI 1/2 & 6/7: Function	Selection list	24460	5F8C	A	2	UNSIGNED_8	1	
C00117	Status of brake output BD	Selection list	24458	5F8A	E	1	UNSIGNED_8	1	
C00118	DigOutX: Inversion	Bit coded	24457	5F89	E	1	UNSIGNED_8	1	
C00120	Setting of motor overload (I²xt)	Linear value	24455	5F87	E	1	INTEGER_16	100	
C00121	Motor temp. warning threshold	Linear value	24454	5F86	A	2	UNSIGNED_16	1	
C00122	Starting value — Overload	Linear value	24453	5F85	A	2	UNSIGNED_16	100	
C00123	Device utilisat. threshold (Ixt)	Linear value	24452	5F84	E	1	INTEGER_16	100	
C00124	Current monitoring: Breaking current	Linear value	24451	5F83	A	1	UNSIGNED_16	100	
C00129	Brake resistance value	Linear value	24446	5F7E	E	1	UNSIGNED_16	10	
C00130	Rated brake resistor power	Linear value	24445	5F7D	E	1	UNSIGNED_16	1	
C00131	Rated heat amount of brake resist.	Linear value	24444	5F7C	E	1	UNSIGNED_16	10	
C00133	Brake resistor utilisation	Linear value	24442	5F7A	E	1	UNSIGNED_16	1	
C00134	L_NSet_1: Ramp smoothing	Selection list	24441	5F79	E	1	UNSIGNED_8	1	
C00136	Communication control words	Bit coded	24439	5F77	A	2	UNSIGNED_16	1	
C00137	Device status	Selection list	24438	5F76	E	1	UNSIGNED_16	1	
C00138	Internal control signals	Bit coded	24437	5F75	A	3	UNSIGNED_16	1	
C00142	Auto-start option	Bit coded	24433	5F71	E	1	UNSIGNED_8	1	
C00144	Thermal switching frequency reduction	Selection list	24431	5F6F	E	1	UNSIGNED_8	1	
C00148	LS_DriveInterface: Error message config.	Bit coded	24427	5F6B	E	1	UNSIGNED_16	1	
C00150	Status word	Bit coded	24425	5F69	E	1	UNSIGNED_16	1	
C00155	Extended status word	Bit coded	24420	5F64	E	1	UNSIGNED_16	1	
C00158	Cause of controller inhibit	Bit coded	24417	5F61	E	1	UNSIGNED_16	1	
C00159	Cause of quick stop QSP	Bit coded	24416	5F60	E	1	UNSIGNED_16	1	
C00160	Status determining error (16-bit)	Linear value	24415	5F5F	A	1	UNSIGNED_16	1	
C00161	LS_SetError_x: Error number	Linear value	24414	5F5E	A	8	UNSIGNED_16	1	
C00162	Error number masked	Linear value	24413	5F5D	A	1	UNSIGNED_32	1	
C00163	Logbook - binary elements	Selection list	24412	5F5C	A	2	UNSIGNED_16	1	
C00164	Logbook - analog elements	Selection list	24411	5F5B	A	1	UNSIGNED_16	1	
C00165	Error information	String	24410	5F5A	A	2	VISIBLE_STRING [14]		
C00166	Error information text	String	24409	5F59	A	6	VISIBLE_STRING [31]		
C00168	Status determining error	Linear value	24407	5F57	E	1	UNSIGNED_32	1	
C00169	Logbook setting	Bit coded	24406	5F56	E	1	UNSIGNED_16	1	
C00170	Current error	Linear value	24405	5F55	E	1	UNSIGNED_32	1	
C00173	Mains voltage	Selection list	24402	5F52	E	1	UNSIGNED_8	1	CINH
C00174	Reduced brake chopper threshold	Linear value	24401	5F51	E	1	UNSIGNED_8	1	
C00175	Brake energy management	Selection list	24400	5F50	E	1	UNSIGNED_8	1	CINH
C00177	Switching cycles	Linear value	24398	5F4E	A	9	UNSIGNED_32	1	
C00178	Elapsed-hour meter	Linear value	24397	5F4D	E	1	UNSIGNED_32	1	
C00179	Power-on time meter	Linear value	24396	5F4C	E	1	UNSIGNED_32	1	
C00180	Running time	Linear value	24395	5F4B	A	3	UNSIGNED_32	1	
C00181	Time settings	Linear value	24394	5F4A	A	1	UNSIGNED_16	1	
C00182	L_NSet_1: S-ramp time PT1	Linear value	24393	5F49	E	1	INTEGER_16	100	
C00184	AutoFailReset repetition time	Linear value	24391	5F47	E	1	UNSIGNED_16	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00185	AutoFailReset residual runtime	Linear value	24390	5F46	E	1	UNSIGNED_16	1	
C00186	Max. number of AutoFailReset processes	Linear value	24389	5F45	E	1	UNSIGNED_8	1	
C00187	Current AutoFailReset processes	Linear value	24388	5F44	E	1	UNSIGNED_8	1	
C00188	AutoFailReset configuration	Selection list	24387	5F43	E	1	UNSIGNED_8	1	
C00189	Resp. to too frequent AutoFailReset	Selection list	24386	5F42	E	1	UNSIGNED_8	1	
C00190	L_NSet_1: Setpoint arithmetic	Selection list	24385	5F41	E	1	UNSIGNED_8	1	
C00191	Logbook access index User	Linear value	24384	5F40	A	1	UNSIGNED_8	1	
C00192	Logbook data User	String	24383	5F3F	A	1	OCTET_STRING [32]		
C00193	Logbook element User	Linear value	24382	5F3E	A	6	UNSIGNED_32	1	
C00199	Description data	String	24376	5F38	A	5	VISIBLE_STRING [32]		
C00200	Firmware product type	String	24375	5F37	E	1	VISIBLE_STRING [19]		
C00201	Firmware	String	24374	5F36	A	9	VISIBLE_STRING [22]		
C00203	Product type code	String	24372	5F34	A	9	VISIBLE_STRING [24]		
C00204	Serial number	String	24371	5F33	A	9	VISIBLE_STRING [24]		
C00219	Identification	Linear value	24356	5F24	A	4	UNSIGNED_32	1	
C00220	L_NSet_1: Acceleration time - add. setpoint	Linear value	24355	5F23	E	1	UNSIGNED_32	1000	
C00221	L_NSet_1: Deceleration time - add. setpoint	Linear value	24354	5F22	E	1	UNSIGNED_32	1000	
C00222	L_PCTRL_1: Vp	Linear value	24353	5F21	E	1	INTEGER_16	10	
C00223	L_PCTRL_1: Tn	Linear value	24352	5F20	E	1	UNSIGNED_16	1	
C00224	L_PCTRL_1: Kd	Linear value	24351	5F1F	E	1	UNSIGNED_16	10	
C00225	L_PCTRL_1: MaxLimit	Linear value	24350	5F1E	E	1	INTEGER_16	100	
C00226	L_PCTRL_1: MinLimit	Linear value	24349	5F1D	E	1	INTEGER_16	100	
C00227	L_PCTRL_1: Acceleration time	Linear value	24348	5F1C	E	1	UNSIGNED_32	1000	
C00228	L_PCTRL_1: Deceleration time	Linear value	24347	5F1B	E	1	UNSIGNED_32	1000	
C00231	L_PCTRL_1: Operating range	Linear value	24344	5F18	A	4	INTEGER_16	100	
C00233	L_PCTRL_1: Root function	Selection list	24342	5F16	E	1	UNSIGNED_8	1	
C00234	Oscillation damping influence	Linear value	24341	5F15	E	1	UNSIGNED_16	100	
C00235	Oscillation damping filter time	Linear value	24340	5F14	E	1	UNSIGNED_8	1	
C00236	Oscillation damping field weakening	Linear value	24339	5F13	E	1	UNSIGNED_8	1	
C00241	L_NSet_1: Hyst. NSet reached	Linear value	24334	5F0E	E	1	INTEGER_16	100	
C00242	L_PCTRL_1: Operating mode	Selection list	24333	5F0D	E	1	UNSIGNED_8	1	
C00243	L_PCTRL_1: Accel. time influence	Linear value	24332	5F0C	E	1	UNSIGNED_32	1000	
C00244	L_PCTRL_1: Deceleration time influence	Linear value	24331	5F0B	E	1	UNSIGNED_32	1000	
C00245	L_PCTRL_1: PID output value	Linear value	24330	5F0A	E	1	INTEGER_16	100	
C00246	L_PCTRL_1: nAct_a internal	Linear value	24329	5F09	E	1	INTEGER_16	100	
C00247	L_PCTRL_1: Window setpoint reached	Linear value	24328	5F08	E	1	INTEGER_16	100	
C00249	L_PT1_1: Time constant	Linear value	24326	5F06	E	1	UNSIGNED_16	1	
C00250	L_PT1 2-3: Time constant	Linear value	24325	5F05	A	2	INTEGER_16	1	
C00251	L_DT1_1: Time constant	Linear value	24324	5F04	E	1	INTEGER_16	1	
C00252	L_DT1_1: Gain	Linear value	24323	5F03	E	1	INTEGER_16	100	
C00253	L_DT1_1: Sensitivity	Selection list	24322	5F02	E	1	UNSIGNED_8	1	
C00254	Kp position controller	Linear value	24321	5F01	E	1	UNSIGNED_16	100	
C00270	SC: Freq. current setpoint filter	Linear value	24305	5EF1	E	1	UNSIGNED_16	10	
C00271	SC: Current setpoint filter width	Linear value	24304	5EF0	E	1	UNSIGNED_16	10	
C00272	SC: Current setpoint filter depth	Linear value	24303	5EEF	E	1	UNSIGNED_16	1	
C00273	Motor moment of inertia	Linear value	24302	5EEE	E	1	UNSIGNED_32	100	
C00274	SC: Max. change in acceleration	Linear value	24301	5EED	E	1	UNSIGNED_16	10	
C00275	Setpoint feedforward control filtering	Linear value	24300	5EEC	E	1	UNSIGNED_16	10	
C00276	SC: max. output voltage	Linear value	24299	5EEB	E	1	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00280	SC: Filter time const. DC detection	Linear value	24295	5EE7	E	1	UNSIGNED_16	1	
C00312	System runtimes	Linear value	24263	5EC7	A	1	UNSIGNED_32	1000	
C00321	Main program runtime	Linear value	24254	5EBE	A	2	UNSIGNED_16	1	
C00322	Transmission mode CAN TxPDOs	Linear value	24253	5EBD	A	4	UNSIGNED_8	1	
C00323	Transmission mode CAN Rx PDOs	Linear value	24252	5EBC	A	4	UNSIGNED_8	1	
C00324	CAN transmit blocking time	Linear value	24251	5EBB	A	5	UNSIGNED_16	1	
C00338	L_Arithmetik_1: Function	Selection list	24237	5EAD	E	1	UNSIGNED_8	1	
C00339	L_Arithmetik_2: Function	Selection list	24236	5EAC	E	1	UNSIGNED_8	1	
C00341	CAN management - error configuration	Bit coded	24234	5EAA	E	1	UNSIGNED_16	1	
C00342	CAN decoupling PDOInOut	Bit coded	24233	5EA9	A	2	UNSIGNED_16	1	
C00343	LP_CanIn decoupling value	Linear value	24232	5EA8	A	16	UNSIGNED_16	1	
C00344	LP_CanOut decoupling value	Linear value	24231	5EA7	A	16	UNSIGNED_16	1	
C00345	CAN error status	Selection list	24230	5EA6	E	1	UNSIGNED_8	1	
C00347	CAN status HeartBeat producer	Selection list	24228	5EA4	A	15	UNSIGNED_8	1	
C00349	CAN setting - DIP switch	Bit coded	24226	5EA2	E	1	UNSIGNED_16	1	
C00350	CAN node address	Linear value	24225	5EA1	E	1	UNSIGNED_8	1	
C00351	CAN baud rate	Selection list	24224	5EA0	E	1	UNSIGNED_8	1	
C00352	CAN slave/master	Selection list	24223	5E9F	E	1	UNSIGNED_8	1	
C00353	CAN IN/OUT COBID source	Selection list	24222	5E9E	A	4	UNSIGNED_8	1	
C00354	COBID	Bit coded	24221	5E9D	A	8	UNSIGNED_32	1	
C00355	Active COBID	Linear value	24220	5E9C	A	8	UNSIGNED_16	1	
C00356	CAN time settings	Linear value	24219	5E9B	A	6	UNSIGNED_16	1	
C00357	CAN monitoring times	Linear value	24218	5E9A	A	4	UNSIGNED_16	1	
C00358	CANx_OUT data length	Linear value	24217	5E99	A	4	UNSIGNED_8	1	
C00359	CAN status	Selection list	24216	5E98	E	1	UNSIGNED_8	1	
C00360	CAN telegram counter	Linear value	24215	5E97	A	14	UNSIGNED_16	1	
C00364	CAN MessageError	Selection list	24211	5E93	E	1	UNSIGNED_8	1	
C00366	Number of CAN SDO channels	Selection list	24209	5E91	E	1	UNSIGNED_8	1	
C00367	CAN SYNC Rx identifier	Bit coded	24208	5E90	E	1	UNSIGNED_16	1	
C00368	CAN SYNC Tx identifier	Bit coded	24207	5E8F	E	1	UNSIGNED_16	1	
C00369	CAN sync transmission cycle time	Linear value	24206	5E8E	E	1	UNSIGNED_16	1	
C00370	SyncTxRxTimes	Linear value	24205	5E8D	A	2	INTEGER_16	1	
C00371	CAN ErrorCode	Linear value	24204	5E8C	A	1	UNSIGNED_16	1	
C00372	CAN_Tx_Rx_Error	Linear value	24203	5E8B	A	4	UNSIGNED_8	1	
C00381	CAN Heartbeat producer time	Linear value	24194	5E82	E	1	UNSIGNED_16	1	
C00385	CAN node addr. HeartBeat producer	Linear value	24190	5E7E	A	15	UNSIGNED_8	1	
C00386	CAN HeartBeat-ConsumerTime	Linear value	24189	5E7D	A	15	UNSIGNED_16	1	
C00387	CAN-GatewayAddr	Linear value	24188	5E7C	A	1	UNSIGNED_8	1	
C00400	LS_PulseGenerator	Linear value	24175	5E6F	A	3	UNSIGNED_16	1	
C00401	CANxInOut: Inversion	Bit coded	24174	5E6E	A	8	UNSIGNED_16	1	
C00407	LP_CanIn mapping	Linear value	24168	5E68	A	8	UNSIGNED_32	1	
C00408	LP_CanIn mapping selection	Selection list	24167	5E67	A	4	UNSIGNED_8	1	
C00409	LP_CanIn mapping	Linear value	24166	5E66	A	16	UNSIGNED_16	1	
C00410	L_SignalMonitor_a: Signal sources	Selection list	24165	5E65	A	4	UNSIGNED_16	1	
C00411	L_SignalMonitor_b: Signal sources	Selection list	24164	5E64	A	4	UNSIGNED_16	1	
C00412	L_SignalMonitor_b: Inversion	Bit coded	24163	5E63	E	1	UNSIGNED_8	1	
C00413	L_SignalMonitor_a: Offs./gain	Linear value	24162	5E62	A	8	INTEGER_16	100	
C00417	Deactivate resolver error comp.	Selection list	24158	5E5E	E	1	UNSIGNED_8	1	
C00420	Number of encoder increments	Linear value	24155	5E5B	A	3	UNSIGNED_16	1	
C00421	LS_MultiEncoder: Supply voltage	Linear value	24154	5E5A	E	1	UNSIGNED_16	10	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00422	LS_MultiEncoder: Encoder type	Selection list	24153	5E59	E	1	UNSIGNED_8	1	CINH
C00423	DOx: Delay times	Linear value	24152	5E58	A	10	UNSIGNED_16	1000	
C00424	Pulse form TTL encoder	Selection list	24151	5E57	E	1	UNSIGNED_8	1	
C00425	Encoder scanning time	Selection list	24150	5E56	A	2	UNSIGNED_8	1	CINH
C00426	SSI encoder: Data bits	Linear value	24149	5E55	A	9	UNSIGNED_8	1	CINH
C00427	SSI encoder: Bit rate	Selection list	24148	5E54	E	1	UNSIGNED_8	1	CINH
C00428	SSI encoder: Coding	Selection list	24147	5E53	E	1	UNSIGNED_8	1	CINH
C00430	LS_MultiEncoder: Max. initialization time	Linear value	24145	5E51	A	1	UNSIGNED_16	1	CINH
C00431	LS_MultiEncoder: bFail selection	Bit coded	24144	5E50	A	1	UNSIGNED_16	1	
C00432	LS_Resolver: bFail selection	Bit coded	24143	5E4F	A	1	UNSIGNED_16	1	
C00434	OxU/I: Gain	Linear value	24141	5E4D	A	4	INTEGER_16	100	
C00435	OxU/I: Offset	Linear value	24140	5E4C	A	4	INTEGER_16	100	
C00436	OxU: Voltage	Linear value	24139	5E4B	A	2	INTEGER_16	100	
C00437	OxI: Current	Linear value	24138	5E4A	A	2	INTEGER_32	1000	
C00439	OxU/I: Input value	Linear value	24136	5E48	A	4	INTEGER_16	100	
C00440	LS_AnalogIn1: PT1 time constant	Linear value	24135	5E47	A	2	UNSIGNED_16	1	
C00441	Decoupling AnalogOut	Bit coded	24134	5E46	E	1	UNSIGNED_16	1	
C00442	AOutx: Decoupling value	Linear value	24133	5E45	A	4	INTEGER_16	100	
C00443	DIx: Level	Bit coded	24132	5E44	A	2	UNSIGNED_16	1	
C00444	DOx: Level	Bit coded	24131	5E43	A	2	UNSIGNED_16	1	
C00445	FreqInxx_nOut_v	Linear value	24130	5E42	A	2	INTEGER_16	1	
C00446	FreqInxx_nOut_a	Linear value	24129	5E41	A	2	INTEGER_16	100	
C00447	DigOut decoupling	Bit coded	24128	5E40	E	1	UNSIGNED_16	1	
C00448	DigOut decoupling value	Bit coded	24127	5E3F	E	1	UNSIGNED_16	1	
C00449	FreqInxx_dnOut_p	Linear value	24126	5E3E	A	1	INTEGER_32	1	
C00450	HTL encoder input frequency	Linear value	24125	5E3D	A	2	UNSIGNED_32	1000	
C00451	LS_MultiEncoder: Information bFail	Bit coded	24124	5E3C	A	1	UNSIGNED_16	1	
C00452	LS_Resolver: Information bFail	Bit coded	24123	5E3B	A	1	UNSIGNED_16	1	
C00453	Keypad: Default manual jog	Linear value	24122	5E3A	A	3	UNSIGNED_32	1000	
C00461	Remote: Acceleration/deceleration time	Linear value	24114	5E32	A	1	UNSIGNED_32	1000	
C00462	Remote: Control	Linear value	24113	5E31	A	2	UNSIGNED_16	1	
C00463	Remote: MCK control	Bit coded	24112	5E30	A	1	UNSIGNED_32	1	
C00464	Remote: Monitoring timeout	Linear value	24111	5E2F	A	1	UNSIGNED_16	1	
C00465	Keypad: Time-out welcome screen	Selection list	24110	5E2E	E	1	INTEGER_32	1	
C00466	Keypad: Default parameter	Linear value	24109	5E2D	E	1	INTEGER_32	1	
C00467	Keypad: Default welcome screen	Selection list	24108	5E2C	E	1	INTEGER_32	1	
C00469	Keypad: Fct. STOP key	Selection list	24106	5E2A	E	1	INTEGER_32	1	CINH
C00470	LS_ParFree_b	Selection list	24105	5E29	A	32	UNSIGNED_8	1	
C00471	LS_ParFree	Bit coded	24104	5E28	A	32	UNSIGNED_16	1	
C00472	LS_ParFree_a	Linear value	24103	5E27	A	16	INTEGER_16	100	
C00473	LS_ParFree_v	Linear value	24102	5E26	A	8	INTEGER_16	1	
C00474	LS_ParFree_p	Linear value	24101	5E25	A	8	INTEGER_32	1	
C00475	LS_ParFreeUnit_1_2	Linear value	24100	5E24	A	32	INTEGER_32	10000	
C00476	LS_ParFree_a_2	Linear value	24099	5E23	A	16	INTEGER_16	100	
C00477	LS_ParFree_2	Bit coded	24098	5E22	A	32	UNSIGNED_16	1	
C00478	LS_ParFree_v_2	Linear value	24097	5E21	A	8	INTEGER_16	1	
C00479	LS_ParFree32	Linear value	24096	5E20	A	8	INTEGER_32	1	
C00480	LS_DisFree_b	Bit coded	24095	5E1F	E	1	UNSIGNED_16	1	
C00481	LS_DisFree	Bit coded	24094	5E1E	A	8	UNSIGNED_16	1	
C00482	LS_DisFree_a	Linear value	24093	5E1D	A	8	INTEGER_16	100	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00483	LS_DisFree_p	Linear value	24092	5E1C	A	8	INTEGER_32	1	
C00484	Application units: Offset	Linear value	24091	5E1B	A	4	INTEGER_16	100	
C00485	Application units: Display factor	Linear value	24090	5E1A	A	4	INTEGER_32	10000	
C00486	Application units: Text	String	24089	5E19	A	4	VISIBLE_STRING [7]		
C00487	Application units	Linear value	24088	5E18	A	4	INTEGER_32	100	
C00488	L_JogCtrlEdgeDetect_1	Selection list	24087	5E17	A	6	UNSIGNED_8	1	
C00490	Position encoder selection	Selection list	24085	5E15	E	1	UNSIGNED_8	1	
C00491	Hiperface messages	Linear value	24084	5E14	A	4	UNSIGNED_8	1	
C00492	Hiperface: Detected TypCode	Linear value	24083	5E13	E	1	UNSIGNED_8	1	
C00493	Hiperface: TypCode	Linear value	24082	5E12	E	1	UNSIGNED_8	1	
C00494	Hiperface: Resolutions	Linear value	24081	5E11	A	2	UNSIGNED_32	1	
C00495	Speed sensor selection	Selection list	24080	5E10	E	1	UNSIGNED_8	1	
C00496	Encoder evaluation method DigIn12	Selection list	24079	5E0F	E	1	UNSIGNED_8	1	CINH
C00497	Filter time constant	Linear value	24078	5E0E	A	4	UNSIGNED_16	10	
C00498	Open-circuit monitoring	Selection list	24077	5E0D	A	1	UNSIGNED_8	1	
C00499	Hiperface: Settings	Bit coded	24076	5E0C	A	1	UNSIGNED_16	1	
C00505	Password data	String	24070	5E06	A	3	VISIBLE_STRING [16]		
C00506	PW protection internal config	Bit coded	24069	5E05	A	1	UNSIGNED_16	1	
C00507	Current password protection	Bit coded	24068	5E04	A	1	UNSIGNED_16	1	
C00508	PW protection config X6(DIAG)	Bit coded	24067	5E03	A	1	UNSIGNED_16	1	
C00509	PW protection config X1 (CAN)	Bit coded	24066	5E02	A	1	UNSIGNED_16	1	
C00510	PW protection config. MCI	Bit coded	24065	5E01	A	1	UNSIGNED_16	1	
C00517	User menu	Linear value	24058	5DFA	A	32	INTEGER_32	1000	
C00540	LS_DFOut: Function	Selection list	24035	5DE3	E	1	UNSIGNED_8	1	
C00545	LS_DFOut: Angular offset	Linear value	24030	5DDE	A	1	INTEGER_32	1	
C00560	Fan switching status	Selection list	24015	5DCF	A	2	UNSIGNED_8	1	
C00561	Failure indication	Selection list	24014	5DCE	A	5	UNSIGNED_8	1	
C00563	Current monitoring: Delay time	Linear value	24012	5DCC	A	1	UNSIGNED_32	1000	
C00565	Resp. to mains phase failure	Selection list	24010	5DCA	E	1	UNSIGNED_8	1	
C00566	Resp. to fan failure	Selection list	24009	5DC9	E	1	UNSIGNED_8	1	
C00567	Resp. to speed controller limited	Selection list	24008	5DC8	E	1	UNSIGNED_8	1	
C00569	Resp. to peak current	Selection list	24006	5DC6	A	1	UNSIGNED_8	1	
C00570	Resp. to controller limitations	Selection list	24005	5DC5	A	4	UNSIGNED_8	1	
C00571	Resp. to wrong controller setting	Selection list	24004	5DC4	A	2	UNSIGNED_8	1	
C00572	Brake resistor overload threshold	Linear value	24003	5DC3	E	1	UNSIGNED_8	1	
C00574	Resp. to brake resist. overtemp.	Selection list	24001	5DC1	E	1	UNSIGNED_8	1	
C00576	SC: Optimisation of field feedforward control	Linear value	23999	5DBF	E	1	UNSIGNED_16	1	
C00577	SC: Vp field weakening controller	Linear value	23998	5DBE	E	1	UNSIGNED_16	10000	
C00578	SC: Tn field weakening controller	Linear value	23997	5DBD	E	1	UNSIGNED_16	10	
C00579	Resp. to max. speed/output freq. reached	Selection list	23996	5DBC	E	1	UNSIGNED_8	1	
C00580	Resp. to operating system error	Selection list	23995	5DBB	A	1	UNSIGNED_8	1	
C00581	Resp. to LS_SetError_x	Selection list	23994	5DBA	A	8	UNSIGNED_8	1	
C00582	Resp. to heatsink temp.> shutdown temp. -5°C	Selection list	23993	5DB9	E	1	UNSIGNED_8	1	
C00583	Resp. to motor temperature KTY	Selection list	23992	5DB8	A	6	UNSIGNED_8	1	
C00584	Resp. to current monitoring	Selection list	23991	5DB7	A	1	UNSIGNED_8	1	
C00585	Resp. to motor overtemp. PTC	Selection list	23990	5DB6	E	1	UNSIGNED_8	1	
C00586	Resp. open circuit HTL encoder	Selection list	23989	5DB5	E	1	UNSIGNED_8	1	
C00588	Resp. to max. speed at switching freq.	Selection list	23987	5DB3	E	1	UNSIGNED_8	1	
C00590	Resp. to switch. frequency red.	Selection list	23985	5DB1	E	1	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00591	Reakt. Achsbusfehler	Selection list	23984	5DB0	A	3	UNSIGNED_8	1	
C00592	Resp. to CAN bus connection	Selection list	23983	5DAF	A	5	UNSIGNED_8	1	
C00593	Resp. to CANx_IN monitoring	Selection list	23982	5DAE	A	4	UNSIGNED_8	1	
C00594	Resp. to control word error	Selection list	23981	5DAD	A	2	UNSIGNED_8	1	
C00595	MCK: Resp. to MCK error	Selection list	23980	5DAC	A	17	UNSIGNED_8	1	
C00597	Resp. to motor phase failure	Selection list	23978	5DAA	E	1	UNSIGNED_8	1	
C00598	Resp. to open circuit AINx	Selection list	23977	5DA9	A	2	UNSIGNED_8	1	
C00599	Motor phase failure threshold	Linear value	23976	5DA8	E	1	INTEGER_16	100	
C00600	Resp. to DC bus voltage	Selection list	23975	5DA7	A	1	UNSIGNED_8	1	
C00601	Delayed resp. to fault: DC bus overvoltage	Linear value	23974	5DA6	A	1	UNSIGNED_16	1000	
C00602	Resp. to earth fault	Selection list	23973	5DA5	E	1	UNSIGNED_8	1	
C00603	Resp. to feedback	Selection list	23972	5DA4	A	9	UNSIGNED_8	1	
C00604	Resp. to device overload (Ixt)	Selection list	23971	5DA3	E	1	UNSIGNED_8	1	
C00605	Resp. to feedback	Selection list	23970	5DA2	A	1	UNSIGNED_8	1	
C00606	Resp. to motor overload (I²xt)	Selection list	23969	5DA1	E	1	UNSIGNED_8	1	
C00607	Resp. to max. freq. feedb. DIG12/67	Selection list	23968	5DA0	E	1	UNSIGNED_8	1	
C00608	Resp. to maximum torque	Selection list	23967	5D9F	E	1	UNSIGNED_8	1	
C00609	Resp. to maximum current	Selection list	23966	5D9E	E	1	UNSIGNED_8	1	
C00620	System connection list: 16-bit	Selection list	23955	5D93	A	111	UNSIGNED_16	1	
C00621	System connection list: Bool	Selection list	23954	5D92	A	199	UNSIGNED_16	1	
C00622	System connection list: Angle	Selection list	23953	5D91	A	25	UNSIGNED_16	1	
C00630	L_Limit 1-2: Min/Max	Linear value	23945	5D89	A	4	INTEGER_16	100	
C00631	L_LimitPhi 1-3: Min/Max	Linear value	23944	5D88	A	6	INTEGER_32	1	
C00632	L_NSet_1: Max. skip freq.	Linear value	23943	5D87	A	3	INTEGER_16	100	
C00633	L_NSet_1: Min. skip freq.	Linear value	23942	5D86	A	3	INTEGER_16	100	
C00634	L_NSet_1: wState	Bit coded	23941	5D85	E	1	UNSIGNED_16	1	
C00635	L_NSet_1: nMaxLimit	Linear value	23940	5D84	E	1	INTEGER_16	100	
C00636	L_NSet_1: nMinLimit	Linear value	23939	5D83	E	1	INTEGER_16	100	
C00637	L_NSet_1: Output blocking zones	Linear value	23938	5D82	E	1	INTEGER_16	100	
C00638	L_NSet_1: Output ramp rounding	Linear value	23937	5D81	E	1	INTEGER_16	100	
C00639	L_NSet_1: Output add.value	Linear value	23936	5D80	E	1	INTEGER_16	100	
C00640	L_NSet_1: nNOut_a	Linear value	23935	5D7F	E	1	INTEGER_16	100	
C00643	Resp. to PLI monitoring	Selection list	23932	5D7C	A	1	UNSIGNED_8	1	
C00644	PLI traversing direction	Selection list	23931	5D7B	A	1	UNSIGNED_8	1	
C00645	PLI max. permissible deflection	Linear value	23930	5D7A	A	2	INTEGER_16	10	
C00646	PLI current amplitude	Linear value	23929	5D79	A	2	UNSIGNED_16	1	
C00647	PLI ramp time	Linear value	23928	5D78	A	2	UNSIGNED_16	1	
C00650	L_Arithmetik 3-5: Function	Selection list	23925	5D75	A	3	UNSIGNED_8	1	
C00653	Sensibility - Setpoint feedforward control	Selection list	23922	5D72	A	1	UNSIGNED_8	1	
C00654	Source of diff. setpoint feedforward control	Selection list	23921	5D71	A	1	UNSIGNED_8	1	
C00660	L_FixSet_a_1: Analog values	Linear value	23915	5D6B	A	16	INTEGER_16	100	
C00661	L_FixSet_w_1: Fixed values	Linear value	23914	5D6A	A	16	UNSIGNED_16	1	
C00662	L_FixSet_w_2: Fixed values	Linear value	23913	5D69	A	16	UNSIGNED_16	1	
C00670	L_OffsetGainP_1: Gain	Linear value	23905	5D61	E	1	INTEGER_32	10000	
C00671	L_OffsetGainP_2: Gain	Linear value	23904	5D60	E	1	INTEGER_32	10000	
C00672	L_OffsetGainP_3: Gain	Linear value	23903	5D5F	E	1	INTEGER_32	10000	
C00673	L_OffsetGainPhiP 1-2: Offset	Linear value	23902	5D5E	A	2	INTEGER_32	1	
C00674	L_OffsetGainPhiP 1-2: Gain	Linear value	23901	5D5D	A	2	INTEGER_32	1	
C00677	L_GainOffsetP 1-3: Parameter	Linear value	23898	5D5A	A	6	INTEGER_16	100	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00678	L_GainOffsetPhiP 1-2: Parameter	Linear value	23897	5D59	A	4	INTEGER_32	1	
C00679	L_MulDiv_2: Parameter	Linear value	23896	5D58	A	2	INTEGER_16	1	
C00680	L_Compare_1: Fct.	Selection list	23895	5D57	E	1	UNSIGNED_8	1	
C00681	L_Compare_1: Hysteresis	Linear value	23894	5D56	E	1	INTEGER_16	100	
C00682	L_Compare_1: Window	Linear value	23893	5D55	E	1	INTEGER_16	100	
C00685	L_Compare_2: Fct.	Selection list	23890	5D52	E	1	UNSIGNED_8	1	
C00686	L_Compare_2: Hysteresis	Linear value	23889	5D51	E	1	INTEGER_16	100	
C00687	L_Compare_2: Window	Linear value	23888	5D50	E	1	INTEGER_16	100	
C00690	L_Compare_3: Function	Selection list	23885	5D4D	E	1	UNSIGNED_8	1	
C00691	L_Compare_3: Hysteresis	Linear value	23884	5D4C	E	1	INTEGER_16	100	
C00692	L_Compare_3: Window	Linear value	23883	5D4B	E	1	INTEGER_16	100	
C00693	L_Compare 4-5: Fct.	Selection list	23882	5D4A	A	2	UNSIGNED_8	1	
C00694	L_Compare 4-5: Hysteresis	Linear value	23881	5D49	A	2	INTEGER_16	100	
C00695	L_Compare 4-5: Window	Linear value	23880	5D48	A	2	INTEGER_16	100	
C00696	L_OffsetGainP_1: Offset	Linear value	23879	5D47	E	1	INTEGER_16	100	
C00697	L_OffsetGainP_2: Offset	Linear value	23878	5D46	E	1	INTEGER_16	100	
C00698	L_OffsetGainP_3: Offset	Linear value	23877	5D45	E	1	INTEGER_16	100	
C00699	L_MulDiv_1: Parameter	Linear value	23876	5D44	A	2	INTEGER_16	1	
C00700	LA_NCtrl: Analog connection list	Selection list	23875	5D43	A	29	UNSIGNED_16	1	
C00701	LA_NCtrl: Digital connection list	Selection list	23874	5D42	A	48	UNSIGNED_16	1	
C00710	LA_TabPos: Analog connection list	Selection list	23865	5D39	A	31	UNSIGNED_16	1	
C00711	LA_TabPos: Digital connection list	Selection list	23864	5D38	A	56	UNSIGNED_16	1	
C00712	LA_TabPos: phi connection list	Selection list	23863	5D37	A	3	UNSIGNED_16	1	
C00720	L_DigitalDelay_1: Delay	Linear value	23855	5D2F	A	2	UNSIGNED_32	1000	
C00721	L_DigitalDelay 2..3: Delay	Linear value	23854	5D2E	A	4	UNSIGNED_32	1000	
C00725	Current switching frequency	Selection list	23850	5D2A	E	1	UNSIGNED_8	1	
C00727	LS_Keypad digital values	Linear value	23848	5D28	A	8	UNSIGNED_8	1	
C00728	Analog values - keypad	Linear value	23847	5D27	A	3	INTEGER_16	100	
C00729	Remote: Setpoint selection	Linear value	23846	5D26	A	2	INTEGER_16	100	
C00760	LA_SwitchPos: Analog connection list	Selection list	23815	5D07	A	25	UNSIGNED_16	1	
C00761	LA_SwitchPos: Digital connection list	Selection list	23814	5D06	A	47	UNSIGNED_16	1	
C00800	L_MPot_1: Upper limit	Linear value	23775	5CDF	E	1	INTEGER_16	100	
C00801	L_MPot_1: Lower limit	Linear value	23774	5CDE	E	1	INTEGER_16	100	
C00802	L_MPot_1: Acceleration time	Linear value	23773	5CDD	E	1	UNSIGNED_16	10	
C00803	L_MPot_1: Deceleration time	Linear value	23772	5CDC	E	1	UNSIGNED_16	10	
C00804	L_MPot_1: Inactive fct.	Selection list	23771	5CDB	E	1	UNSIGNED_8	1	
C00805	L_MPot_1: Init fct.	Selection list	23770	5CDA	E	1	UNSIGNED_8	1	
C00806	L_MPot_1: Use	Selection list	23769	5CD9	E	1	UNSIGNED_8	1	
C00807	L_NLim_1: Max.SkipFrq.	Linear value	23768	5CD8	A	3	INTEGER_16	100	
C00808	L_NLim_1: Min.SkipFrq.	Linear value	23767	5CD7	A	3	INTEGER_16	100	
C00809	L_NLim_2: Max. skip freq.	Linear value	23766	5CD6	A	3	INTEGER_16	100	
C00810	L_NLim_2: Min. skip freq.	Linear value	23765	5CD5	A	3	INTEGER_16	100	
C00811	L_NLim_1: Current output value	Linear value	23764	5CD4	A	2	INTEGER_16	100	
C00812	L_NLim: Current status	Bit coded	23763	5CD3	A	2	UNSIGNED_16	1	
C00820	L_DigitalLogic_1: Function	Selection list	23755	5CCB	E	1	UNSIGNED_8	1	
C00821	L_DigitalLogic_1: Truth table	Selection list	23754	5CCA	A	24	UNSIGNED_8	1	
C00822	L_DigitalLogic_2: Function	Selection list	23753	5CC9	E	1	UNSIGNED_8	1	
C00823	L_DigitalLogic_2: Truth table	Selection list	23752	5CC8	A	24	UNSIGNED_8	1	
C00824	L_DigitalLogic5_1: Function	Selection list	23751	5CC7	E	1	UNSIGNED_8	1	
C00825	L_DigitalLogic5_1: Truth table	Selection list	23750	5CC6	A	32	UNSIGNED_8	1	
C00826	L_DigitalLogic5_2: Function	Selection list	23749	5CC5	E	1	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00827	L_DigitalLogic5_2: Truth table	Selection list	23748	5CC4	A	32	UNSIGNED_8	1	
C00828	L_DigitalLogic_3: Function	Selection list	23747	5CC3	E	1	UNSIGNED_8	1	
C00829	L_DigitalLogic_3: truth table	Selection list	23746	5CC2	A	24	UNSIGNED_8	1	
C00830	16-bit inputs [%]	Linear value	23745	5CC1	A	105	INTEGER_16	100	
C00831	16-bit inputs	Bit coded	23744	5CC0	A	105	UNSIGNED_16	1	
C00832	16-bit inputs [incr./ms]	Linear value	23743	5CBF	A	105	INTEGER_16	1	
C00833	Binary inputs	Selection list	23742	5CBE	A	131	UNSIGNED_8	1	
C00834	32-bit inputs [incr]	Linear value	23741	5CBD	A	14	INTEGER_32	1	
C00835	16-bit inputs [%] (Set2)	Linear value	23740	5CBC	A	162	INTEGER_16	100	
C00836	16-bit inputs (Set2)	Bit coded	23739	5CBB	A	162	UNSIGNED_16	1	
C00837	16-bit inputs [incr./ms] (Set2)	Linear value	23738	5CBA	A	162	INTEGER_16	1	
C00838	Binary inputs (Set2)	Selection list	23737	5CB9	A	213	UNSIGNED_8	1	
C00839	32-bit inputs [incr] (Set2)	Linear value	23736	5CB8	A	96	INTEGER_32	1	
C00840	16-bit inputs I/O level [%]	Linear value	23735	5CB7	A	111	INTEGER_16	100	
C00841	16-bit inputs I/O level	Bit coded	23734	5CB6	A	111	UNSIGNED_16	1	
C00843	Binary inputs I/O level	Selection list	23732	5CB4	A	199	UNSIGNED_8	1	
C00844	32-bit inputs I/O level [incr]	Linear value	23731	5CB3	A	25	INTEGER_32	1	
C00866	CAN input words	Bit coded	23709	5C9D	A	16	UNSIGNED_16	1	
C00868	CAN output words	Bit coded	23707	5C9B	A	16	UNSIGNED_16	1	
C00876	MCI input words	Bit coded	23699	5C93	A	16	UNSIGNED_16	1	
C00877	MCI output words	Bit coded	23698	5C92	A	16	UNSIGNED_16	1	
C00890	MCI_InOut: Inversion	Bit coded	23685	5C85	A	4	UNSIGNED_16	1	
C00905	Motor phase direction of rotation	Selection list	23670	5C76	E	1	UNSIGNED_8	1	CINH
C00909	Speed limitation	Linear value	23666	5C72	A	2	INTEGER_16	100	
C00910	Frequency limitation	Linear value	23665	5C71	A	2	UNSIGNED_16	1	
C00915	Motor cable length	Linear value	23660	5C6C	E	1	UNSIGNED_16	10	
C00916	Motor cable cross-section	Linear value	23659	5C6B	E	1	UNSIGNED_16	100	
C00917	Motor cable resistance	Linear value	23658	5C6A	E	1	UNSIGNED_16	1	
C00918	SC: Start motor magnetising current	Linear value	23657	5C69	E	1	UNSIGNED_8	1	
C00919	Load moment of inertia	Linear value	23656	5C68	A	1	UNSIGNED_32	100	
C00920	Rated device currents	Linear value	23655	5C67	A	8	UNSIGNED_16	10	
C00925	LS_Resolver: Number of pole pairs	Linear value	23650	5C62	E	1	UNSIGNED_8	1	
C00926	Pole position	Linear value	23649	5C61	A	3	INTEGER_16	10	
C00927	Motor rotor position	Linear value	23648	5C60	E	1	UNSIGNED_16	1	
C00936	SLPSM: Load value — speed controller	Linear value	23639	5C57	A	1	UNSIGNED_16	100	
C00937	Field-oriented motor currents	Linear value	23638	5C56	A	2	INTEGER_16	100	
C00938	PSM: Maximum motor current field weakening	Linear value	23637	5C55	E	1	UNSIGNED_16	100	
C00939	Ultimate motor current	Linear value	23636	5C54	E	1	UNSIGNED_16	10	
C00940	L_ConvW numerator	Linear value	23635	5C53	A	4	INTEGER_16	1	
C00941	L_ConvW denominator	Linear value	23634	5C52	A	4	INTEGER_16	1	
C00942	L_ConvW conversion method	Selection list	23633	5C51	A	4	UNSIGNED_8	1	
C00950	L_Interpolator_1: Activation FB functions	Selection list	23625	5C49	A	3	UNSIGNED_8	1	
C00951	L_Interpolator_1: No. of interpolation steps	Linear value	23624	5C48	E	1	UNSIGNED_16	1	
C00952	L_Interpolator_1: Limit value - error cycles	Linear value	23623	5C47	E	1	UNSIGNED_16	1	
C00953	L_Interpolator_1: Speed-up	Linear value	23622	5C46	E	1	UNSIGNED_8	1	
C00954	L_Interpolator_1: Synchronisation mode	Selection list	23621	5C45	E	1	UNSIGNED_8	1	
C00959	L_Curve: Current output value	Linear value	23616	5C40	A	3	INTEGER_16	100	
C00960	L_Curve_1: Selected curve type	Selection list	23615	5C3F	A	1	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00961	L_Curve_1: Input limitation	Linear value	23614	5C3E	A	2	INTEGER_16	100	
C00963	L_Curve_1: Table X-values	Linear value	23612	5C3C	A	32	INTEGER_16	1	
C00964	L_Curve_1: Table Y-values	Linear value	23611	5C3B	A	32	INTEGER_16	1	
C00965	Max. motor speed	Linear value	23610	5C3A	E	1	UNSIGNED_16	1	
C00966	VFC: Time const. slip comp.	Linear value	23609	5C39	E	1	UNSIGNED_16	1	
C00967	VFC: Frequency interpol. point n	Linear value	23608	5C38	A	11	INTEGER_16	10	
C00968	VFC: Voltage interpol. point n	Linear value	23607	5C37	A	11	UNSIGNED_16	100	
C00969	Motor parameters	Linear value	23606	5C36	A	1	UNSIGNED_8	1	
C00970	Rated device voltage	Linear value	23605	5C35	A	1	UNSIGNED_16	1	
C00971	VFC: V/f +encoder limitation	Linear value	23604	5C34	A	2	UNSIGNED_16	100	
C00972	VFC: Vp V/f +encoder	Linear value	23603	5C33	E	1	UNSIGNED_16	1000	
C00973	VFC: Ti V/f +encoder	Linear value	23602	5C32	E	1	UNSIGNED_16	10	
C00975	VFC-ECO: Vp CosPhi controller	Linear value	23600	5C30	E	1	UNSIGNED_16	1000	
C00976	VFC-ECO: Ti CosPhi controller	Linear value	23599	5C2F	E	1	UNSIGNED_16	10	
C00977	VFC-ECO: Minimum voltage V/f	Linear value	23598	5C2E	E	1	INTEGER_16	100	
C00978	VFC-ECO: Voltage reduction	Linear value	23597	5C2D	E	1	INTEGER_16	1	
C00979	Cosine phi	Linear value	23596	5C2C	A	2	INTEGER_16	100	
C00980	Performance indication	Linear value	23595	5C2B	A	4	INTEGER_32	1000	
C00981	Energy display	Linear value	23594	5C2A	A	2	INTEGER_32	100	
C00982	VFC-ECO: Voltage reduction ramp	Linear value	23593	5C29	E	1	UNSIGNED_8	10	
C00983	Delay	Linear value	23592	5C28	A	2	UNSIGNED_16	1	
C00985	SLVC: Gain of field current controller	Linear value	23590	5C26	E	1	INTEGER_16	100	
C00986	SLVC: Gain of cross current controller	Linear value	23589	5C25	E	1	INTEGER_16	100	
C00987	Inverter motor brake: nAdd	Linear value	23588	5C24	E	1	INTEGER_16	1	
C00988	Inverter motor brake: PT1 filter time	Linear value	23587	5C23	E	1	INTEGER_16	10	
C00989	Restart on the fly: Flying restart frequency fd_add	Linear value	23586	5C22	A	1	INTEGER_16	100	
C00990	Flying restart fct.: Activate	Selection list	23585	5C21	E	1	UNSIGNED_8	1	CINH
C00991	Flying restart fct.: Process	Selection list	23584	5C20	E	1	UNSIGNED_16	1	
C00992	Flying restart: Start frequency	Linear value	23583	5C1F	E	1	INTEGER_16	1	
C00993	Flying restart: Integration time	Linear value	23582	5C1E	E	1	UNSIGNED_16	10	
C00994	Flying restart: Current	Linear value	23581	5C1D	E	1	INTEGER_16	100	
C00995	SLPSM: Controlled current setpoint	Linear value	23580	5C1C	A	2	UNSIGNED_16	100	
C00996	SLPSM: Switching speed	Linear value	23579	5C1B	A	2	INTEGER_16	100	
C00997	SLPSM: Filter cutoff frequency	Linear value	23578	5C1A	E	1	INTEGER_16	100	
C00998	SLPSM: Filter time rotor position	Linear value	23577	5C19	A	2	INTEGER_16	10	
C00999	SLPSM: PLL gain	Linear value	23576	5C18	E	1	INTEGER_16	1	
C01000	MCTRL: Status	Bit coded	23575	5C17	E	1	UNSIGNED_16	1	
C01001	Manual entry of motor type	Selection list	23574	5C16	A	1	UNSIGNED_8	1	
C01004	Device command: Configuration	Bit coded	23571	5C13	A	1	UNSIGNED_16	1	
C01010	L_ArithmetikPhi 1-3: Function	Selection list	23565	5C0D	A	3	UNSIGNED_8	1	
C01011	L_ArithmetikPhi 4-6: Function	Selection list	23564	5C0C	A	3	UNSIGNED_8	1	
C01012	L_PhiDiv: byDivision	Linear value	23563	5C0B	A	1	INTEGER_8	1	
C01020	L_Odometer_1: Memory length	Selection list	23555	5C03	E	1	UNSIGNED_8	1	
C01021	L_Odometer_1: Memory type	Selection list	23554	5C02	E	1	UNSIGNED_8	1	
C01022	L_Odometer_1: Input selection	Selection list	23553	5C01	E	1	UNSIGNED_8	1	
C01023	L_Odometer_1: Edge selection	Selection list	23552	5C00	E	1	UNSIGNED_8	1	
C01025	L_Curve_2: Selected curve type	Selection list	23550	5BFE	A	1	UNSIGNED_8	1	
C01026	L_Curve_2: Input limitation	Linear value	23549	5BFD	A	2	INTEGER_16	100	
C01028	L_Curve_2: Table X-values	Linear value	23547	5BFB	A	32	INTEGER_16	1	
C01029	L_Curve_2: Table Y-values	Linear value	23546	5BFA	A	32	INTEGER_16	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C01030	L_Curve_3: Selected curve type	Selection list	23545	5BF9	A	1	UNSIGNED_8	1	
C01031	L_Curve_3: Input limitation	Linear value	23544	5BF8	A	2	INTEGER_16	100	
C01033	L_Curve_3: Table X-values	Linear value	23542	5BF6	A	32	INTEGER_16	1	
C01034	L_Curve_3: Table Y-values	Linear value	23541	5BF5	A	32	INTEGER_16	1	
C01035	L_Curve_3: SelectCurve	Selection list	23540	5BF4	A	1	UNSIGNED_16	1	
C01040	L_SRFG_1..2 linear ramp time	Linear value	23535	5BEF	A	2	UNSIGNED_32	1000	
C01041	L_SRFG_1..2 S-ramp time	Linear value	23534	5BEE	A	2	UNSIGNED_32	1000	
C01042	L_SRFG_1..2 limitations of output values	Linear value	23533	5BED	A	4	INTEGER_16	100	
C01045	L_ConvAP 1-3: numerator/denominator	Linear value	23530	5BEA	A	6	INTEGER_16	1	
C01046	L_ConvPA 1-3: byDivision	Linear value	23529	5BE9	A	3	INTEGER_8	1	
C01047	L_GearComp_1: Offset	Linear value	23528	5BE8	A	1	INTEGER_16	1	
C01048	L_GearComp_1: Num_Denom	Linear value	23527	5BE7	A	2	INTEGER_16	1	
C01049	L_CalcDiameter_1: Status	Linear value	23526	5BE6	A	1	INTEGER_16	1	
C01050	L_CalcDiameter_1: Diameter recalculation	Linear value	23525	5BE5	A	2	UNSIGNED_32	1000	
C01051	L_CalcDiameter_1: Filter time constant	Linear value	23524	5BE4	A	1	UNSIGNED_16	1000	
C01052	L_CalcDiameter_1: Web break monitoring	Linear value	23523	5BE3	A	1	INTEGER_16	100	
C01053	L_ProcessCtrl_1: Controller times	Linear value	23522	5BE2	A	5	UNSIGNED_16	1000	
C01054	L_ProcessCtrl_1: System deviation	Linear value	23521	5BE1	A	2	INTEGER_16	100	
C01055	L_ProcessCtrl_1: Correcting variable limitation	Selection list	23520	5BE0	A	1	UNSIGNED_8	1	
C01056	L_ProcessCtrl_1: Controller gain	Linear value	23519	5BDF	A	1	UNSIGNED_16	100	
C01057	L_CalcDiameter_1: Current diameter	Linear value	23518	5BDE	A	1	UNSIGNED_32	1000	
C01058	L_PosCtrlLin_1-2: bDisable	Selection list	23517	5BDD	A	2	UNSIGNED_8	1	
C01059	L_PosCtrlLin 1-2: Positioning behaviour	Selection list	23516	5BDC	A	2	UNSIGNED_8	1	
C01060	L_PosCtrlLin 1-2: Ramps	Linear value	23515	5BDB	A	6	INTEGER_32	1000	
C01061	L_PosCtrlLin 1-2: Traversing speed	Linear value	23514	5BDA	A	4	INTEGER_16	1	
C01062	L_SwitchPoint_1: Dead time	Linear value	23513	5BD9	A	8	UNSIGNED_16	1	
C01063	L_SwitchPoint_1: Hysteresis	Linear value	23512	5BD8	A	8	UNSIGNED_16	1	
C01064	L_SwitchPoint_1: CenterMode	Selection list	23511	5BD7	A	8	UNSIGNED_8	1	
C01065	L_SwitchPoint_1: Running time	Linear value	23510	5BD6	A	8	UNSIGNED_16	1	
C01066	L_SwitchPoint_1: Status	Selection list	23509	5BD5	A	8	INTEGER_16	1	
C01067	Inversion of gearbox stages	Selection list	23508	5BD4	A	3	UNSIGNED_8	1	
C01068	L_SwitchPoint_1: Invert	Bit coded	23507	5BD3	A	1	UNSIGNED_16	1	
C01069	L_DFSET_1: Ramp settings	Linear value	23506	5BD2	A	2	UNSIGNED_16	1	
C01070	L_DFSET_1: Angular trimming	Linear value	23505	5BD1	A	1	INTEGER_32	1	
C01071	L_DFSET_1: Following error limit	Linear value	23504	5BD0	A	1	UNSIGNED_32	1	
C01072	L_DFSET_1: Multiplier - angular trimming	Linear value	23503	5BCF	A	1	INTEGER_16	1	
C01073	L_DFSET_1: Adjustment	Bit coded	23502	5BCE	A	1	UNSIGNED_8	1	
C01074	L_DFSET_1: Zero pulse divider	Linear value	23501	5BCD	A	2	UNSIGNED_16	1	
C01075	L_DFSET_1: Synchronisation mode	Selection list	23500	5BCC	A	1	UNSIGNED_8	1	
C01076	L_DFRFG_1: Times	Linear value	23499	5BCB	A	2	UNSIGNED_32	1000	
C01077	L_DFRFG_1: Max. speed-up	Linear value	23498	5BCA	A	1	INTEGER_16	1	
C01078	L_DFRFG_1: Following error	Linear value	23497	5BC9	A	1	UNSIGNED_32	1	
C01079	L_DFRFG_1: Synchronisation window	Linear value	23496	5BC8	A	1	UNSIGNED_16	1	
C01080	L_DFRFG_1: Offset	Linear value	23495	5BC7	A	1	INTEGER_32	1	
C01081	L_DFRFG_1: Sync. direction / TP function	Selection list	23494	5BC6	A	1	UNSIGNED_8	1	
C01082	LS_WriteParamList: Execute Mode	Selection list	23493	5BC5	E	1	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C01083	LS_WriteParamList: Error status	Linear value	23492	5BC4	E	1	UNSIGNED_16	1	
C01084	LS_WriteParamList: Error line	Linear value	23491	5BC3	E	1	UNSIGNED_8	1	
C01085	LS_WriteParamList: Index	Linear value	23490	5BC2	A	32	INTEGER_32	1000	
C01086	LS_WriteParamList: WriteValue_1	Linear value	23489	5BC1	A	32	INTEGER_32	1	
C01087	LS_WriteParamList: WriteValue_2	Linear value	23488	5BC0	A	32	INTEGER_32	1	
C01088	LS_WriteParamList: WriteValue_3	Linear value	23487	5BBF	A	32	INTEGER_32	1	
C01089	LS_WriteParamList: WriteValue_4	Linear value	23486	5BBE	A	32	INTEGER_32	1	
C01090	LS_ParReadWrite 1-6: Index	Linear value	23485	5BBD	A	6	INTEGER_32	1000	
C01091	LS_ParReadWrite 1-6: Cycle time	Selection list	23484	5BBC	A	6	UNSIGNED_16	1	
C01092	LS_ParReadWrite 1-6: FailState	Linear value	23483	5BBB	A	6	UNSIGNED_16	1	
C01093	LS_ParReadWrite 1-6: Arithmetic mode	Selection list	23482	5BBA	A	6	UNSIGNED_8	1	
C01094	LS_ParReadWrite 1-6: Numerator	Linear value	23481	5BB9	A	6	INTEGER_16	1	
C01095	LS_ParReadWrite 1-6: Denominator	Linear value	23480	5BB8	A	6	INTEGER_16	1	
C01098	LS_ParReadWrite 1-6: Configuration	Bit coded	23477	5BB5	A	6	UNSIGNED_16	1	
C01100	Function L_Counter 1-3	Selection list	23475	5BB3	A	3	UNSIGNED_8	1	
C01101	Comparison L_Counter 1-3	Selection list	23474	5BB2	A	3	UNSIGNED_8	1	
C01108	L_SwitchPoint_1: Deadtime factor	Selection list	23467	5BAB	A	8	UNSIGNED_8	1	
C01109	L_SwitchPointPar_1: Deadtime factor	Selection list	23466	5BAA	A	8	UNSIGNED_8	1	
C01110	LS_MultiEncoder: Solid measure	Selection list	23465	5BA9	E	1	UNSIGNED_8	1	
C01111	LS_MultiEncoder: Encoder constant	Linear value	23464	5BA8	A	3	INTEGER_32	1	
C01112	LS_MultiEncoder: Position values	Linear value	23463	5BA7	A	4	INTEGER_32	10000	
C01119	LS_MultiEncoder: Current position	Linear value	23456	5BA0	A	2	INTEGER_32	10000	
C01120	Sync signal source	Selection list	23455	5B9F	E	1	UNSIGNED_8	1	
C01121	Sync cycle time setpoint	Linear value	23454	5B9E	E	1	UNSIGNED_16	1	
C01122	Sync phase position	Linear value	23453	5B9D	E	1	UNSIGNED_16	1	
C01123	Sync window	Linear value	23452	5B9C	E	1	UNSIGNED_16	1	
C01124	Sync correction width	Selection list	23451	5B9B	E	1	UNSIGNED_8	1	
C01130	LS_RetainData: Selection	Bit coded	23445	5B95	A	4	UNSIGNED_16	1	
C01131	LS_RetainData: 16Bit data	Linear value	23444	5B94	A	4	UNSIGNED_16	1	
C01132	LS_RetainData: 32Bit data	Linear value	23443	5B93	A	4	INTEGER_32	1	
C01133	LS_RetainData: Bool data	Selection list	23442	5B92	A	4	UNSIGNED_8	1	
C01134	LS_RetainData: 16Bit data	Linear value	23441	5B91	A	4	UNSIGNED_16	1	
C01135	LS_RetainData: 32Bit data	Linear value	23440	5B90	A	4	INTEGER_32	1	
C01136	LS_RetainData: Bool data	Selection list	23439	5B8F	A	4	UNSIGNED_8	1	
C01138	L_Transient 1-4: Function	Selection list	23437	5B8D	A	4	UNSIGNED_8	1	
C01139	L_Transient 1-4: Pulse duration	Linear value	23436	5B8C	A	4	UNSIGNED_16	1000	
C01140	L_Transient 5-8: Function	Selection list	23435	5B8B	A	4	UNSIGNED_8	1	
C01141	L_Transient 5-8 pulse duration	Linear value	23434	5B8A	A	4	UNSIGNED_16	1000	
C01150	L_PhaseIntK: Function	Selection list	23425	5B81	A	2	UNSIGNED_8	1	
C01151	L_PhaseIntK: Compare	Linear value	23424	5B80	A	2	INTEGER_32	1	
C01152	L_SwitchPointPar_1: Dead time	Linear value	23423	5B7F	A	8	UNSIGNED_16	1	
C01153	L_SwitchPointPar_1: Hysteresis	Linear value	23422	5B7E	A	8	UNSIGNED_16	1	
C01154	L_SwitchPointPar_1: CenterMode	Selection list	23421	5B7D	A	8	UNSIGNED_8	1	
C01155	L_SwitchPointPar_1: Running time	Linear value	23420	5B7C	A	8	UNSIGNED_16	1	
C01156	L_SwitchPointPar_1: Status	Selection list	23419	5B7B	A	8	INTEGER_16	1	
C01157	L_SwitchPointPar_1:Position	Linear value	23418	5B7A	A	16	INTEGER_32	10000	
C01158	L_SwitchPointPar_1: Invert	Bit coded	23417	5B79	A	1	UNSIGNED_16	1	
C01190	Motor thermal sensor	Selection list	23385	5B59	A	2	UNSIGNED_8	1	
C01191	PTC characteristic: Temperature 1/2	Linear value	23384	5B58	A	2	UNSIGNED_8	1	
C01192	PTC characteristic: Resistance 1/2	Linear value	23383	5B57	A	2	INTEGER_32	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C01193	Motor temp. feedback system	Selection list	23382	5B56	E	1	UNSIGNED_8	1	
C01201	MCK: Cycle	Linear value	23374	5B4E	A	1	INTEGER_32	10000	
C01202	MCK: iM motor/process	Linear value	23373	5B4D	A	2	UNSIGNED_16	1	
C01203	MCK: iG motor/position encoder	Linear value	23372	5B4C	A	2	UNSIGNED_16	1	
C01204	MCK: Feed constant	Linear value	23371	5B4B	E	1	INTEGER_32	10000	
C01205	MCK: Position resolution	Linear value	23370	5B4A	E	1	INTEGER_32	10000	
C01206	MCK: Mounting direction	Selection list	23369	5B49	A	3	UNSIGNED_8	1	CINH
C01210	MCK: Current positions	Linear value	23365	5B45	A	9	INTEGER_32	10000	
C01211	MCK: Speed	Linear value	23364	5B44	A	3	INTEGER_32	10000	
C01213	MCK: Max. traversing distance	Linear value	23362	5B42	A	1	INTEGER_32	1	
C01215	MCK: Following error	Linear value	23360	5B40	A	3	INTEGER_32	10000	
C01216	MCK: Positioning setting	Bit coded	23359	5B3F	E	1	UNSIGNED_8	1	
C01218	MCK: Position follower setting	Bit coded	23357	5B3D	E	1	UNSIGNED_8	1	
C01219	MCK: Speed follower setting	Bit coded	23356	5B3C	E	1	UNSIGNED_8	1	
C01220	MCK: Ref. setting	Bit coded	23355	5B3B	A	1	UNSIGNED_16	1	
C01221	MCK: Homing mode	Selection list	23354	5B3A	E	1	UNSIGNED_8	1	
C01222	MCK: Ref. M limit mode 14/15	Linear value	23353	5B39	E	1	INTEGER_16	100	
C01223	MCK: Ref. waiting time mode 14/15	Linear value	23352	5B38	E	1	UNSIGNED_16	1	
C01224	MCK: Ref. speeds	Linear value	23351	5B37	A	2	INTEGER_32	10000	
C01225	MCK: Ref. accelerations	Linear value	23350	5B36	A	2	INTEGER_32	10000	
C01226	MCK: Ref. S-ramp time	Linear value	23349	5B35	A	1	UNSIGNED_16	1000	
C01227	MCK: Ref. positions	Linear value	23348	5B34	A	2	INTEGER_32	10000	
C01228	MCK: Ref. sequence profile	Linear value	23347	5B33	E	1	UNSIGNED_8	1	
C01229	MCK: Position limiting values	Linear value	23346	5B32	A	2	INTEGER_32	10000	
C01230	Manual jog: Setting	Bit coded	23345	5B31	E	1	UNSIGNED_8	1	
C01231	MCK: Manual jog speeds	Linear value	23344	5B30	A	2	INTEGER_32	10000	
C01232	MCK: Manual jog accelerations	Linear value	23343	5B2F	A	2	INTEGER_32	10000	
C01233	MCK: Manual jog S-ramp time	Linear value	23342	5B2E	A	1	UNSIGNED_16	1000	
C01234	MCK: Manual jog breakpoints	Linear value	23341	5B2D	A	4	INTEGER_32	10000	
C01235	MCK: Manual jog waiting times	Linear value	23340	5B2C	A	1	UNSIGNED_16	1000	
C01236	MCK: Speed follower	Linear value	23339	5B2B	A	1	INTEGER_32	10000	
C01237	MCK: Acceleration follower	Linear value	23338	5B2A	A	2	INTEGER_32	10000	
C01238	MCK: S-ramp time follower	Linear value	23337	5B29	A	1	UNSIGNED_16	1000	
C01239	Setpoint holding	Linear value	23336	5B28	E	1	UNSIGNED_32	1	
C01240	MCK: Control word	Bit coded	23335	5B27	E	1	UNSIGNED_32	1	
C01241	MCK: Status word	Bit coded	23334	5B26	E	1	UNSIGNED_32	1	
C01242	MCK: Current pos profile number	Linear value	23333	5B25	E	1	UNSIGNED_8	1	
C01243	MCK: Current operating mode	Selection list	23332	5B24	E	1	UNSIGNED_8	1	
C01244	MCK: Target detection - times	Linear value	23331	5B23	A	3	UNSIGNED_16	1	
C01245	MCK: Target detection - positions	Linear value	23330	5B22	A	3	INTEGER_32	10000	
C01246	MCK: Select signal source	Selection list	23329	5B21	A	2	UNSIGNED_8	1	
C01247	MCK: Diagnostic word	Bit coded	23328	5B20	E	1	UNSIGNED_32	1	
C01248	MCK: Ref. step	Selection list	23327	5B1F	A	1	UNSIGNED_8	1	
C01251	MCK: Acceleration stop	Linear value	23324	5B1C	A	1	INTEGER_32	10000	
C01252	MCK: S-ramp times stop	Linear value	23323	5B1B	A	1	UNSIGNED_16	1000	
C01292	MCK: Positioning mode	Selection list	23283	5AF3	A	1	UNSIGNED_16	1	
C01293	L_MckStateInterface_1: Status	Bit coded	23282	5AF2	A	8	UNSIGNED_8	1	
C01294	Mode: Position calculation	Selection list	23281	5AF1	A	3	UNSIGNED_8	1	
C01295	L_MCKStateInterface_1: Pos. selection	Selection list	23280	5AF0	E	1	UNSIGNED_8	1	
C01296	Mode: Position calculation	Selection list	23279	5AEF	A	2	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C01297	L_MckCtrlInterface_1: Alternative function	Bit coded	23278	5AEE	E	1	UNSIGNED_8	1	
C01298	MCK: Operating mode change with profile no.	Selection list	23277	5AED	A	4	UNSIGNED_8	1	
C01299	MCKI: Status MCKInterface	Bit coded	23276	5AEC	E	1	UNSIGNED_8	1	
C01300	Profile data: Positioning mode	Selection list	23275	5AEB	A	15	UNSIGNED_8	1	
C01301	Profile data: Position	Linear value	23274	5AEA	A	15	INTEGER_32	10000	
C01302	Profile data: Speed	Linear value	23273	5AE9	A	15	INTEGER_32	10000	
C01303	Profile data: Acceleration	Linear value	23272	5AE8	A	15	INTEGER_32	10000	
C01304	Profile data: Deceleration	Linear value	23271	5AE7	A	15	INTEGER_32	10000	
C01305	Profile data: Final speed	Linear value	23270	5AE6	A	15	INTEGER_32	10000	
C01306	Profile data: S-ramp time	Linear value	23269	5AE5	A	15	UNSIGNED_16	1000	
C01307	Profile data: Sequence profile	Linear value	23268	5AE4	A	15	UNSIGNED_8	1	
C01308	Profile data: TP profile	Linear value	23267	5AE3	A	15	UNSIGNED_8	1	
C01309	Profile data: TP signal source	Selection list	23266	5AE2	A	15	UNSIGNED_8	1	
C01310	Profile data: PI position	Selection list	23265	5AE1	A	15	UNSIGNED_8	1	
C01311	Profile data: PI speed	Selection list	23264	5AE0	A	15	UNSIGNED_8	1	
C01312	Profile data: PI accel.	Selection list	23263	5ADF	A	15	UNSIGNED_8	1	
C01313	Profile data: PI decel.	Selection list	23262	5ADE	A	15	UNSIGNED_8	1	
C01314	Profile data: PI final speed	Selection list	23261	5ADD	A	15	UNSIGNED_8	1	
C01315	Profile data: PI S-ramp time	Selection list	23260	5ADC	A	15	UNSIGNED_8	1	
C01320	Profile data: Position	Linear value	23255	5AD7	A	15	INTEGER_32	10000	
C01321	Profile data: Speed	Linear value	23254	5AD6	A	15	INTEGER_32	10000	
C01322	Profile data: Acceleration	Linear value	23253	5AD5	A	15	INTEGER_32	10000	
C01323	Profile data: Deceleration	Linear value	23252	5AD4	A	15	INTEGER_32	10000	
C01324	Profile data: Final speed	Linear value	23251	5AD3	A	15	INTEGER_32	10000	
C01325	Profile data: S-ramp time	Linear value	23250	5AD2	A	15	UNSIGNED_16	1000	
C01350	ACDrive: Drive mode	Selection list	23225	5AB9	A	1	UNSIGNED_8	1	
C01351	ACDrive: Control word	Bit coded	23224	5AB8	A	1	UNSIGNED_16	1	
C01352	ACDrive: Status word	Bit coded	23223	5AB7	A	1	UNSIGNED_16	1	
C01353	ACDrive: Setpoint scaling	Linear value	23222	5AB6	A	2	INTEGER_8	1	
C01400	L_Sequencer_1: Sequence step	Linear value	23175	5A87	A	100	UNSIGNED_16	1	
C01401	L_Sequencer_1: Current step	Linear value	23174	5A86	A	1	UNSIGNED_16	1	
C01402	L_Sequencer_1: Sequence control: Status	Linear value	23173	5A85	A	1	UNSIGNED_16	1	
C01403	L_Sequencer_1: Sequence control: Control word	Linear value	23172	5A84	A	1	UNSIGNED_16	1	
C01404	L_Sequencer_1: Step for bCancel = TRUE	Linear value	23171	5A83	A	1	UNSIGNED_16	1	
C01405	L_Sequencer_1: PSInput	Selection list	23170	5A82	A	50	UNSIGNED_16	1	
C01406	L_Sequencer_1: PSLevel	Linear value	23169	5A81	A	50	UNSIGNED_8	1	
C01407	L_Sequencer_1: PSProfil	Linear value	23168	5A80	A	50	UNSIGNED_16	1	
C01408	L_Sequencer_1: PSStep	Linear value	23167	5A7F	A	50	UNSIGNED_16	1	
C01409	L_Sequencer_1: PS_WD_Time	Linear value	23166	5A7E	A	50	UNSIGNED_32	1000	
C01410	L_Sequencer_1: PS_WD_Step	Linear value	23165	5A7D	A	50	UNSIGNED_16	1	
C01411	L_Sequencer_1: SwitchOut1	Selection list	23164	5A7C	A	16	UNSIGNED_16	1	
C01412	L_Sequencer_1: SwitchLevel1	Linear value	23163	5A7B	A	16	UNSIGNED_8	1	
C01413	L_Sequencer_1: SwitchOut2	Selection list	23162	5A7A	A	16	UNSIGNED_16	1	
C01414	L_Sequencer_1: SwitchLevel2	Linear value	23161	5A79	A	16	UNSIGNED_8	1	
C01415	L_Sequencer_1: BranchIn	Selection list	23160	5A78	A	16	UNSIGNED_16	1	
C01416	L_Sequencer_1: BranchLevel	Linear value	23159	5A77	A	16	UNSIGNED_8	1	
C01417	L_Sequencer_1: BranchStep	Linear value	23158	5A76	A	16	UNSIGNED_16	1	
C01418	L_Sequencer_1: BranchStep1	Linear value	23157	5A75	A	2	UNSIGNED_16	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C01419	L_Sequencer_1: BranchStep2	Linear value	23156	5A74	A	2	UNSIGNED_16	1	
C01420	L_Sequencer_1: BranchStep3	Linear value	23155	5A73	A	2	UNSIGNED_16	1	
C01421	L_Sequencer_1: BranchStep4	Linear value	23154	5A72	A	2	UNSIGNED_16	1	
C01422	L_Sequencer_1: BranchStep5	Linear value	23153	5A71	A	2	UNSIGNED_16	1	
C01423	L_Sequencer_1: BranchStep6	Linear value	23152	5A70	A	2	UNSIGNED_16	1	
C01424	L_Sequencer_1: BranchStep7	Linear value	23151	5A6F	A	2	UNSIGNED_16	1	
C01425	L_Sequencer_1: BranchStep8	Linear value	23150	5A6E	A	2	UNSIGNED_16	1	
C01426	L_Sequencer_1: BranchStep9	Linear value	23149	5A6D	A	2	UNSIGNED_16	1	
C01427	L_Sequencer_1: BranchStep10	Linear value	23148	5A6C	A	2	UNSIGNED_16	1	
C01428	L_Sequencer_1: BranchStep11	Linear value	23147	5A6B	A	2	UNSIGNED_16	1	
C01429	L_Sequencer_1: BranchStep12	Linear value	23146	5A6A	A	2	UNSIGNED_16	1	
C01430	L_Sequencer_1: BranchStep13	Linear value	23145	5A69	A	2	UNSIGNED_16	1	
C01431	L_Sequencer_1: BranchStep14	Linear value	23144	5A68	A	2	UNSIGNED_16	1	
C01432	L_Sequencer_1: BranchStep15	Linear value	23143	5A67	A	2	UNSIGNED_16	1	
C01433	L_Sequencer_1: BranchStep16	Linear value	23142	5A66	A	2	UNSIGNED_16	1	
C01434	L_Sequencer_1: BranchStep17	Linear value	23141	5A65	A	2	UNSIGNED_16	1	
C01435	L_Sequencer_1: BranchStep18	Linear value	23140	5A64	A	2	UNSIGNED_16	1	
C01436	L_Sequencer_1: BranchStep19	Linear value	23139	5A63	A	2	UNSIGNED_16	1	
C01437	L_Sequencer_1: BranchStep20	Linear value	23138	5A62	A	2	UNSIGNED_16	1	
C01438	L_Sequencer_1: Wartezeit	Linear value	23137	5A61	A	8	UNSIGNED_32	1000	
C01439	L_Sequencer_1: Wait Eingang f. Weiter	Selection list	23136	5A60	A	8	UNSIGNED_16	1	
C01440	L_Sequencer_1: WaitLevel	Linear value	23135	5A5F	A	8	UNSIGNED_8	1	
C01441	L_Sequencer_1: SetZählernummer	Linear value	23134	5A5E	A	5	UNSIGNED_16	1	
C01442	L_Sequencer_1: SetZählerStartwert	Linear value	23133	5A5D	A	5	INTEGER_32	1	
C01443	L_Sequencer_1: Aktueller Zählerstand	Linear value	23132	5A5C	A	5	INTEGER_32	1	
C01444	L_Sequencer_1: ZählerNummer	Linear value	23131	5A5B	A	8	UNSIGNED_16	1	
C01445	L_Sequencer_1: SchrittwertZähler	Linear value	23130	5A5A	A	8	INTEGER_32	1	
C01446	L_Sequencer_1: ZählerVergleichswert	Linear value	23129	5A59	A	8	INTEGER_32	1	
C01447	L_Sequencer_1: ZählerSprungziel	Linear value	23128	5A58	A	8	UNSIGNED_16	1	
C01448	L_Sequencer_1: ZählerVergleichsfunktion	Selection list	23127	5A57	A	8	UNSIGNED_16	1	
C01449	L_Sequencer_1: Standby Eingang f. Ende	Selection list	23126	5A56	A	5	UNSIGNED_16	1	
C01450	L_Sequencer_1: Standby PolaritätEingang	Linear value	23125	5A55	A	5	UNSIGNED_8	1	
C01451	L_Sequencer_1: Standby Modusfolger	Selection list	23124	5A54	A	5	UNSIGNED_16	1	
C01452	L_Sequencer_1: StandbySollwert	Linear value	23123	5A53	A	5	INTEGER_16	100	
C01453	L_Sequencer_1: Bool signal	Linear value	23122	5A52	A	16	UNSIGNED_8	1	
C01454	L_Sequencer_1: INT signal	Linear value	23121	5A51	A	1	INTEGER_16	100	
C01455	L_Sequencer_1: Word signal	Linear value	23120	5A50	A	13	UNSIGNED_16	1	
C01460	L_ConvActPos: TConst	Linear value	23115	5A4B	A	1	UNSIGNED_16	1	
C01461	L_ConvActPos: MaxPos/MinPos	Linear value	23114	5A4A	A	2	INTEGER_16	100	
C01462	L_ConvActPos: Length	Linear value	23113	5A49	A	1	UNSIGNED_32	1	
C01463	L_ConvActPos: UpperRef/LowerRef	Linear value	23112	5A48	A	2	INTEGER_16	100	
C01464	L_ConvActPos: WindowSetPos	Linear value	23111	5A47	A	1	INTEGER_16	100	
C01465	L_ConvActPos: DelayTime	Linear value	23110	5A46	A	1	UNSIGNED_16	1000	
C01466	L_ConvActPos: TConstVAdd	Linear value	23109	5A45	A	1	UNSIGNED_16	1	
C01467	L_ConvActPos: DeadBand	Linear value	23108	5A44	A	1	INTEGER_16	100	
C01468	L_ConvActPos: Damping	Linear value	23107	5A43	A	1	UNSIGNED_16	100	
C01469	L_MFail: Vp	Linear value	23106	5A42	A	1	UNSIGNED_16	1000	
C01470	L_MFail: Tn	Linear value	23105	5A41	A	3	UNSIGNED_16	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C01501	Resp. to communication error with MCI	Selection list	23074	5A22	A	2	UNSIGNED_8	1	
C01670	L_ComparePhi 1-5: Function	Selection list	22905	5979	A	5	UNSIGNED_8	1	
C01671	L_ComparePhi 1-5: Hysteresis	Linear value	22904	5978	A	5	INTEGER_32	1	
C01672	L_ComparePhi 1-5: Window	Linear value	22903	5977	A	5	INTEGER_32	1	
C01700	Energy saving mode: Mode	Linear value	22875	595B	A	2	UNSIGNED_8	1	
C01701	Energy saving mode: toff min	Linear value	22874	595A	A	1	UNSIGNED_16	1	
C01702	Energy saving mode: toff	Linear value	22873	5959	A	1	UNSIGNED_16	1	
C01703	Energy saving mode: ton	Linear value	22872	5958	A	1	UNSIGNED_16	1	
C01704	Energy saving mode: Function	Bit coded	22871	5957	A	1	UNSIGNED_16	1	
C01709	Energy saving mode: Status	Linear value	22866	5952	A	1	UNSIGNED_8	1	
C01770	Filter time - earth-fault detect. is running	Linear value	22805	5915	E	1	UNSIGNED_8	1	
C01902	Diagnostics X6: Max. baud rate	Selection list	22673	5891	E	1	UNSIGNED_16	1	
C01903	Diagnostics X6: Change baud rate	Selection list	22672	5890	E	1	UNSIGNED_8	1	
C01905	Diagnostics X6: Current baud rate	Linear value	22670	588E	E	1	UNSIGNED_32	1	
C02200	LS_WriteParamList: Function	Selection list	22375	5767	A	2	UNSIGNED_8	1	
C02210	LS_WriteParamList: Motor control	Selection list	22365	575D	A	4	UNSIGNED_8	1	
C02212	LS_WriteParamList: VFC: V/f base frequency	Linear value	22363	575B	A	4	UNSIGNED_16	10	
C02213	LS_WriteParamList: VFC: Vmin boost	Linear value	22362	575A	A	4	UNSIGNED_16	100	
C02214	LS_WriteParamList: Switching frequency	Selection list	22361	5759	A	4	UNSIGNED_8	1	
C02215	LS_WriteParamList: Auto-DCB: Threshold	Linear value	22360	5758	A	4	UNSIGNED_16	1	
C02216	LS_WriteParamList: Slip compensation	Linear value	22359	5757	A	4	INTEGER_16	100	
C02217	LS_WriteParamList: Imax in motor mode	Linear value	22358	5756	A	4	UNSIGNED_16	100	
C02218	LS_WriteParamList: Imax in generator mode	Linear value	22357	5755	A	4	INTEGER_16	100	
C02219	LS_WriteParamList: DC braking: Current	Linear value	22356	5754	A	4	INTEGER_16	100	
C02220	LS_WriteParamList: Vp speed controller	Linear value	22355	5753	A	12	UNSIGNED_16	100	
C02221	LS_WriteParamList: Ti speed controller	Linear value	22354	5752	A	12	UNSIGNED_16	10	
C02222	LS_WriteParamList: SC: Tdn speed controller	Linear value	22353	5751	A	4	UNSIGNED_16	100	
C02223	LS_WriteParamList: Imax/M controller gain	Linear value	22352	5750	A	8	UNSIGNED_16	100	
C02224	LS_WriteParamList: Imax/M controller reset time	Linear value	22351	574F	A	8	UNSIGNED_16	1	
C02225	LS_WriteParamList: Vp current controller	Linear value	22350	574E	A	4	UNSIGNED_16	100	
C02226	LS_WriteParamList: Ti current controller	Linear value	22349	574D	A	4	UNSIGNED_16	100	
C02227	LS_WriteParamList: SC: Vp field controller	Linear value	22348	574C	A	4	UNSIGNED_16	100	
C02228	LS_WriteParamList: SC: Tn field controller	Linear value	22347	574B	A	4	UNSIGNED_16	10	
C02229	LS_WriteParamList: SC: settings	Selection list	22346	574A	A	16	UNSIGNED_8	1	
C02230	LS_WriteParamList: Override point of field weakening	Linear value	22345	5749	A	4	INTEGER_16	1	
C02231	LS_WriteParamList: Rated motor power	Linear value	22344	5748	A	4	UNSIGNED_16	100	
C02232	LS_WriteParamList: Motor rotor resistance	Linear value	22343	5747	A	4	UNSIGNED_32	1	
C02233	LS_WriteParamList: Motor stator resistance	Linear value	22342	5746	A	4	UNSIGNED_32	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02234	LS_WriteParamList: Motor stator leakage inductance	Linear value	22341	5745	A	4	UNSIGNED_16	100	
C02236	LS_WriteParamList: Rated motor speed	Linear value	22339	5743	A	4	UNSIGNED_16	1	
C02237	LS_WriteParamList: Rated motor current	Linear value	22338	5742	A	4	UNSIGNED_16	100	
C02238	LS_WriteParamList: Rated motor frequency	Linear value	22337	5741	A	4	UNSIGNED_16	1	
C02239	LS_WriteParamList: Rated motor voltage	Linear value	22336	5740	A	4	UNSIGNED_16	1	
C02240	LS_WriteParamList: Motor cosine phi	Linear value	22335	573F	A	4	UNSIGNED_8	100	
C02241	LS_WriteParamList: Motor magnetizing inductance	Linear value	22334	573E	A	4	UNSIGNED_16	10	
C02242	LS_WriteParamList: Motor magnetising current	Linear value	22333	573D	A	4	UNSIGNED_16	100	
C02244	LS_WriteParamList: Auto-DCB: hold time	Linear value	22331	573B	A	4	UNSIGNED_32	1000	
C02245	LS_WriteParamList: DC braking: Hold time	Linear value	22330	573A	A	4	UNSIGNED_32	1000	
C02246	LS_WriteParamList: Setting of motor overload (I²xt)	Linear value	22329	5739	A	4	INTEGER_16	100	
C02249	LS_WriteParamList: Oscillation damping influence	Linear value	22326	5736	A	4	UNSIGNED_16	100	
C02250	LS_WriteParamList: Filter time - oscill. damping	Linear value	22325	5735	A	4	UNSIGNED_8	1	
C02251	LS_WriteParamList: Oscillation damping field weakening	Linear value	22324	5734	A	4	UNSIGNED_8	1	
C02252	LS_WriteParamList: Kp position controller	Linear value	22323	5733	A	4	UNSIGNED_16	100	
C02256	LS_WriteParamList: Moment of inertia	Linear value	22319	572F	A	4	UNSIGNED_32	100	
C02260	LS_WriteParamList: Speed sensor selection	Selection list	22315	572B	A	4	UNSIGNED_8	1	
C02261	LS_WriteParamList: SC: Field feedforward control	Linear value	22314	572A	A	4	UNSIGNED_16	1	
C02262	LS_WriteParamList: SC: Vp field weakening controller	Linear value	22313	5729	A	4	UNSIGNED_16	10000	
C02263	LS_WriteParamList: SC: Tn field weakening controller	Linear value	22312	5728	A	4	UNSIGNED_16	10	
C02264	LS_WriteParamList: Sensibility - Setpoint feedforward control	Selection list	22311	5727	A	4	UNSIGNED_8	1	
C02272	LS_WriteParamList: Motor phase direction of rotation	Selection list	22303	571F	A	4	UNSIGNED_8	1	CINH
C02273	LS_WriteParamList: Speed limitation	Linear value	22302	571E	A	8	INTEGER_16	100	
C02274	LS_WriteParamList: Frequency limitation	Linear value	22301	571D	A	8	UNSIGNED_16	1	
C02275	LS_WriteParamList: Motor cable length	Linear value	22300	571C	A	4	UNSIGNED_16	10	
C02276	LS_WriteParamList: Motor cable cross-section	Linear value	22299	571B	A	4	UNSIGNED_16	100	
C02278	LS_WriteParamList: PSM: Maximum motor current field weakening	Linear value	22297	5719	A	4	UNSIGNED_16	100	
C02279	LS_WriteParamList: Ultimate motor current	Linear value	22296	5718	A	4	UNSIGNED_16	10	
C02280	LS_WriteParamList: Max. motor speed	Linear value	22295	5717	A	4	UNSIGNED_16	1	
C02281	LS_WriteParamList: VFC: Time const. slip comp.	Linear value	22294	5716	A	4	UNSIGNED_16	1	
C02284	LS_WriteParamList: VFC: limitation V/f +encoder	Linear value	22291	5713	A	8	UNSIGNED_16	100	
C02285	LS_WriteParamList: VFC: Vp V/f +encoder	Linear value	22290	5712	A	4	UNSIGNED_16	1000	
C02286	LS_WriteParamList: VFC: Ti V/f +encoder	Linear value	22289	5711	A	4	UNSIGNED_16	10	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02287	LS_WriteParamList: VFC-ECO: Vp CosPhi controller	Linear value	22288	5710	A	4	UNSIGNED_16	1000	
C02288	LS_WriteParamList: VFC-ECO: Ti CosPhi controller	Linear value	22287	570F	A	4	UNSIGNED_16	10	
C02289	LS_WriteParamList: VFC-ECO: Minimum voltage U/f	Linear value	22286	570E	A	4	INTEGER_16	100	
C02290	LS_WriteParamList: VFC-ECO: Voltage reduction ramp	Linear value	22285	570D	A	4	UNSIGNED_8	10	
C02291	LS_WriteParamList: SLVC: Field current controller gain	Linear value	22284	570C	A	4	INTEGER_16	100	
C02292	LS_WriteParamList: SLVC: Cross current controller gain	Linear value	22283	570B	A	4	INTEGER_16	100	
C02293	LS_WriteParamList: Inverter motor brake: nAdd	Linear value	22282	570A	A	4	INTEGER_16	1	
C02294	LS_WriteParamList: Inverter motor brake: PT1 filter time	Linear value	22281	5709	A	4	INTEGER_16	10	
C02295	LS_WriteParamList: Flying restart fct.: activation	Selection list	22280	5708	A	4	UNSIGNED_8	1	CINH
C02296	LS_WriteParamList: Flying restart fct.: process	Selection list	22279	5707	A	4	UNSIGNED_16	1	
C02297	LS_WriteParamList: Flying restart: start frequency	Linear value	22278	5706	A	4	INTEGER_16	1	
C02298	LS_WriteParamList: Flying restart fct: int. time	Linear value	22277	5705	A	4	UNSIGNED_16	10	
C02299	LS_WriteParamList: Flying restart fct.: current	Linear value	22276	5704	A	4	INTEGER_16	100	
C02300	LS_WriteParamList: SLPSM: Controlled current setpoint	Linear value	22275	5703	A	8	UNSIGNED_16	100	
C02301	LS_WriteParamList: SLPSM: Switching speed	Linear value	22274	5702	A	8	INTEGER_16	100	
C02302	LS_WriteParamList: SLPSM: Filter time - rotor position	Linear value	22273	5701	A	4	INTEGER_16	100	
C02303	LS_WriteParamList: SLPSM: Filter time rotor position	Linear value	22272	5700	A	8	INTEGER_16	10	
C02304	LS_WriteParamList: SLPSM: PLL gain	Linear value	22271	56FF	A	4	INTEGER_16	1	
C02305	LS_WriteParamList: PSM: Ppp saturation characteristic	Linear value	22270	56FE	A	68	UNSIGNED_8	1	
C02306	LS_WriteParamList: PSM: Imax Ppp saturation characteristic	Linear value	22269	56FD	A	4	UNSIGNED_16	10	
C02307	LS_WriteParamList: PSM: Activate Ppp saturation char.	Selection list	22268	56FC	A	4	UNSIGNED_8	1	
C02311	LS_WriteParamList: PLI without motion: Adaptation of time duration	Linear value	22264	56F8	A	4	INTEGER_8	1	
C02312	LS_WriteParamList: PLI without motion	Bit coded	22263	56F7	A	4	UNSIGNED_16	1	
C02313	LS_WriteParamList: PLI without motion: Adaptation of ident angle	Linear value	22262	56F6	A	4	INTEGER_8	1	
C02315	LS_WriteParamList: Manual entry of motor type	Selection list	22260	56F4	A	4	UNSIGNED_8	1	
C02430	Axis bus address and no. of nodes	Linear value	22145	5681	A	2	UNSIGNED_8	1	
C02431	Axis bus time settings	Linear value	22144	5680	A	3	UNSIGNED_16	1	
C02435	Axis bus status	Selection list	22140	567C	E	1	UNSIGNED_8	1	
C02436	Axis bus error status	Selection list	22139	567B	E	1	UNSIGNED_8	1	
C02437	Axis bus MessageError	Selection list	22138	567A	E	1	UNSIGNED_8	1	
C02438	CAB_Tx_Rx_diagnosis	Linear value	22137	5679	A	4	UNSIGNED_16	1	
C02440	AxisBusIO slave/master	Selection list	22135	5677	A	1	UNSIGNED_8	1	
C02442	CAB decoupling inputs	Bit coded	22133	5675	A	1	UNSIGNED_16	1	
C02443	CAB decoupling value	Linear value	22132	5674	A	11	UNSIGNED_16	1	
C02444	CAB configuration	Bit coded	22131	5673	A	1	UNSIGNED_16	1	
C02556	Position controller: Limitation	Linear value	22019	5603	A	1	INTEGER_32	1	
C02580	Holding brake: Operating mode	Selection list	21995	55EB	E	1	UNSIGNED_8	1	

Code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02581	Holding brake: Speed thresholds	Linear value	21994	55EA	A	5	INTEGER_16	100	
C02582	Holding brake: Setting	Bit coded	21993	55E9	E	1	UNSIGNED_8	1	
C02589	Holding brake: Time system	Linear value	21986	55E2	A	4	UNSIGNED_16	1	
C02593	Holding brake: Activation time	Linear value	21982	55DE	A	4	UNSIGNED_32	1000	
C02607	Holding brake: Status	Bit coded	21968	55D0	E	1	UNSIGNED_16	1	
C02610	MCK: Accel./decel. times	Linear value	21965	55CD	A	3	UNSIGNED_32	1000	
C02611	MCK: Limitations	Linear value	21964	55CC	A	5	INTEGER_16	100	
C02652	Settings of measuring system	Bit coded	21923	55A3	E	1	UNSIGNED_16	1	
C02810	TP: Edge selection	Selection list	21765	5505	A	9	UNSIGNED_8	1	
C02811	TP: Sensor delay	Linear value	21764	5504	A	9	UNSIGNED_16	1	
C02812	TP: Position offset	Linear value	21763	5503	A	9	INTEGER_32	10000	
C02813	TP: Pos window start	Linear value	21762	5502	A	5	INTEGER_32	10000	
C02814	TP: Pos window end	Linear value	21761	5501	A	5	INTEGER_32	10000	
C02815	TP: Position source	Selection list	21760	5500	A	9	UNSIGNED_8	1	
C02816	TP: Signal counter	Linear value	21759	54FF	A	9	UNSIGNED_16	1	
C02817	TP: TouchProbe position	Linear value	21758	54FE	A	9	INTEGER_32	10000	
C02830	DIx: Debounce time	Selection list	21745	54F1	A	7	UNSIGNED_8	1	
C02840	CountInx: Parameter	Linear value	21735	54E7	A	4	UNSIGNED_32	1	
C02841	CountInx: Counter content	Linear value	21734	54E6	A	2	UNSIGNED_32	1	
C02842	FreqInxx: Offset	Linear value	21733	54E5	A	2	INTEGER_16	100	
C02843	FreqInxx: Gain	Linear value	21732	54E4	A	2	INTEGER_16	100	
C02844	FreqIn12: Function	Selection list	21731	54E3	A	1	UNSIGNED_8	1	
C02845	FreqIn12: PosIn comparison value	Linear value	21730	54E2	E	1	INTEGER_32	1	
C02853	PSM: Lss saturation characteristic	Linear value	21722	54DA	A	17	UNSIGNED_8	1	
C02855	PSM: Imax Lss saturation characteristic	Linear value	21720	54D8	E	1	UNSIGNED_16	10	
C02859	PSM: Activate Ppp saturation char.	Selection list	21716	54D4	E	1	UNSIGNED_8	1	
C02862	Resolver gain	Linear value	21713	54D1	A	2	UNSIGNED_16	100	
C02863	Resolver: Phase error	Linear value	21712	54D0	E	1	INTEGER_16	100	
C02864	MCTRL: Optimisations	Bit coded	21711	54CF	A	1	UNSIGNED_16	1	
C02865	MCTRL: Special settings	Bit coded	21710	54CE	A	1	UNSIGNED_16	1	
C02866	MCTRL: Special settings	Selection list	21709	54CD	A	3	UNSIGNED_8	1	CINH
C02867	Identification process	Selection list	21708	54CC	A	1	UNSIGNED_8	1	CINH
C02868	Setting of compatibility	Bit coded	21707	54CB	A	1	UNSIGNED_32	1	
C02869	MCTRL: Special settings 2	Bit coded	21706	54CA	A	1	UNSIGNED_16	1	
C02870	PLI without movement: degree of optimisation	Linear value	21705	54C9	A	1	INTEGER_16	100	
C02871	PLI without movement: runtime	Linear value	21704	54C8	A	1	INTEGER_16	100	
C02872	PLI without movement: adaptation of time duration	Linear value	21703	54C7	A	1	INTEGER_8	1	
C02873	PLI without movement: Ident. el. rotor displ. angle	Linear value	21702	54C6	A	1	INTEGER_16	1	
C02874	PLI without movement	Bit coded	21701	54C5	A	1	UNSIGNED_16	1	
C02875	PLI without movement: adaptation of ident angle	Linear value	21700	54C4	A	1	INTEGER_8	1	
C02876	PSM: Max. motor temperature	Linear value	21699	54C3	A	1	UNSIGNED_8	1	
C02877	PSM temperature coefficient	Linear value	21698	54C2	A	1	INTEGER_16	100	
C02878	KTY motor temperature compensation	Bit coded	21697	54C1	A	1	UNSIGNED_16	1	
C02879	Slip calculation from equivalent circuit diagram	Bit coded	21696	54C0	A	1	UNSIGNED_16	1	

18 Working with the FB Editor

The function block editor (in the following called "FB Editor") is available in the »Engineer« from the "StateLine" device version.

The FB Editor can be used to carry out an online monitoring of the technology application running in the device (e.g. for diagnostic purposes) and reconfigure the I/O interconnection of the technology application.

Functional range

The options for processing function block interconnections depend on the device version:

Function	Inverter Drives 8400		
	StateLine C	HighLine C	TopLine C
I/O interconnection can be reconfigured	●	●	●
Application interconnection can be reconfigured	●	●	●
Free interconnection	(●)	●	●
(●) Free interconnection for "StateLine C" is only available from version 12.00.00 and »Engineer« V2.17.			

Note!

The illustrations of the FB Editor user interface and the dialog boxes in this documentation are based on the »Engineer« V2.10.

18.1 Basics

Using the function block interconnection, any signal interconnection can be implemented. Various FBs are available for digital signal processing, signal conversion and logic modules.

For special tasks it has proved of value to use the integrated technology applications as a basis for modifications or extensions of the available FB interconnections. Depending on the device version (see Functional range), the experienced user has the opportunity to implement own drive solutions independent of the predefined technology applications by using the "free interconnection".

For this purpose, the FB Editor provides the following functions:

- Copying & pasting of interconnection elements (also device-independent)
- Export & import of the interconnection
- Comparison of two interconnections (also online <-> offline comparison)
- Overview window and zoom functions
- Comments on the signal flow
- Online monitoring

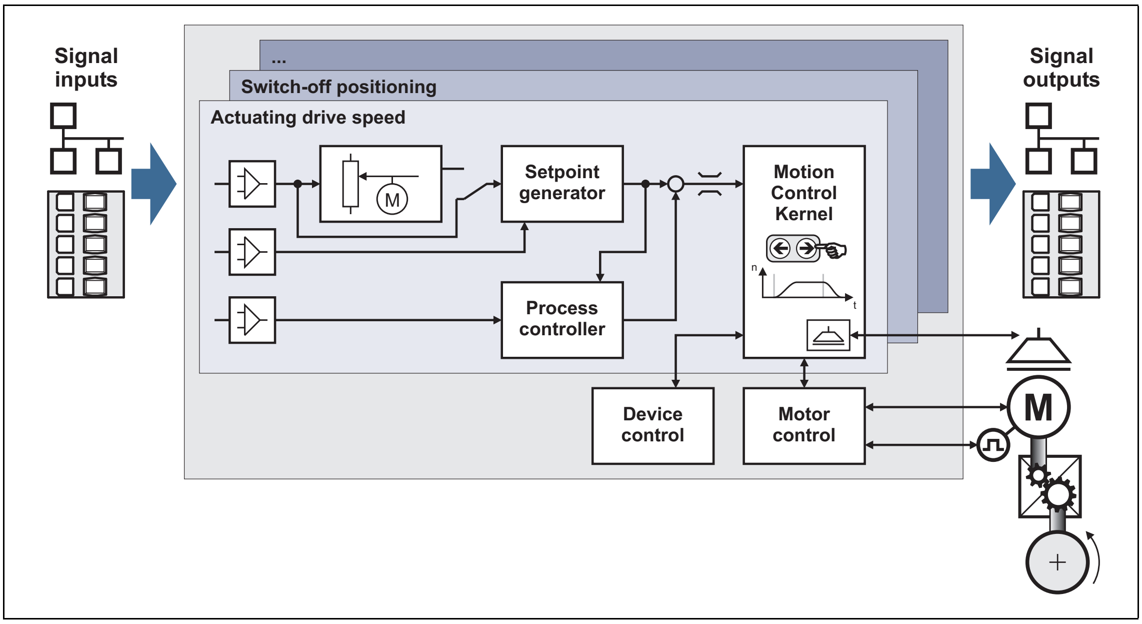
The option to mask out non-used inputs and outputs of modules is also sensible to minimise the complexity of the FB interconnection and to adapt the clarity of the interconnection to the customers need.

All graphical information of the FB interconnection view (positions of the FBs, line or flag presentation of the connection, visibility of the inputs/outputs) are saved with the parameter set in the memory module of the inverter and can be uploaded any time into the FB Editor of the »Engineer« even if the Engineer project is not available.

18.1.1 Basic components of a drive solution

A drive solution consists of the following basic components:

- Signal inputs (for control and setpoint signals)
- Signal flow of the technology application
- Signal outputs (for status and actual value signals)



[18-1] Basic components of a drive solution

Regarding the 8400 device series, these three components are available for the FB interconnection and classified as follows:

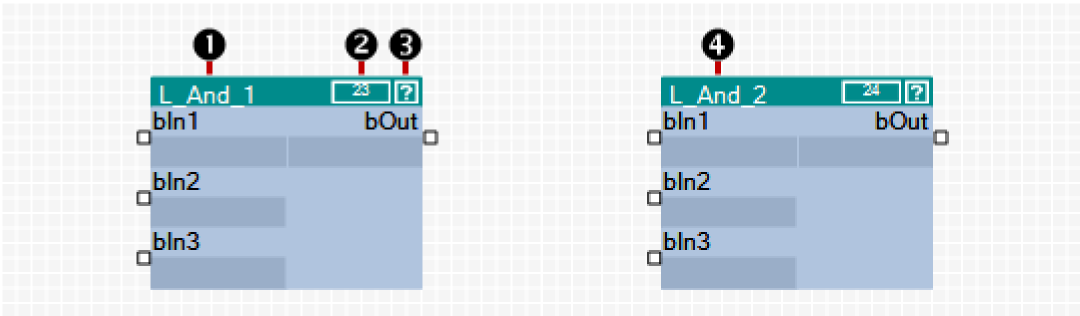
Module type	Name	Task	Example
Function block	L_*name*	General function block for free interconnection (only HighLine)	L_Compare_1 L_PCTRL_1
System block	LS_*name*	Signal interface to inverter-internal functions	LS_DigitalInput LS_DriveInterface
Port block	LP_*name*	• Process data communication via a fieldbus using a communication module • Process data communication via CAN on board	LP_CanIn1 LP_CanOut1 LP_MciIn LP_MciOut
Application block	LA_*name*	Block for a technology application	LA_NCtrl LA_SwitchPos

Further information on the individual modules can be obtained from the following subchapters!

18.1.1.1 What is a function block?

A function block (FB) can be compared with an integrated circuit that contains a certain control logic and provides one or several values when being executed.

- The function blocks are classified alphabetically in a "function library".
- Each function block has a unique identifier and a processing number which defines the position at which the function block is calculated during runtime.



❶ Name of the function block
❷ Processing number
❸ Button for opening the online help for the function block
❹ Another available function block of the same type (numbering in ascending order)

[18-2] Information on a function block in the FB Editor

Tip!

A detailed description of all available function blocks can be found in the main chapter "Function library". (📖 1468)

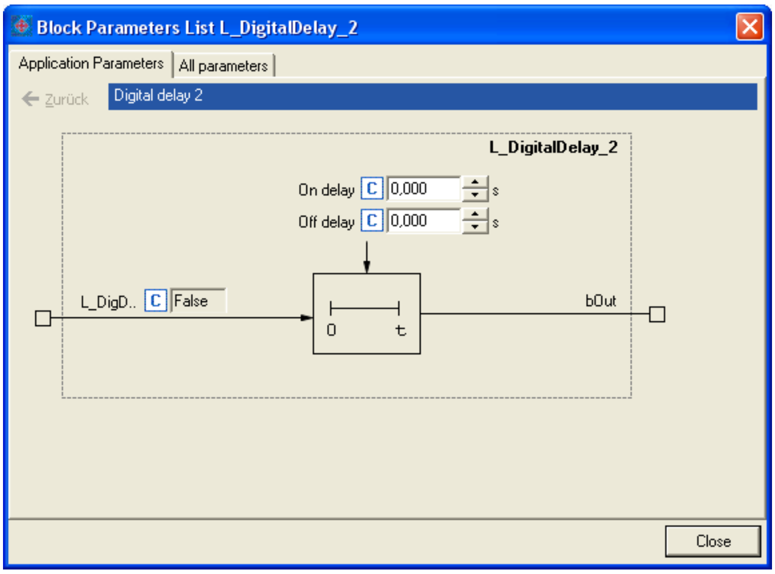
18.1.1.2 Parameterisable function blocks

Some function blocks have parameters which serve to change particular settings during operation, if required, or which display actual values & status information.

- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

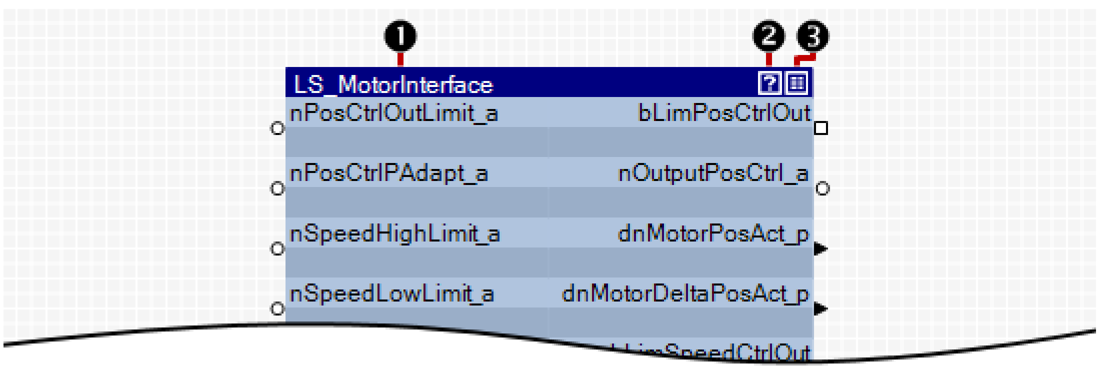
Example

Parameterisation dialog for the FB **L_DigitalDelay_2**:



18.1.1.3 What is a system block?

System blocks are a special variant of a function block. They partly activate real hardware, e. g. the digital and analog inputs/outputs and the motor control.



❶ Name of the system block
❷ Button for opening the online help for the system block
❸ Button for opening the parameterisation dialog or the parameter list for the system block

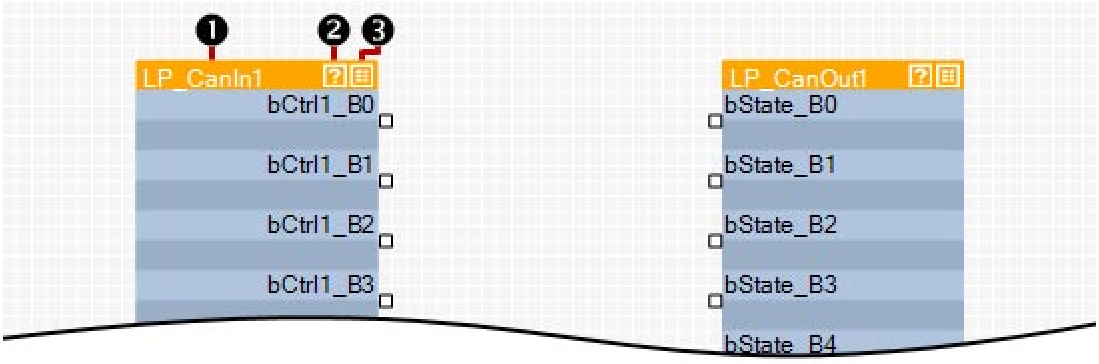
[18-3] Example: System block "LS_MotorInterface" for mapping the motor control

18.1.1.4 What is a port block?

A port block is a signal interface to a fieldbus. Input/output ports represent the input and output process data of the fieldbus.

- Port blocks LP_CanIn/LP_CanOut: Signal interface to the CAN bus
- Port blocks LP_MciIn/LP_MciOut: Signal interface to a plugged fieldbus module

If, for instance, the inverter is to be controlled via CAN bus or a fieldbus module, the input/output ports are connected to the application block (device-internal signal processing) in the I/O level of the FB Editor.



❶ Name of the port block
❷ Button for opening the online help for the port block
❸ Button for opening the parameterisation dialog or the parameter list for the port block

[18-4] Example: Input port "LP_CanIn1" and output port "LP_CanOut1"

18.1.1.5 What is an application block?

The application/technology function set in C00005 is shown as application block in the I/O level of the FB Editor.

The application block comprises the signal flow processing generated via function block interconnection for the selected application in each case (e.g. "actuating drive speed" or "switch-off positioning"). The function block interconnection is shown in detail on the application level.

Tip!

Every application block features so-called "free inputs and outputs" which you can use to transfer signals from the I/O level to the application level and vice versa.

- In the Lenze setting, these connectors are hidden in the function block editor.
- These connections can be shown via the **Connector visibilities** command in the *Context menu* of the application block.

18.1.2 Conventions used for input/output identifiers

This chapter describes the conventions used for the identifiers of the inputs/outputs of the blocks. The conventions ensure a uniform and consistent terminology and make reading and comprehending the interconnection and application easier.

Tip!

The conventions used by Lenze are based on the "Hungarian Notation". This ensures that the most significant characteristics of the corresponding input/output (e.g. the data type) can be instantly recognised from its identifier.

An identifier consists of

- a data type entry
- an identifier (the "proper" name of the input/output)
- an (optional) signal type specification

Data type entry

The data type entry provides information about the data type of the corresponding input/output:

Data type entry	Meaning	Resolution	Value range
b	BOOL	1 bit	0 ≡ FALSE / 1 ≡ TRUE
dn	DINT	32 bits	-2147483647 ... 2147483647
n	INT	16 bits	-32767 ... 32767
w	WORD	16 bits	0 ... 65535

Identifier

The identifier is the proper name of the input/output and should indicate the application or function.

- Identifiers always start with a capital letter.
- If an identifier consists of several "words", then each "word" must start with a capital letter.
- All other letters are written in lower case.

Signal type entry

In general, it is possible to assign a certain signal type to the inputs and outputs of the Lenze function blocks. There are e.g. digital, scaled, position, acceleration and speed signals.

- A corresponding ending (preceded by an underscore) is added to the identifier of the corresponding input/output to indicate the signal type.

Signal type entry & port symbol in the FB Editor		Meaning	Resolution	Value range
_a	○	Analog/scaled	16 bits	± 199.99 %
_v	◁/▷	Angular velocity	16 bits	± 30000.0 rpm
_p	◀/▶	Position	32 bits	-2^{31} ... 2^{31}-1 increments
	□	Digital (BOOL)	8 bits	0 ≡ FALSE; 1 ≡ TRUE
	▬	Other (WORD)	16 bits	0 ... 65535
	■	Other (DINT)	32 bits	-2147483647 ... 2147483647

18.1.3 Scaling of physical units

With regard to the parameter setting & configuration of the inverter it is very helpful to know the signal types and their scaling listed in the following table, which are used to process physical values (e.g. an angular velocity or position) in the function block interconnection.

Signal type entry & port symbol in the FB Editor		Meaning	Scaling External value	≡ internal value
_a	○	Analog/scaled	100 %	≡ 2^{14} ≡ 16384
_v	◁/▷	Angular velocity	15000 rpm	≡ 2^{14} ≡ 16384
_p	◀/▶	Position	1 encoder revolution	≡ 2^{16} increments

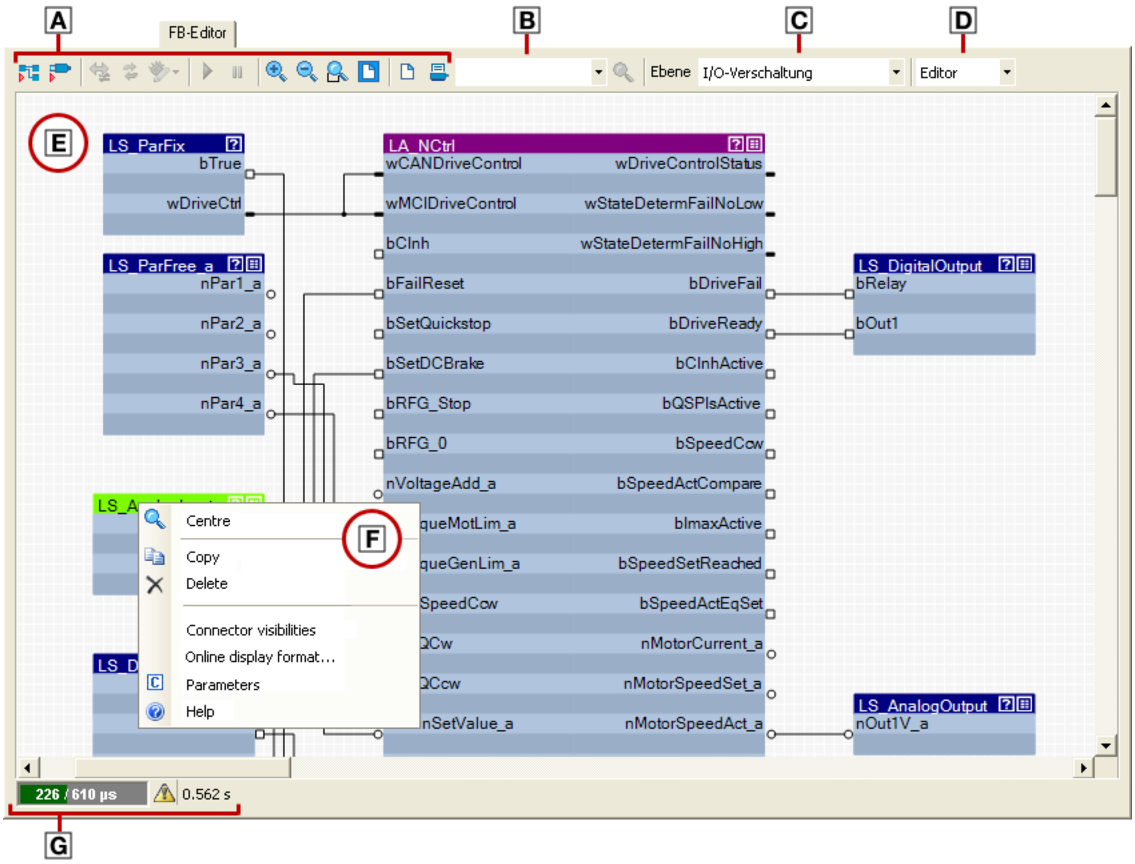
18.2 User interface

How to access the FB Editor:

1. Go to the *Project view* and select the 8400 inverter.
2. Go to *Workspace* and select the **FB Editor** tab.

   The FB Editor displays the wiring of the technology function selected in C00005. The interconnection of the I/Os of the inverter depends on the control mode selected in C00007.

The user interface of the FB Editor includes the following control and function elements:



A Toolbar	B Search function	C Plane selection	D Editor view/overview
E Drawing area	F Context menu	G Status bar	Not shown: Overview

Tip!

Go to the *»Engineer« toolbar* and click the icon to hide the *Project View* and the *Message Window*. This increases the *Workspace* available for the FB Editor. A renewed click on the symbol shows the *Project View* and the *Message Window* again.

18.2.1 Toolbar

The FB Editor is provided with an individual toolbar in the upper position which in the following text is called *FB Editor toolbar*.

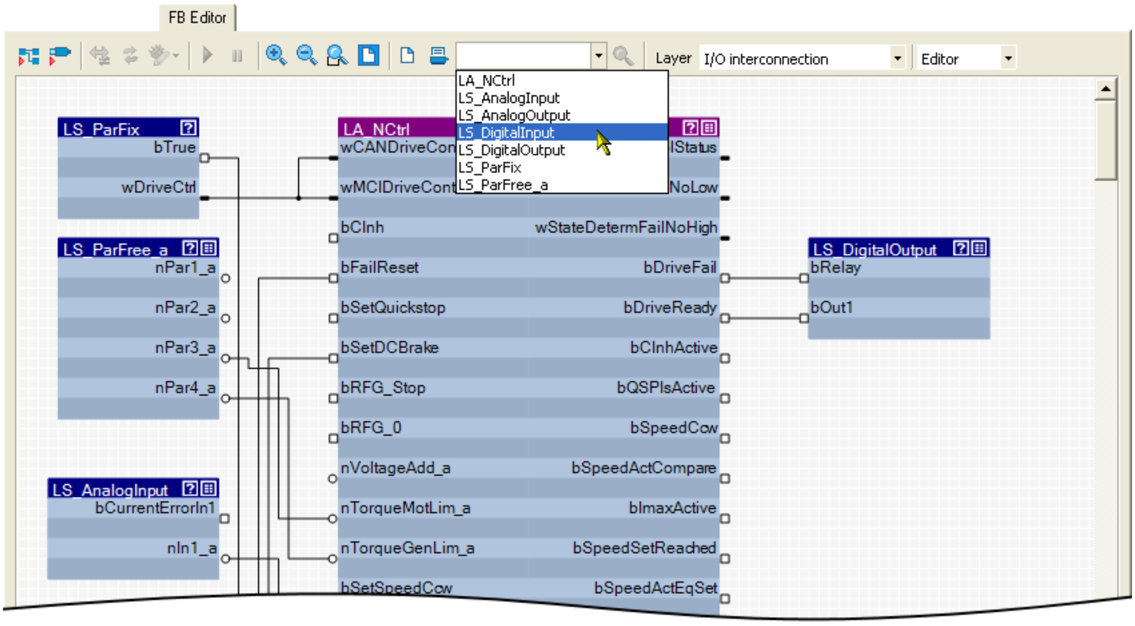
- Click on an icon to execute the corresponding function.

Symbol	Function
	Insert function block or system block ▸ Inserting a function block (🕮 1438) ▸ Inserting a system block (🕮 1440)
	Insert port block (🕮 1442)
	Adjusting online and offline interconnection (🕮 1461)
	Acknowledge error in the interconnection / reload interconnection
	Correct interconnection
	Start online monitoring
	Interrupt online monitoring
	Close online monitoring
	Enlarge view of interconnection
	Reduce view of interconnection
	Enlarge cutout of interconnection
	Show entire interconnection in the drawing area
	Show print view
	Printing the interconnection (🕮 1462)
	Search function (🕮 1425)

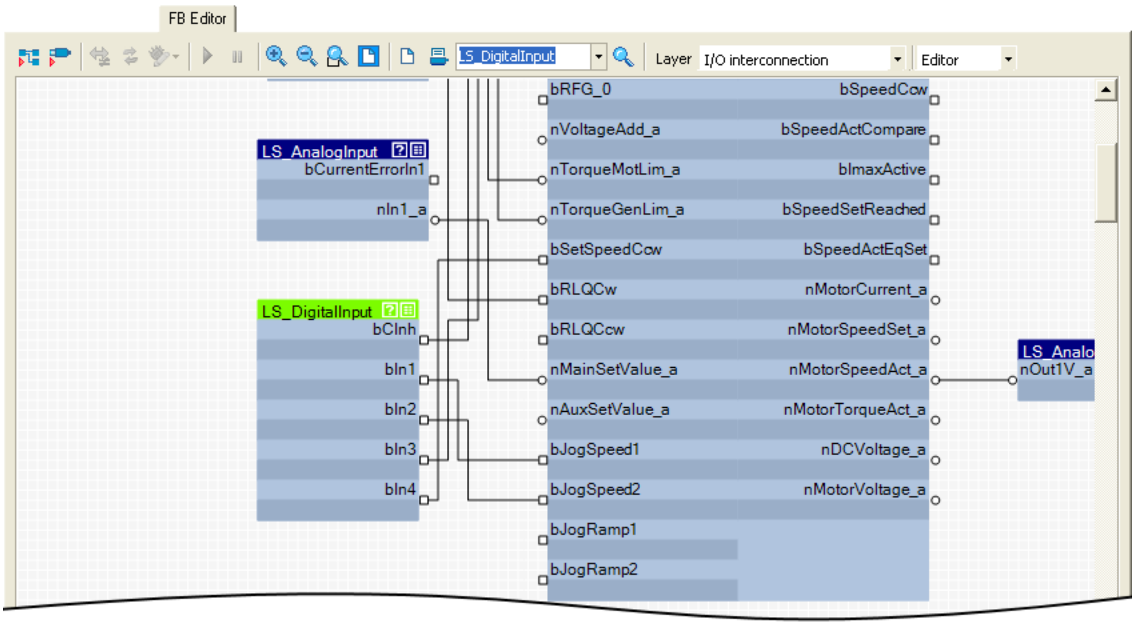
18.2.2 Search function

Use the search function to get quickly to a certain module of the interconnection.

- The list field of the search function contains all function blocks, system blocks, and port blocks of the interconnection:



- When you select a module in the list field, this module is zoomed in and selected at the same time (the following example shows the **LS_DigitalInput** system block):



Tip!

You can also enter any search text in the input field.

- If you click the icon, the cutout is moved to the object which contains this search text.
- Another click on the icon leads to a new search. Thus, you can navigate successively to all objects which contain the entered search text.
- The search text does not consider case sensitivity.

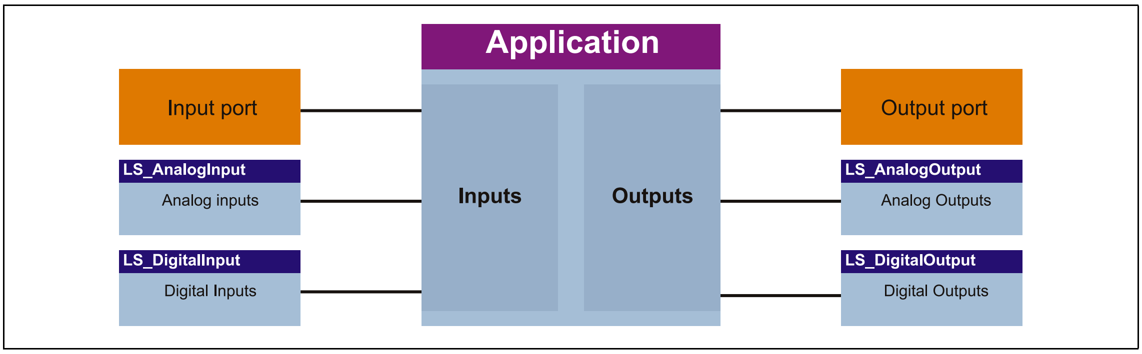
18.2.3 Plane selection

Go to the **Level selection** list field and select the interconnection level to be displayed.

"I/O interconnection" level

This level displays only the I/O interconnection of the currently selected technology application for a better overview.

- Details of the application are masked out in this level.
- The interconnection of the I/Os of the inverter with the inputs and outputs of the application in detail depends on the control mode selected in C00007.
- The parameterisation dialogs on the **Application parameter** tab correspond to the application block displayed in this level.

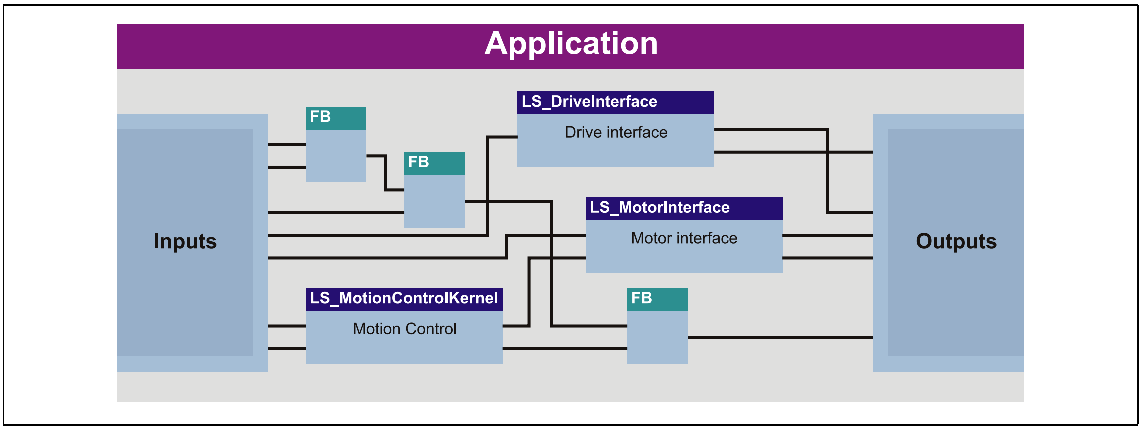


[18-5] Schematic diagram of "I/O interconnection"

"Application interconnection" level

This level displays the interconnection of the application selected in C00005 in detail. All function blocks used in the application and the system blocks which provide the interfaces to the drive and motor interface and to the MotionControlKernel (MCK) are displayed with their connections.

- The interconnection of the I/Os of the inverter with the inputs and outputs of the application is masked out in this level.



[18-6] Schematic diagram of "Application interconnection"

Tip!

Every application block features so-called "free inputs and outputs" which you can use to transfer signals from the I/O level to the application level and vice versa.

- In the Lenze setting, these connectors are hidden in the function block editor.
- These connections can be shown via the **Connector visibilities** command in the *Context menu* of the application block.

"Free interconnection" level

This level serves to implement an individual drive solution for the device versions "StateLine C" (from version 12.00.00 and »Engineer« V2.17), "HighLine C" and "TopLine C".

Note!

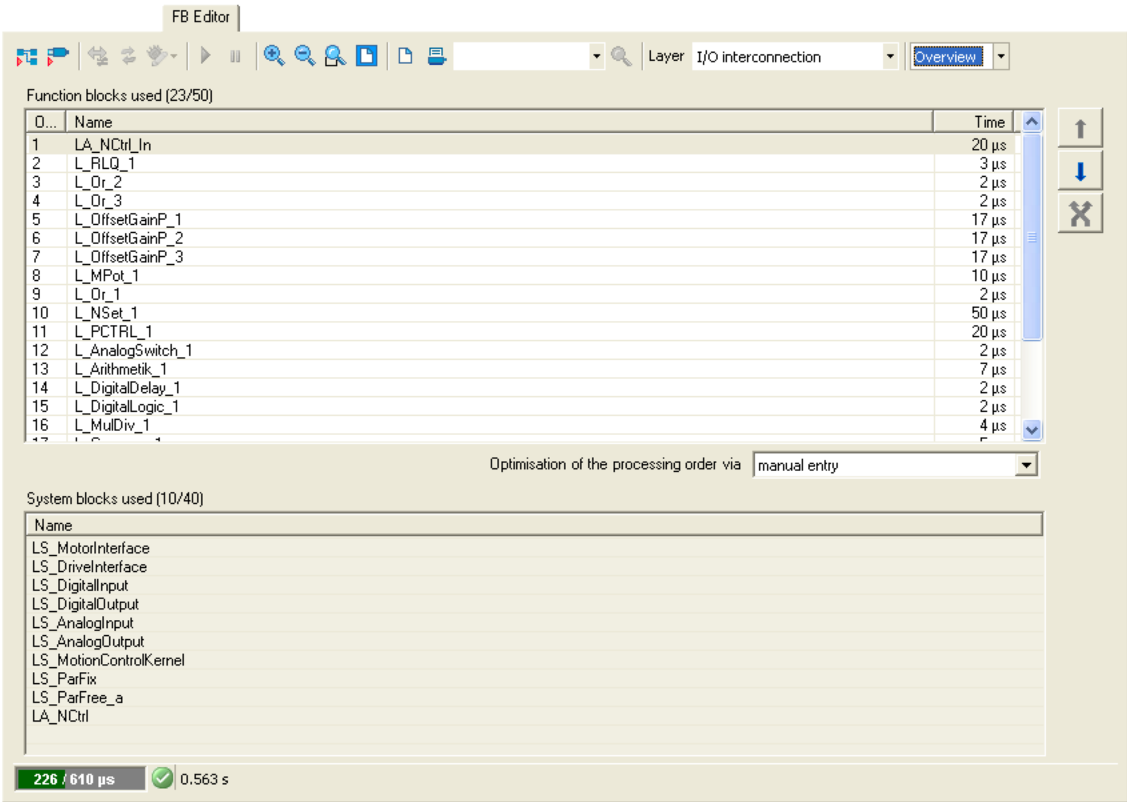
When you select the "Free interconnection" level for the first time, you are prompted to confirm whether the interconnection from the I/O level and the application level are to be combined and copied into this level.

When you confirm this confirmation prompt with **Yes**, the I/O level and the application level are not available anymore. This action can only be undone by resetting the application to a predefined Lenze application! ▶ Resetting changed interconnection (📖 1460)

18.2.4 Editor view/overview

Use the list field at the top right to change from the Editor to the overview and vice versa.

The overview shows all function blocks used of the interconnection in the upper list field in the order of their processing. The lower list field shows all used system blocks.

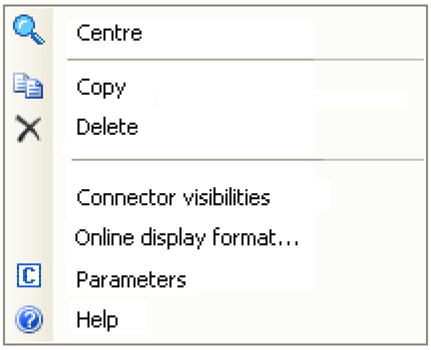


- The processing order of the function blocks can be optimised manually or according to an automatically generated selection. ▶ Changing the processing order (📖 1455)

18.2.5 Context menu

You can open a *context menu* via the right mouse button for each object (function block, system block, line, comment, etc.) and for the drawing area:

- The contents of the *context menu* depend on the type of object you click on.
- Example: *Context menu* for a function block:



18.2.6 Status bar

The status bar of the FB Editor shows, among other things, information about the system load and the error status of the interconnection:

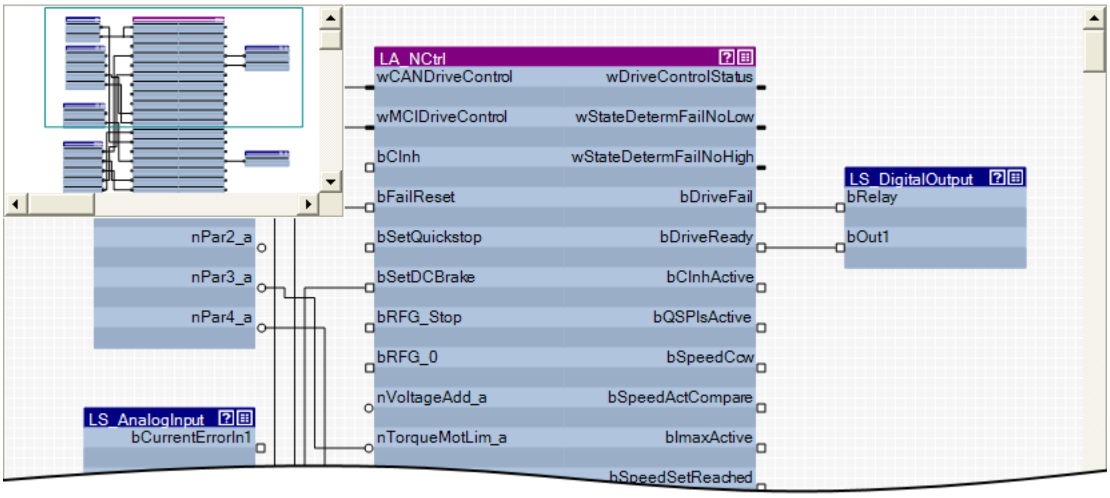
Symbol	Meaning
A System load	
	Here: out of the available computing time of 610 µs, 226 µs are required by the application.
B Error status of the interconnection	
	The interconnection has no errors and no warnings
	The interconnection has errors and/or warnings
C Communication status	
	Offline
	Online
	Communication error
D Adjustment status	
	Offline and online interconnection match
	Offline and online interconnection are different
E Update rate for monitoring values	

18.2.7 Overview

The overview window shows the drawing area in a reduced view. The overview window serves to e.g. move quickly through a more complex interconnection.

How to show the monitor window:

Go to the *Context Menu* of the drawing area and select the **Overview Window**.
- If you execute this command again, the overview window is hidden again.



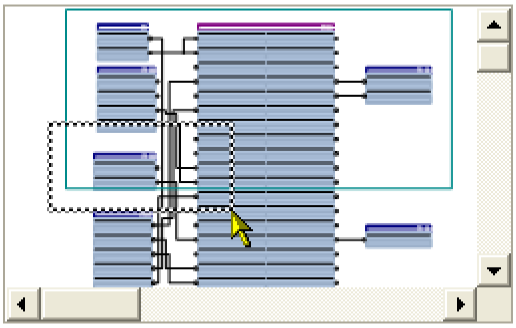
- The green frame in the overview window indicates the interconnection cutout that is currently displayed in the drawing area.
- Use the mouse pointer to shift and resize the cutout to be displayed.

How to shift the cutout presented in the drawing area:

1. Position the mouse pointer to the green frame in the overview window.
   - The mouse pointer symbol becomes a positioning cross.
2. Click left mouse button and shift the green frame to its new position by keeping the mouse button pressed, so that the desired cutout of the interconnection is displayed in the drawing area.

How to redefine the cutout to be presented:

In the overview window draw a frame around the area of the interconnection which is to be presented in the drawing window by keeping the left mouse button pressed:



- The aspect ratio of the frame is automatically adapted to the aspect ratio of the drawing area.
- According to the size of the frame that is drawn, also the presentation size of the objects in the drawing area changes.

Tip!

Go to the *FB Editor toolbar* and click the icon to adapt the view size so that all objects included in the interconnection are visible in the drawing area.

Automatic scroll ("AutoScroll function")

If you reach a window limitation in the drawing area when shifting an object or in the overview window when shifting the green frame, and if you then shortly hold the mouse pointer in this position, an automatic scrolling into the corresponding direction is carried out:

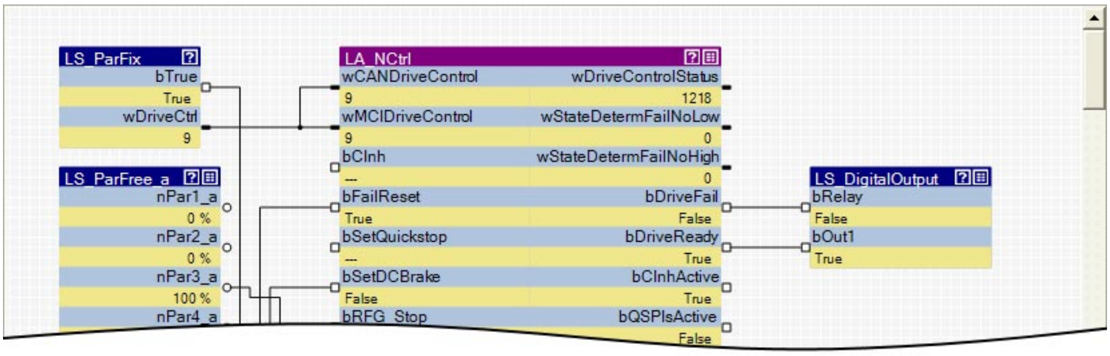
18.3 Using the FB Editor as "Viewer"

The main purpose of the FB Editor is the individual configuration of the selected technology application. However, you can also use the FB Editor to

- make a diagnosis of the application (when an online connection has been established),
- get a better understanding for the operating mode of the application,
- use the interconnection as an alternative parameterisation access.

Diagnostics of the application

When an online connection to the inverter has been established, the current values are displayed at the inputs and outputs of the objects:



- Process-scaled signals can be scaled in a "user-defined" way for easy diagnostics in the FB Editor.
  ▸ Change online display format (📖 1435)

Getting a better understanding for the operating mode of the application

Make yourself familiar with the signal flow of the interconnection to get a better understanding of the operating mode of the application or individual functional areas.

- The [?] symbol in the head of the block or the **Help** command in the *context menu* for the block serve to open the online help for the block.

Using the interconnection as an alternative parameterisation access

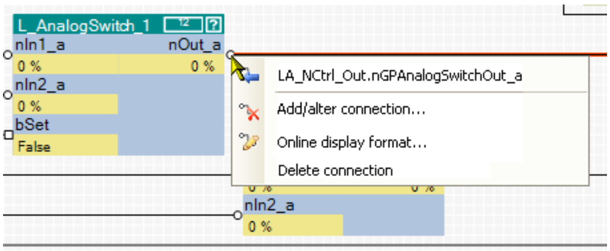
- The [▤] icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

18.3.1 Following connections of inputs and outputs

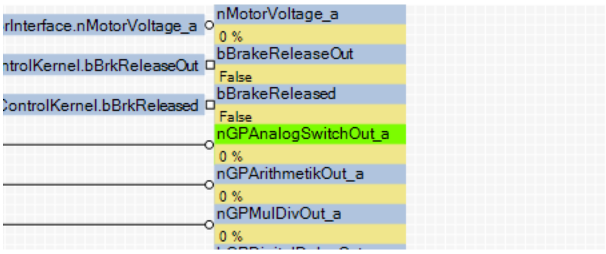
In addition to the Search function you can use the *context menu* of inputs and outputs to follow connections and quickly reach certain signals.

How to navigate from one output to another connected input:

1. Open the *context menu* (right mouse button) of the port symbol at the output.
   - The *context menu* for the port symbol contains all inputs which are connected to the output:



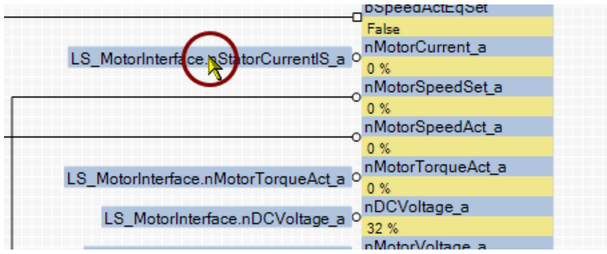
2. Select input in the *context menu* to which you want to navigate.
   - As a result, the selected input is displayed in the centre of the drawing area (in this example: nGPAnalogSwitchOut_a):



How to navigate from one input to another connected output:

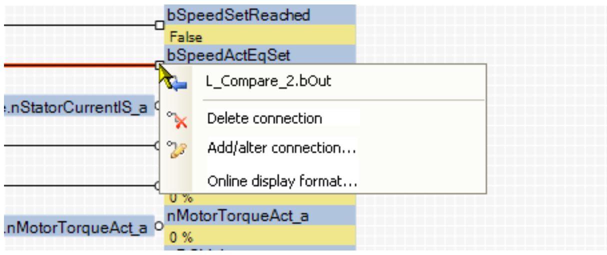
If the input is connected to a flag:

- Double-click the flag:



If the input is connected to a line:

1. Open the *context menu* (right mouse button) of the port symbol at the output:



2. Select output in the *context menu*.
   - Since an output can only be connected to an input, the *context menu* contains only an output.

The output is displayed in the centre of the drawing area.

18.3.2 Keyboard commands for navigation

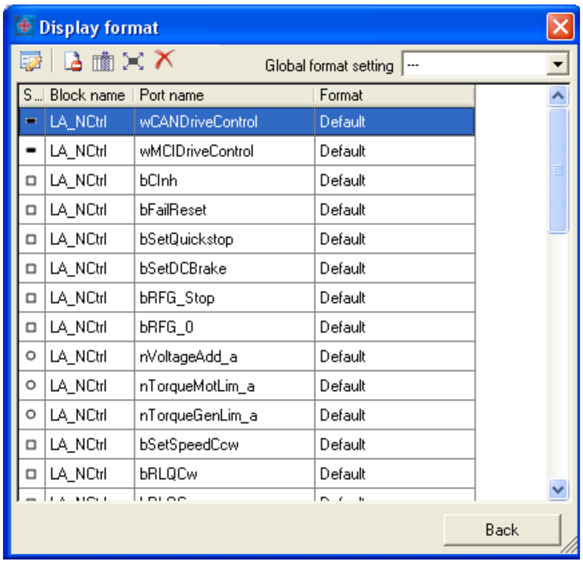
Keyboard command	Function
<Picture ▲ >	Scroll up
<Picture ▼ >	Scroll down
<Shift> + <picture ▲ >	Scroll to the left
<Shift> + <picture ▼ >	Scroll to the right
<POS1>	Scroll to the left edge of the interconnection
<END>	Scroll to the right edge of the interconnection
<Ctrl> + <Pos1>	Scroll to the left upper corner of the interconnection
<Ctrl> + <End>	Scroll to the right lower corner of the interconnection

18.3.3 Change online display format

For online monitoring in the FB Editor the display format of the input and output data of a block can be adapted individually. Process-scaled signals can be scaled in a "user-defined" way for easy diagnostics in the FB Editor. Thus, the display of these signals gets a process reference.

How to change the data display format of block inputs/outputs:

1. Go to the *context menu* of the block and select the **Online display format** command.
   - **Tip**: You can call the *context menu* of a block by clicking with the right mouse button on the header of the block.
   - The *Display format* dialog box is displayed:



2. Select the inputs/outputs from the list the display format of which is to be changed.
   - **Note**: In the **Global format setting** list field the "---" entry must be selected so that the display format can be changed.
   - If you click further inputs/outputs while pressing **<Ctrl>** they are added to an already existing selection (multi-selection).
   - The **<Shift>** key serves to select a related area of inputs/outputs.
   - More functions:

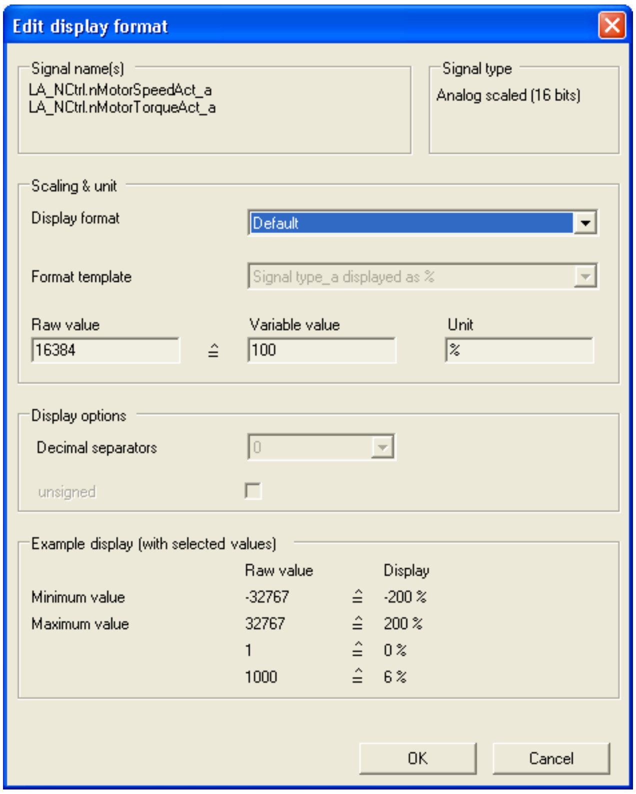
     Display masked out connections

     Display additional information

     Select all inputs/outputs

     Reset all format information

3. Click the  symbol to edit the display format of the selected inputs/outputs.
   - The *Edit display format* dialog box is displayed:



4. Go to the **Display format** list field and select the "User-defined" entry.
5. Go to the **Format template** list field and select "No template".
6. Select the required scaling, unit, number of decimal positions, and sign handling.
7. Click **OK** to accept the settings and close the *Edit display format* dialog box.
   - The *Display format* dialog box now displays the text "User-defined" for the changed inputs/outputs in the **Format** column.

   After all required formats have been changed:
8. Click **Back** to close the *Display format* dialog box.
   - For online monitoring, the changed format is used.

18.4 Reconfiguring the predefined interconnection

How to proceed:

1. Insert additionally required objects into the interconnection.
2. Hide unneeded inputs/outputs of function blocks and system blocks to obtain a clearly arranged interconnection.
3. Arrange the objects in the drawing area in a reasonable manner.
4. Establish the connections required for the desired function.
5. If required, change (optimise) the processing order of the function blocks.

Tip!

Detailed information on the individual steps can be obtained from the following subchapters!

Note!

With the "StateLine" version, the interconnection shown in the application level cannot be edited.

18.4.1 Inserting/Deleting objects

Objects can be inserted in the interconnection via the *FB Editor toolbar* and the *context menu* of the drawing area. The following subchapters provide detailed information on how to insert/delete the different objects.

Symbol	Function
	Inserting a function block (📖 1438)
	Inserting a system block (📖 1440)
	Insert port block (📖 1442)
	Inserting a comment (📖 1444)

Tip!

Use the *context menu* of the drawing area to insert a function block, system block, port block or comment directly to the current position of the mouse pointer in the drawing area.

If you insert an object via the corresponding icon in the *FB Editor toolbar*, the object is always placed at the top left corner in the drawing area.

Interconnection elements cannot only be copied within the same interconnection but also across all devices within the same project, as long as the devices stem from the same product family. ▶ Copying interconnection elements (across all devices) (📖 1457)

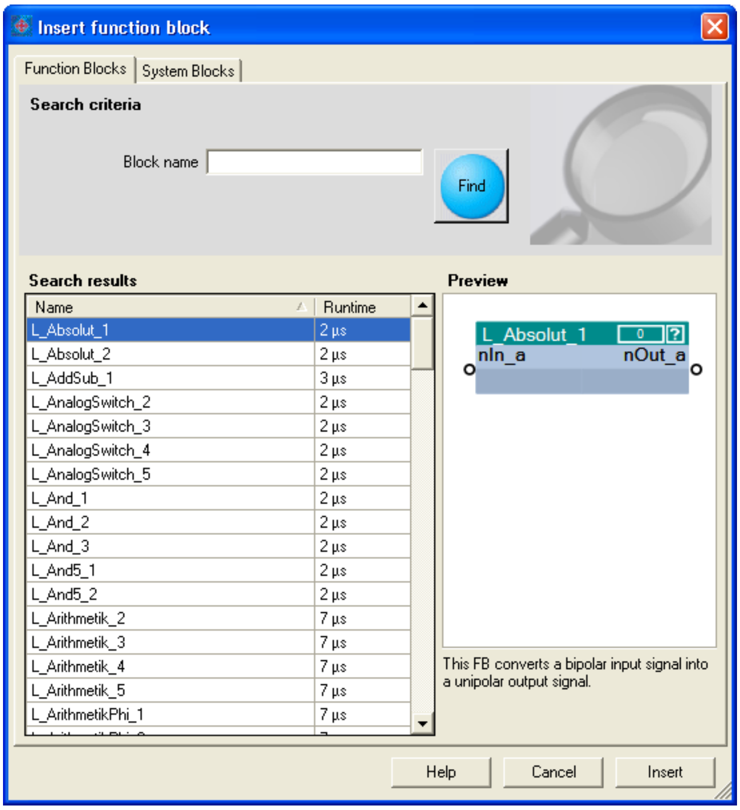
18.4.1.1 Inserting a function block

Note!

In the FB Editor, function blocks are only available in the "Application interconnection" level!

How to insert a function block into the interconnection:

1. In the *FB Editor toolbar*, click the icon.
   - The *Insert Function Block* dialog box appears:
2. Unless it is already displayed, select the **Function Blocks** tab.
   - All function blocks available are displayed in the **Search results** list field.



   - A preview of the selected function block is displayed.
   - A detailed description of all available function blocks can be found in the main chapter "Function library". (📖 1468)
3. If required, define **Search criteria** to narrow down the available function blocks:
   - **Block name**:
     String which must be contained in the name of the function block.
4. After changing the search criteria, press the **Find** button to update the selection.
   - Then, only the function blocks complying with the features set in the search criteria are shown in the **Search Results** list field.
   - If no search criteria are set, all function blocks available are shown.
5. Select the function block to be inserted in the **Search results** list field.

6. Press **Insert** button.
   - The dialog box is closed and the selected function block is inserted into the interconnection.

Context menu for the function block

If you right-click on the header of a function block, a *context menu* opens via which you can execute the following functions in addition to the general processing functions (Copy, Insert, Delete):

Command	Function
Centre	Move the visible cutout of the drawing area so that the block is centred.
Connector visibilities...	Define visible inputs and outputs of the block. Changing connector visibilities (1447)
Online display format...	Adapt the display format of the input and output data of the block individually for online monitoring. Change online display format (1435)
Parameters...	Open the parameter list/parameterisation dialog for the block. • Only if function block is parameterisable.
Help	Show online help for the block.

Related topics

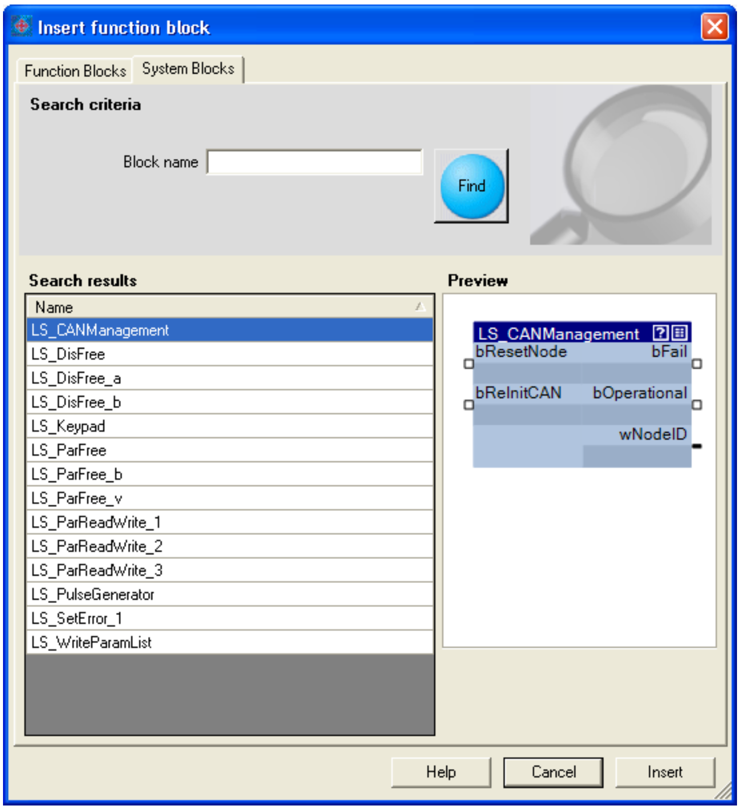
- Deleting objects that are no longer required (1446)
- Changing connector visibilities (1447)
- Arranging objects in the drawing area (1448)
- Creating/deleting connections (1449)
- Changing the processing order (1455)

18.4.1.2 Inserting a system block

A system block is inserted similarly to the way a function block is inserted.

How to insert a system block into the interconnection:

1. In the *FB Editor toolbar*, click the icon.
   - The *Insert Function Block* dialog box appears:
2. Unless it is already displayed, select the **System Blocks** tab.
   - All system blocks available are displayed in the **Search results** list field.



   - A preview of the selected function block is displayed.
3. If required, define **Search criteria** to accordingly narrow down the system blocks available:
   - **Block name**:
     String which must be contained in the name of the system block.
4. After changing the search criteria, press the **Find** button to update the selection.
   - Then, only the system blocks complying with the features set in the search criteria are shown in the **Search Results** list field.
   - If no search criteria are set, all system blocks available are shown.

5. Select the system block to be inserted in the **Search results** list field.
6. Press **Insert** button.
   • The dialog box is closed and the selected system block is inserted into the interconnection.

Context menu for the system block

If you right-click on the header of a system block, a *context menu* opens via which you can execute the following functions in addition to the general processing functions (Copy, Insert, Delete):

Command		Function
	Centre	Move the visible cutout of the drawing area so that the block is centred.
	Connector visibilities...	Define visible inputs and outputs of the block. ▸ Changing connector visibilities (📖 1447)
	Online display format...	Adapt the display format of the input and output data of the block individually for online monitoring. ▸ Change online display format (📖 1435)
	Parameters...	Open the parameter list/parameterisation dialog for the block.
	Help	Show online help for the block.

Related topics

▸ Deleting objects that are no longer required (📖 1446)
▸ Changing connector visibilities (📖 1447)
▸ Arranging objects in the drawing area (📖 1448)
▸ Creating/deleting connections (📖 1449)

18.4.1.3 Insert port block

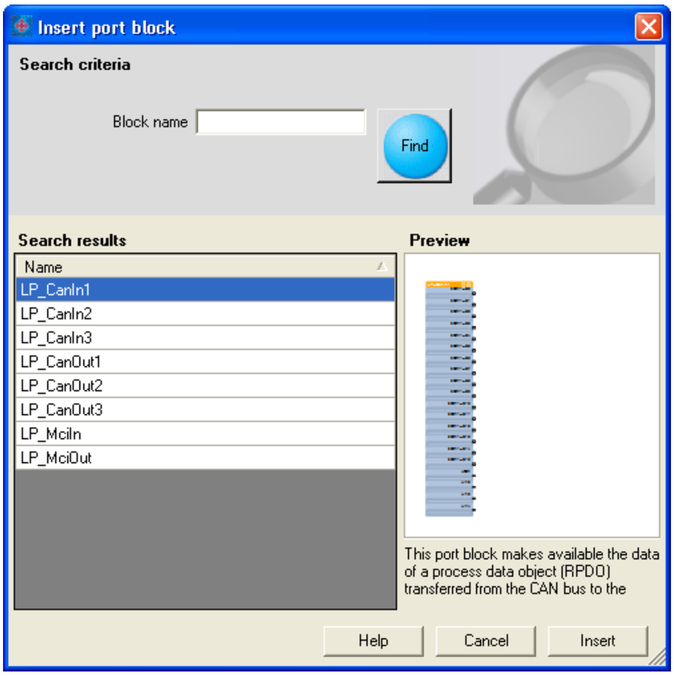
All input/output ports defined for the application on the **Ports** tab can be inserted into the interconnection in the form of port blocks in order to get access to the associated element variables.

Tip!

You can change between the **Ports** and **FB Editor** tabs at any time to define new ports and afterwards insert them into the interconnection.

How to insert a port block into the interconnection:

1. In the *FB Editor toolbar*, click the icon.
   - The *Insert port block* dialog box appears.
   - All port blocks available are displayed in the **Search results** list field.



   - A preview of the selected port block is displayed.
2. If required, define **search criteria** to accordingly narrow down the port blocks available:
   - **Block name**:
     String which must be contained in the name of the port block.
3. After changing the search criteria, press the **Find** button to update the selection.
   - Then, only the port blocks complying with the features set in the search criteria are shown in the **Search Results** list field.
   - If no search criteria are set, all port blocks available are shown.
4. Select the port block to be inserted in the **Search results** list field.
5. Press **Insert** button.
   - The dialog box is closed and the selected port block is inserted into the interconnection.

Context menu for the port block

If you right-click on the header of a port block, a *context menu* opens via which you can execute the following functions in addition to the general processing functions (Copy, Insert, Delete):

Command		Function
	Centre	Move the visible cutout of the drawing area so that the block is centred.
	Connector visibilities...	Define visible inputs and outputs of the block. Changing connector visibilities (1447)
	Online display format...	Adapt the display format of the input and output data of the block individually for online monitoring. Change online display format (1435)
	Parameters...	Open the parameter list/parameterisation dialog for the block.
	Help	Show online help for the block.

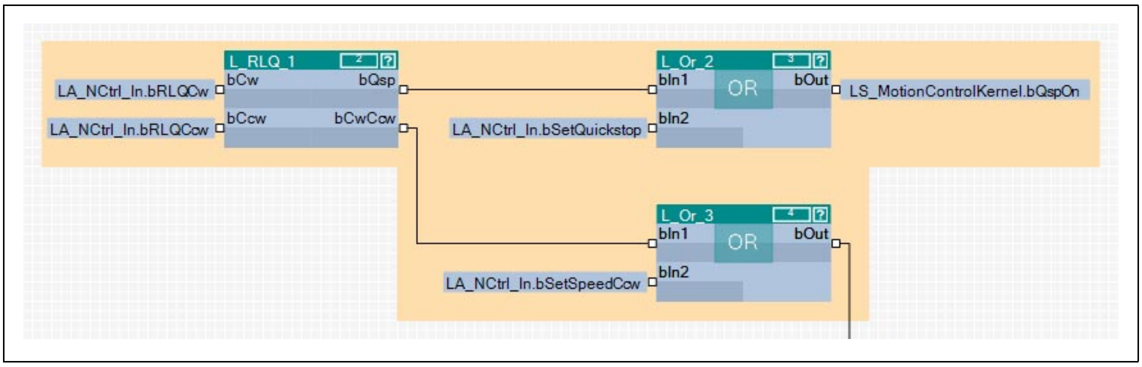
Related topics

- Deleting objects that are no longer required (1446)
- Changing connector visibilities (1447)
- Arranging objects in the drawing area (1448)
- Creating/deleting connections (1449)

18.4.1.4 Inserting a comment

Comments can be inserted at any position in the drawing area.

As of the »Engineer« V2.10, the interior colour and text alignment of a comment can be changed via a properties dialog. Now the sizes of comments can also be changed using the mouse pointer. When using different interior colours you can use comments to graphically arrange areas that belong together in terms of function or separate them from other areas:



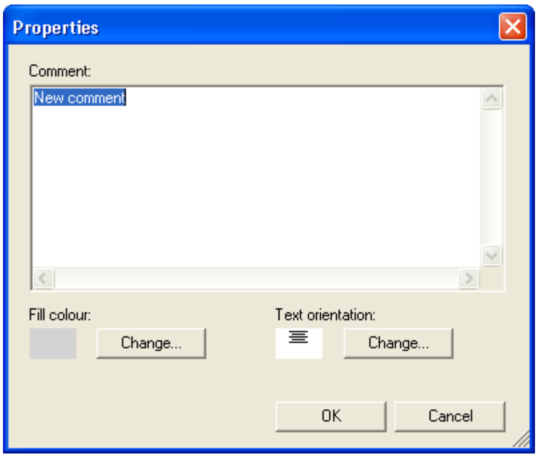
[18-7] Example: Graphical arrangement of FBs by means of two comments that overlap.

Note!

The term "Arrangement" does not mean a logical arrangement of the function blocks. The comments are only graphical presentation elements of the FB Editor.

How to insert a new comment into the interconnection:

1. Move the mouse pointer to the (free) position in the drawing area where the comment is to be inserted.
2. Go to the *Context menu* (right mouse key) and select the **New comment** command.
   - The *Properties* dialog box is displayed:



3. Enter the required comment into the text field.
4. Optional: Change preset interior colour.
   - For this purpose, click the left **Change...** button to open the *Colour* dialog box to select another interior colour.

5. Optional: Change preset text alignment.
   - For this purpose, click the right **Change...** button to open the *Text alignment* dialog box to select another text alignment.
6. Press **OK** to close the *Properties* dialog box and insert the comment.
   - After being inserted, the corner points of the comment are shown:

7. Optional: Change size of the comment.
   - For this purpose click one of the corner points with the left mouse button and enlarge the comment to the required size with the mouse button pressed.

8. Optional: Drag comment.
   - For this purpose click the comment with the left mouse button and move the comment to the required position with the mouse button pressed.

Tip!

The *Properties* dialog box for a comment already available can be opened by double-clicking the comment.

Related topics

▸ Deleting objects that are no longer required (📖 1446)
▸ Arranging objects in the drawing area (📖 1448)
▸ Creating/deleting connections (📖 1449)

18.4.1.5 Deleting objects that are no longer required

Objects that are no longer required can be easily deleted again. "Delete" only means that the object is removed from the drawing area. If you have deleted an object from the drawing area, you can reinsert it any time into the interconnection.

Note!

Deleting an object cannot be undone.

Together with the object, all available connections to this object are deleted.

How to delete objects that are no longer required:

1. Select objects to be deleted.
   - You can select a single object by clicking the header of the object.
   - You can select objects that are placed together by drawing a frame around these objects while keeping the mouse button pressed.
   - If you click the header of further objects while pressing **<Ctrl>**, these will be added to an already existing selection (multi-selection).
   - All selected objects are highlighted by a light green header.
2. Press **<Del>**.

Related topics

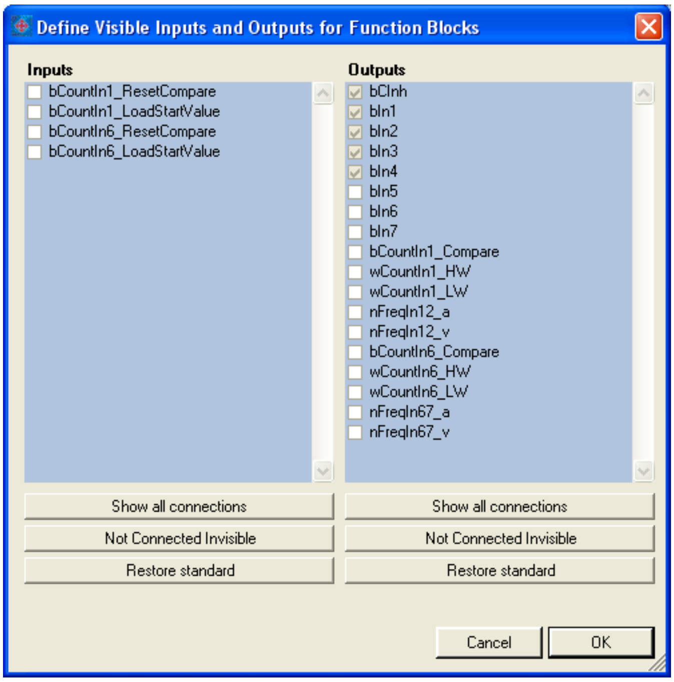
▸ Deleting connections that are no longer required (1454)

18.4.2 Changing connector visibilities

Inputs and outputs that are not connected can be hidden for each block. This serves to reduce the dimension of the block. The interconnection becomes clearer.

How to define the visible inputs and outputs:

1. Go to the context menu of the block and select the **Connector visibilities** command.
   - The *Define Visible Inputs and Outputs for Function Blocks* is displayed:



   - All visible connections have a checkmark.
   - In case of a block that is inserted anew, all inputs and outputs are visible at first.
   - Inputs and outputs with a light grey checkbox are already connected and thus cannot be hidden.
2. By setting/removing the checkmarks or via the buttons you can define the visible inputs and outputs.
3. Press **OK** to accept the selected definition and close the dialog box.

18.4.3 Arranging objects in the drawing area

All objects can be freely arranged in the drawing area by dragging with the mouse.

We recommend to make an arrangement in which the required connections between the inputs and outputs can be created easily. A division into functional areas may also be sensible to get a better understanding of the application.

Objects which are already connected, can also be dragged to another (free) position in the drawing area. The available connections will be automatically re-routed after dragging.

How to drag an object:

1. Click the header of the object (and keep the button pressed).
2. Keep the button pressed and drag the object to the required position in the drawing area.
   - Via **<Esc>** you can cancel this action.

How to drag several objects at the same time:

1. Select the objects to be dragged.
   - You can select a single object by clicking the header of the object.
   - If you click the header of further objects while pressing **<Ctrl>**, these will be added to an already existing selection (multi-selection).
   - You can easily select objects that are placed together by drawing a frame around these objects while keeping the mouse button pressed.
   - All selected objects are highlighted by a light green header.
2. Keep the mouse button pressed on the header of one of the selected objects and drag it to the required position in the drawing area.
   - Via **<Esc>** you can cancel this action.

Note!

A red header indicates that the object overlaps with other objects in the drawing area!

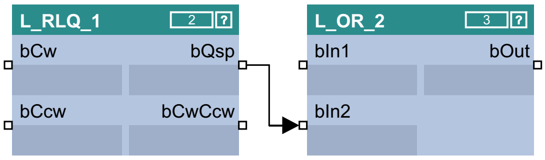
Arrange the objects so that no overlap occurs.

18.4.4 Creating/deleting connections

After adding objects and arranging them in a reasonable manner within the drawing area, you can create the connections between the available objects which are required for the desired function.

A connection always has a direction and therefore always has a source and a target.

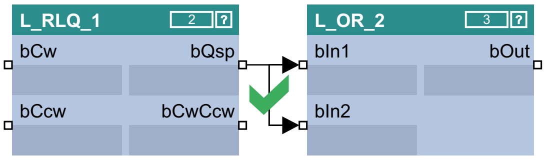
- An output represents a possible source in the interconnection.
- An input represents a possible target in the interconnection.



Permissible/impermissible connections

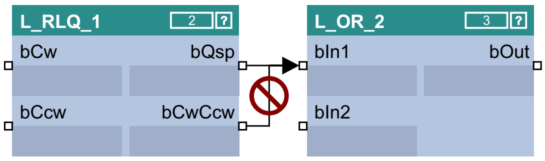
Several connections can lead from one output.

- Therefore it is always possible to start a new connection from an output.



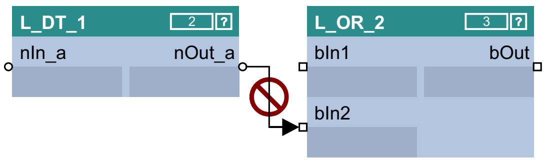
However, maximally one connection may end in an input.

- Therefore it is only possible to start a new connection from an input if there is no connection already ending in this input.



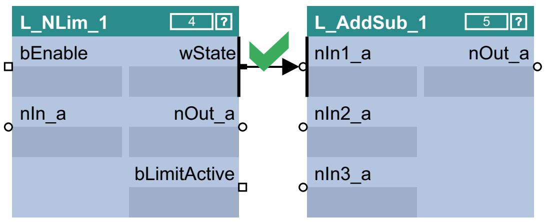
Only inputs/outputs of the same signal type can be connected.

- Thus, a connection between different port symbol cannot be established.



From the »Engineer« V2.12 "Analog/scaled" (_a) and "Miscellaneous (WORD)" signal types can also be interconnected.

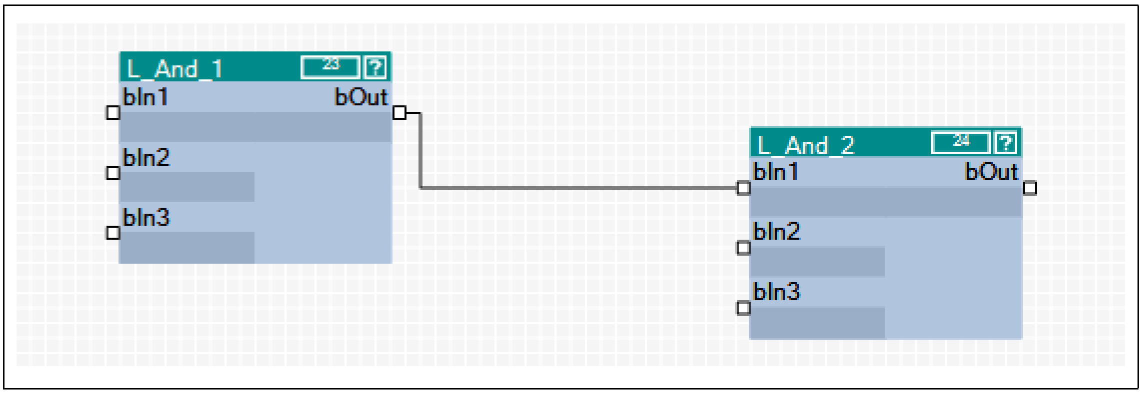
- The implicit type conversion is indicated by a vertical black bar at the port symbol.



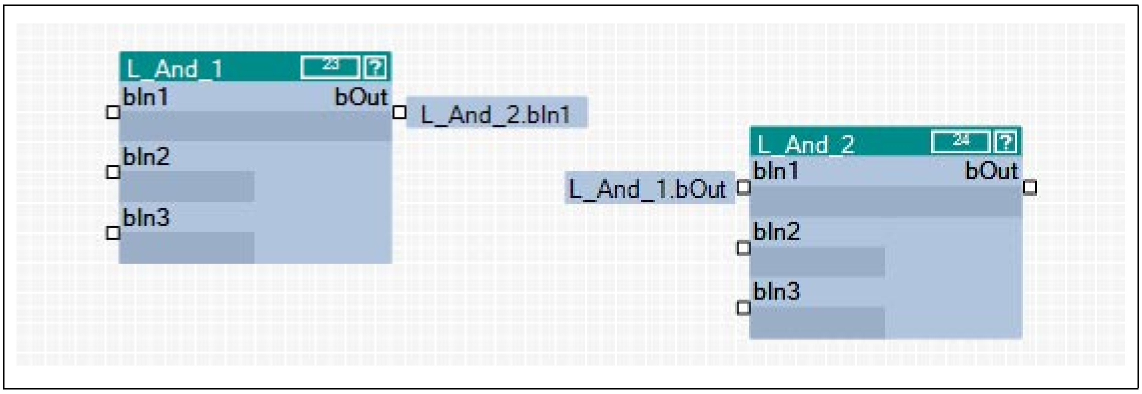
From the »Engineer« V2.13 "Analog/scaled" (_a) and "Angular velocity" (_v) signal types can also be interconnected.

Connection types

Connections can either be created by means of connection lines or port identifiers ("flags")



[18-8] Example 1: Connection via connection line



[18-9] Example 2: Connection via flags

Tip!

The commands **Show as flag** or **Show as line** in the *context menu* of a connection serve to change the representation of the connection at any time.

When an output is connected to several inputs via flags, three points are displayed ("...") at the output instead of the concrete input identifier. The *context menu* of the port symbol shows all inputs which are connected to the output.

18.4.4.1 Creating a connection using the connection line

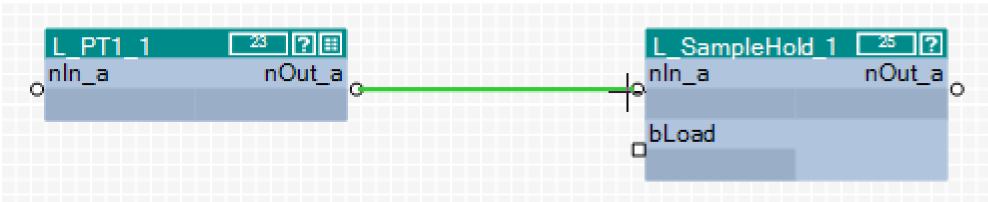
How to create a connection using the connection line:

1. Click the port symbol from which the new connection is to be started.
   - It is only possible to start a new connection from an input if there is no connection already ending in this input.
   - If you then move the mouse pointer away from the port symbol, a new connection is "drawn" from this port symbol.
   - Via **<Esc>** you can cancel this action.
2. Click the port symbol where the connection is to end.
   - Thereupon the corresponding connection is routed automatically if the connection is permissible.

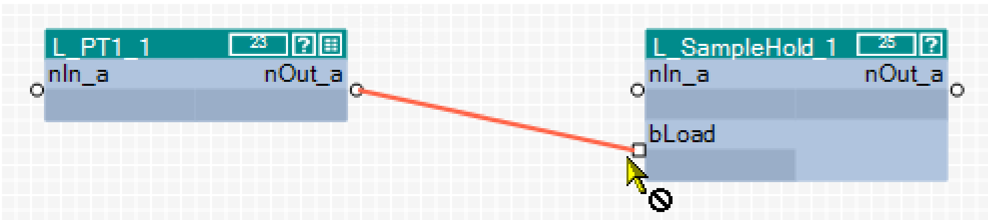
Tip!

If you move the mouse pointer across the port symbol while drawing a new connection, you can see whether the connection is permissible or not from the colour of the drawn line and from the mouse pointer symbol.

- Permissible connection:



- Impermissible connection (different port symbol):

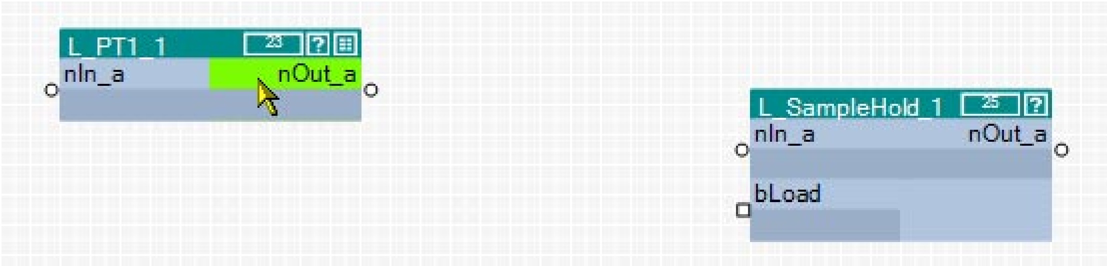


The command **Show as flag** in the *context menu* of a line serves to change the representation of the connection at any time.

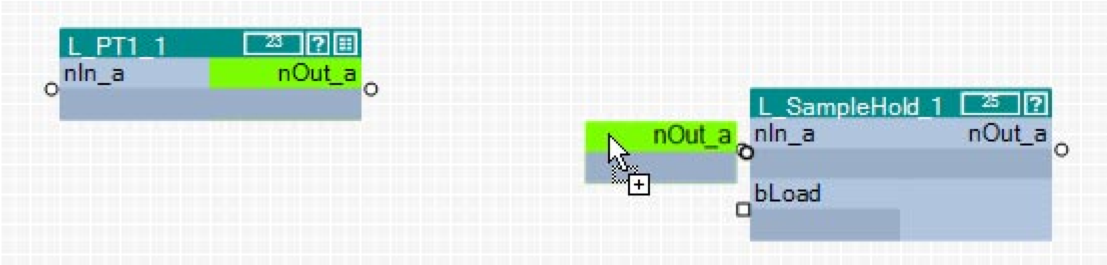
18.4.4.2 Creating a connection using port identifiers

How to create a connection with port identifiers:

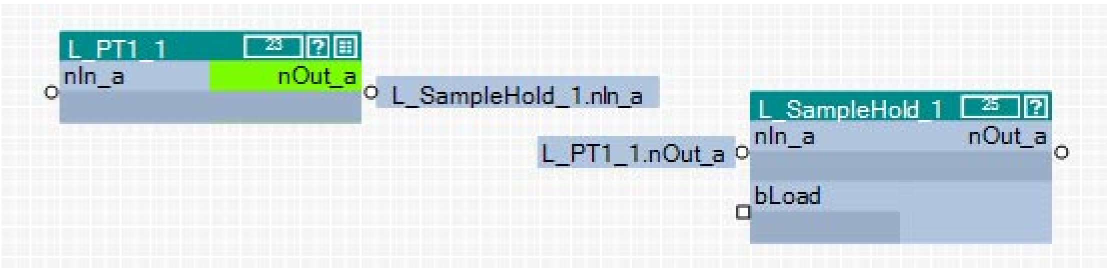
1. Click the port identifier.
   - The selected port is highlighted in light green:



2. Drag the port segment to the required port while keeping the left mouse button pressed:



After releasing the mouse button, the connection via port identifiers (flags) is created. The corresponding port identifier consists of the block name and the name of the input/output:



Tip!

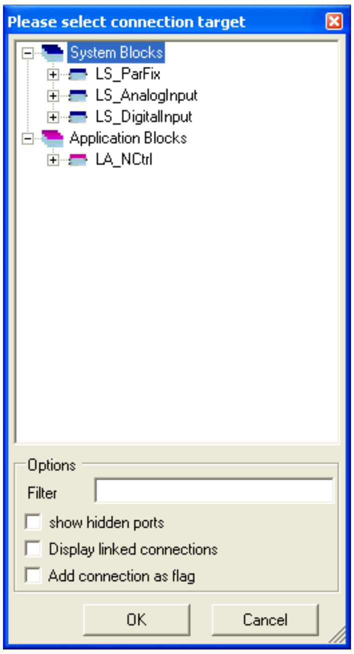
The command **Show as line** in the *context menu* of a flag serves to change the representation of the connection at any time.

18.4.4.3 Creating a connection via connection dialog

You can also create connections by means of a selection dialog instead of dragging by mouse. This especially makes sense if there is a great distance between the ports to be connected in the drawing area.

How to create a connection using the selection dialog:

1. Right-click the port identifier or click the port symbol from which the connection is to start.
   - The *context menu* for the port is displayed.
2. Go to the *context menu* for the port and select the **Add/change connection...** command.
   - The *Add/change connection* dialog box is displayed:



   - In a tree structure all inputs and outputs of the application are shown to which a connection is permissible.
   - You can enter an optional text into the **Filter** input field to reduce the selection to the blocks or ports which contain the entered text.
   - If you activate the **Show hidden ports** control field, the hidden ports for system and function blocks are shown as well.
3. Select the port where the connection is to end from the tree structure.
4. Activate the **Add connection as flag** control field if a port identifier (flag) is to be inserted instead of a connection line.
5. Press **OK** to create the connection to the selected port and close the dialog box.

18.4.4.4 Deleting connections that are no longer required

How to delete connection lines:

1. Select connection lines to be deleted.
   - Select a single connection line by directly clicking on the connection line with the right mouse button.
   - If you click further connection lines while pressing **<Ctrl>** they are added to an already existing selection (multi-selection).
   - All connection lines are highlighted in red.
2. Press **<Del>**.

How to delete port identifiers/flags:

1. Select the port identifiers to be deleted.
   - Select a single port identifier by directly clicking on the port identifier with the left mouse button.
   - If you click further port identifiers while pressing **<Ctrl>** they are added to an already existing selection (multi-selection).
   - All selected port identifiers are highlighted by a light green header.
2. Press **<Del>**.

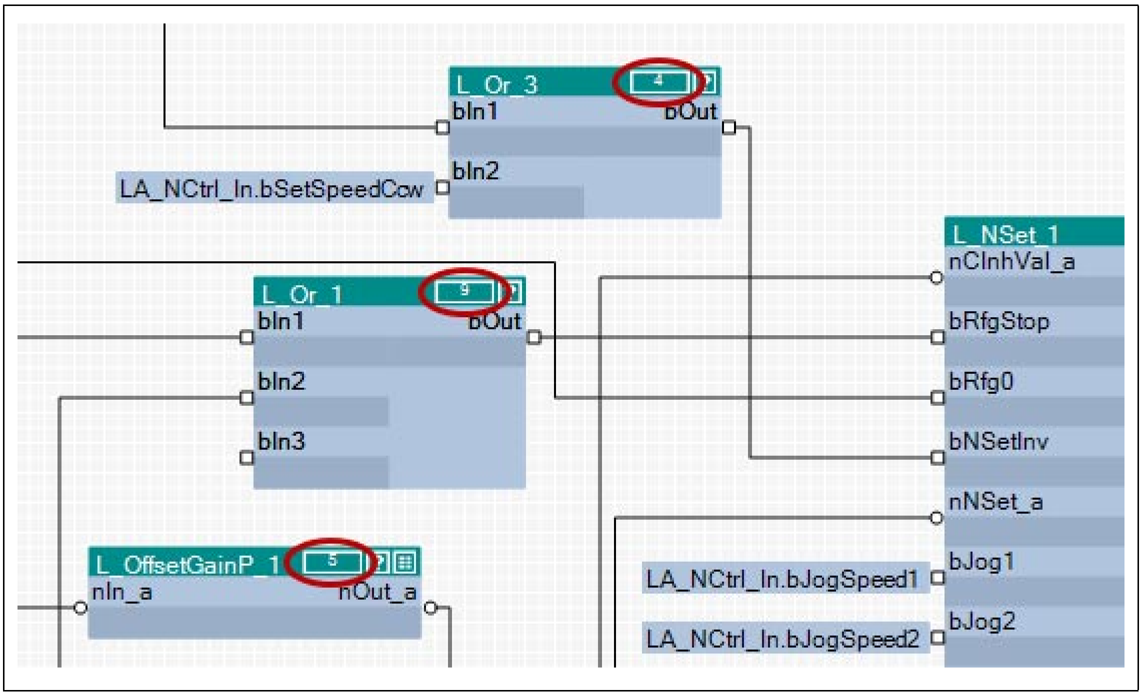
Related topics

▸ Deleting objects that are no longer required (📖 1446)

18.4.5 Changing the processing order

If you insert a function block into the interconnection, an order index is automatically assigned to this function block. By means of this order index it is defined in which order the individual function blocks are calculated at runtime.

- The first function block inserted contains the order index "1", the next function block inserted contains the order index "2", etc.
- The respective order index is displayed in the header of the function block in the rectangle after the block name.



[18-10] Example: Function blocks with order index

Note!

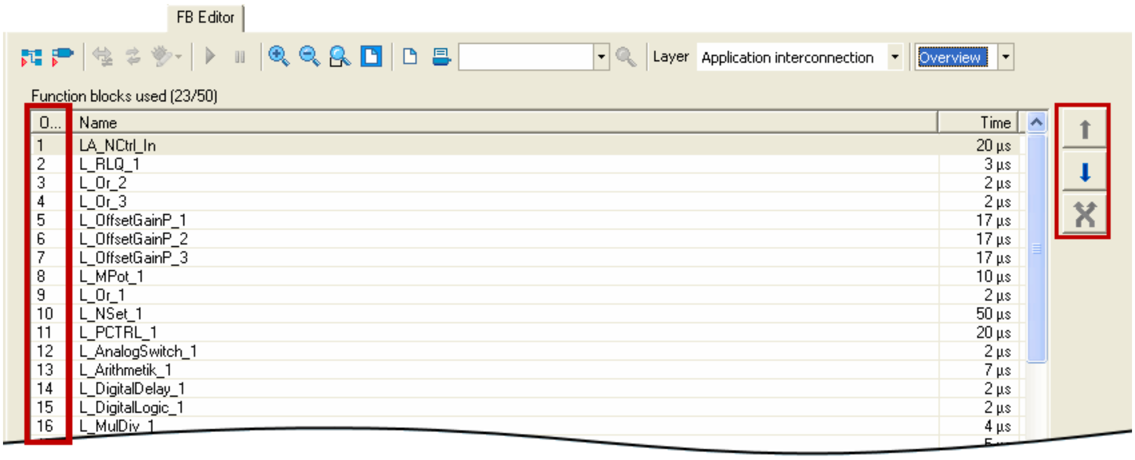
When a function block is shifted, its order index is maintained.

The processing order influences the result!

- In certain cases it may be sensible to change the processing order, but if you select an unfavourable processing order, errors may arise!

How to change the processing order manually:

1. Use the list field at the top right to change from the Editor to the overview.
   - The overview displays all function blocks of the interconnection in the order of their processing
   - In the first "Order" column the order index of each function block is listed.
2. Unless already selected, select the entry "Manual selection" in the **Optimisation...** list field.



3. Select the function block which is to receive a different position within the processing order.
   - If you click further function blocks while pressing **<Ctrl>** they are added to an already existing selection (multi-selection).
   - The **<Shift>** key serves to select a related area of function blocks.
4. Move the function block(s) to the desired position using the and buttons.
   - The button serves to exchange two selected function blocks with regard to their order.
5. Repeat steps 3 and 4 until the required processing order has been established.

Changing the processing order according to an automatically generated selection

In addition to the manual selection, the **Optimisation...** list field also offers two options for an automatic adaptation of the processing order:

- **Signal flow**: The processing order is optimised according to the signal flow.
- **Topology**: The processing order is optimised according to the x/y arrangement of the function blocks in the FB Editor.

As long as an automatic adaptation has been selected, a manual change of the processing order is not possible.

18.4.6 Copying interconnection elements (across all devices)

Interconnection elements can be copied across the devices within the project if the devices belong to the same product family (e.g. Inverter Drives 8400).

All types of blocks and comments can be copied to the clipboard via the **Copy** command or the **<Ctrl>+<c>** shortcut and then be inserted into the FB interconnection of the same or another project device of the same product family using the **Paste** command or the **<Ctrl>+<v>** shortcut.

- During the copy process into the clipboard, existing connections between copied blocks are copied as well, and the layout is kept too. Moreover, the separate technical objects (e.g port definition) are copied. Selected connections cannot be copied on their own.
- The **Paste** command is available if the clipboard is not empty and if it was copied from a device of the same product family. Within this product family, all device types (e.g. 8400 xxxxLine Vxx.xx) are permitted.
- After the **Paste** command has been selected, a dialog box is displayed which serves to select which elements are to be inserted from the clipboard and how to solve name conflicts, if any.
- After inserting the elements, they are marked in the target interconnection in order to be repositioned or deleted again to undo the insertion.
- Inserting from the clipboard can be repeated. The originally copied contents of the clipboard remains unchanged when it is inserted.

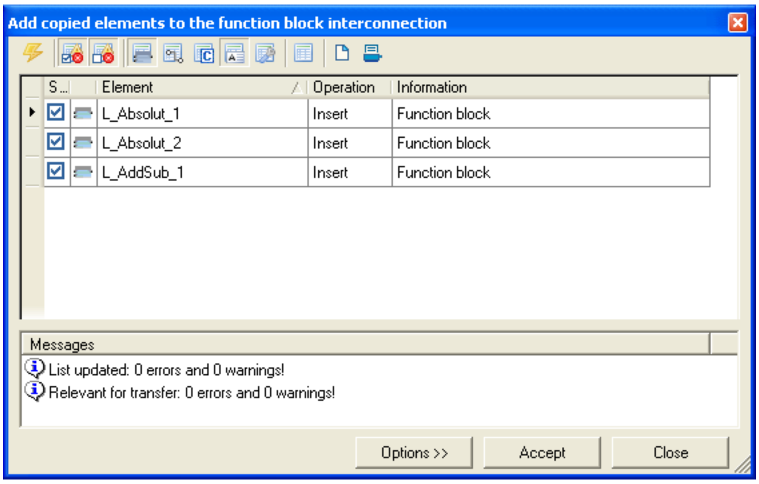
How to copy one or several interconnection elements:

1. Select the objects to be copied.
   - You can select a single object by clicking the header of the object.
   - If you click the header of further objects while pressing **<Ctrl>**, these will be added to an already existing selection (multi-selection).
   - You can easily select elements that are placed together by drawing a frame around these elements while keeping the mouse button pressed.
   - All selected objects are highlighted by a light green header.
2. Go to the *context menu* and select the **Copy** command (or **<Ctrl>+<c>**).
   - The selected elements are copied into the clipboard of the FB Editor.
3. If the elements are to be copied into a function block interconnection of another project device, change to the corresponding interconnection via the *project view*.
4. Go to the *context menu* and select the **Paste** command (or **<Ctrl>+<v>**).
5. Go to the *Insert FB interconnection* dialog box and select the elements to be inserted from the clipboard.
   - Detailed information on this dialog box can be obtained from the following subchapter "Insert options for copied elements". (📖 1459)

6. Click **Insert** to insert the selected elements into the target interconnection as defined.
   - Only possible if at least one element in the list has been selected for insertion.
   - Insertion is also possible via the **<Enter>** button if at least one element is selected from the list for insertion.
   - The original layout and the relative position of the inserted blocks to each other are maintained.
   - When copying across the devices, you also insert the corresponding separate technical objects (e.g. port definition).
   - The inserted elements are deleted from the list. If the list is empty, the dialog box is closed and the connections are inserted depending on the selected option.
7. If there are still elements to be entered in the list, repeat steps 5 and 6 until all elements are inserted as intended.
8. Press **Close** to stop the insertion and close the dialog box.
   - You can also use **<Esc>** or **<Enter>** to close the dialog box if "Insert" is not active.
   - The elements inserted into the target interconnection so far are maintained.
   - The connections for the blocks inserted so far are inserted depending on the selected option.

18.4.6.1 Insert options for copied elements

If interconnection elements have been copied to the clipboard, the »Engineer« will display a list of all elements contained in the clipboard when selecting the command **Insert** in the *Insert FB interconnection* dialog box:



The list shows the elements which can be added to the target interconnection, and the elements which cannot be added.

- In the "Selection" column, you can check/uncheck the elements to be added.
- Connections are only inserted when the dialog box is closed, which applies to all modules inserted so far. They are displayed as lines or flags, like in the original, but re-routed.
- The symbols in the *Toolbar* serve to execute the following functions:

Symbol	Function
	Add the selected elements to the interconnection
	Show the elements to be added but are marked with an error or warning.
	Show the elements not to be added and marked with an error or warning.
	Show blocks
	Show connections
	Show parameters
	Show comments
	Show system elements
	Show all
	Print Preview
	Print list

- The buttons serve to execute the following functions:

Button	Function
Paste	Add elements selected in the list to the target interconnection • Only possible if at least one element in the list has been selected for insertion. • Insertion is also possible via the **<Enter>** button if at least one element is selected from the list for insertion. • The original layout and the relative position of the inserted blocks to each other are maintained. • When copying across the devices, you also insert the corresponding separate technical objects (e.g. port definition). • The added elements are simultaneously deleted from the list. The connections are added depending on the selected option.
Close	Close dialog box. • You can also use **<Esc>** or **<Enter>** to close the dialog box if "Insert" is not active. • The elements inserted into the target interconnection so far are maintained. • The connections for the blocks inserted so far are inserted depending on the selected option.

18.4.7 Resetting changed interconnection

If you only made changes on the I/O level, you can reset them by selecting a predefined control scheme in C00007. If you have also made changes on the application level, you must first reset the changed application to a predefined application in C00005.

How to reset the application interconnection to a predefined application:

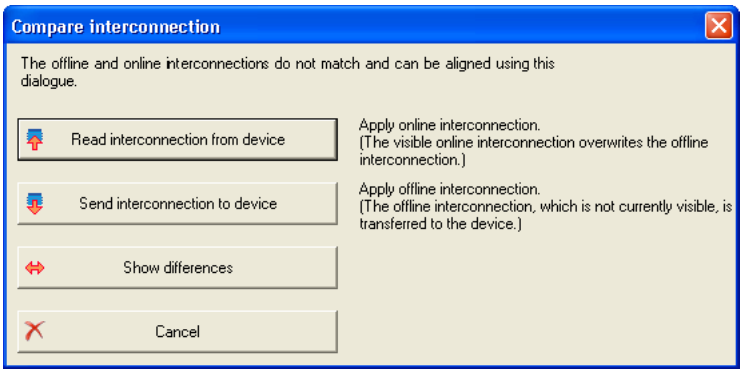
1. Go to the **Application parameters** tab.
2. Select the required application in the **Application** list field.

How to reset the I/O interconnection to a predefined control scheme:

1. Go to the **Application parameters** tab.
2. Select the required control scheme in the **Control source** list field.

18.5 Adjusting online and offline interconnection

If the »Engineer« detects that online and offline interconnection differ from each other, the *Compare interconnection* dialog box is displayed with various options for the adjustment:



Tip!

The dialog box can also be opened via the symbol in the *FB Editor toolbar*.

Button	Function
Read interconnection from device	Add the interconnection in the device to the FB Editor. The interconnection existing in the FB Editor will be overwritten by this action.
Send interconnection to device	Transfer the offline interconnection which is currently not visible in the FB Editor to the device. The interconnection existing in the device will be overwritten by this action.
Show differences	Showing differences between online and offline interconnection.
Cancel	Close the *Adjust interconnection* dialog box without making an adjustment.

18.6 Printing the interconnection

The interconnection can be printed for documentation purposes, optionally on one page, on four pages, or not scaled.

Tip!

By clicking the icon in the *FB Editor toolbar*, you can get a print view before printing.

How to print the interconnection:

1. In the *FB Editor toolbar*, click the icon.
   - The *Circuit print size* dialog box is displayed.
2. Select the desired size and press **OK**.
   - The standard dialog box *Print* appears.
3. Press **OK** to start the printing process.

18.7 Comparing interconnections

The comparison operation serves to compare FB interconnections of 8400 devices within the project. An offline<>online comparison and the comparison of two online devices are possible.

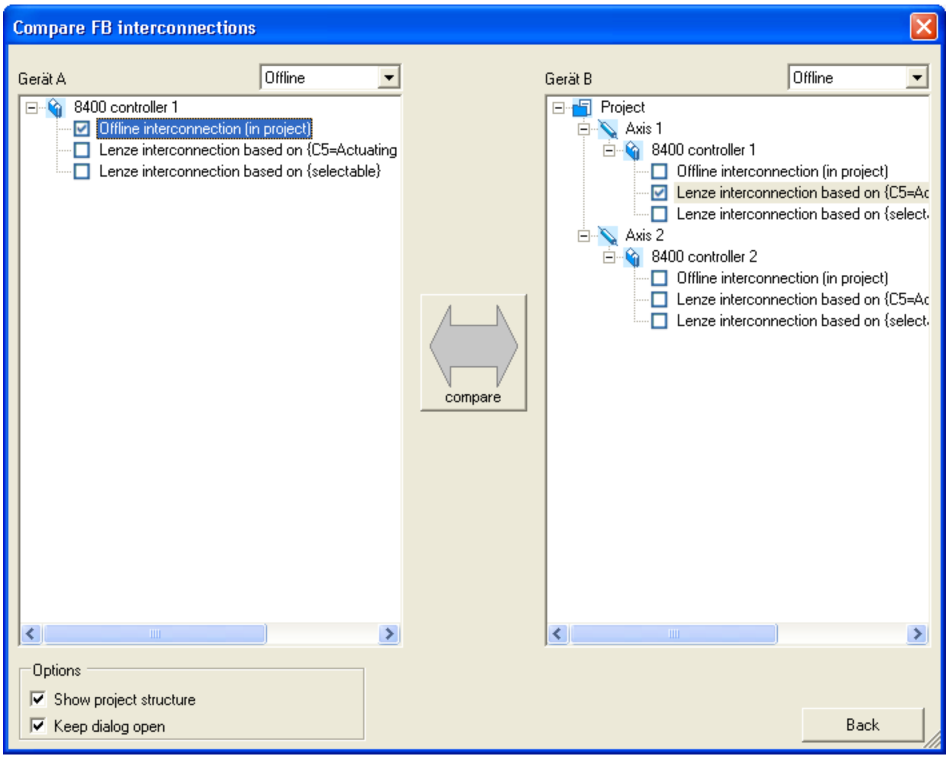
Note!

Only applications can be compared which have been enabled in the FB Editor!

Block positions, line representations, and connector visibilities are not compared.

How to compare two FB interconnections:

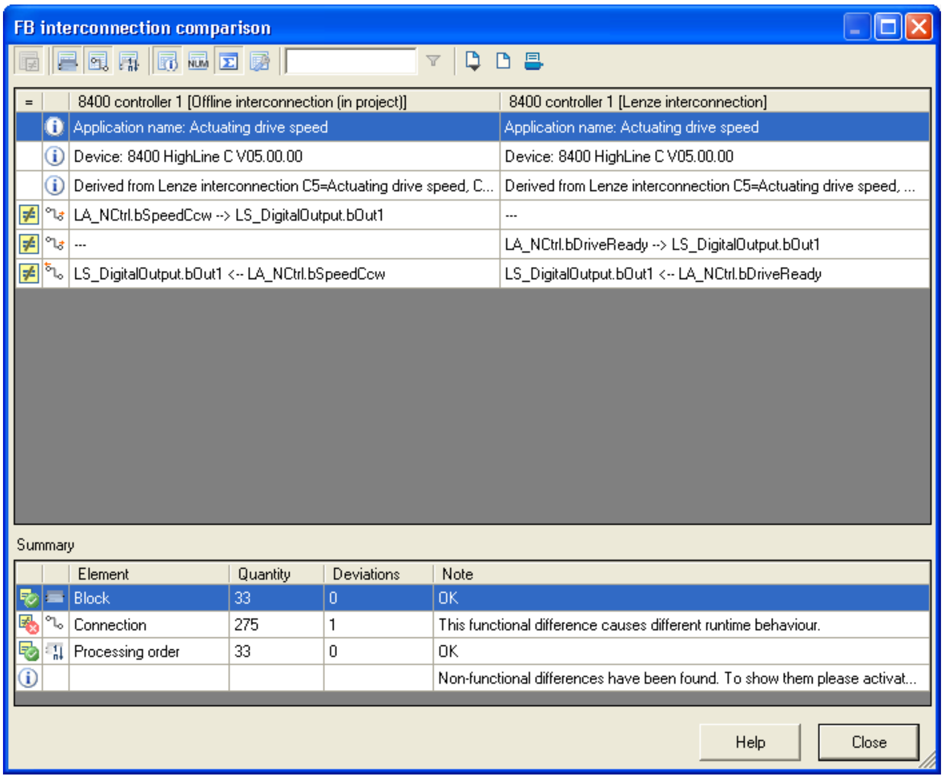
1. Select the command **Application data→Compare FB interconnections...**.
   - The *Compare FB interconnections* dialog box is displayed:



2. Select the interconnections to be compared in the project view represented on the left and right.
   - In order to execute a comparison with an online device, select "Online" in one of the two upper list fields. Then all available online devices are displayed for selection.
   - If you select "Online" in one of the two upper list fields, you can also compare the interconnections of two available online devices.
3. Click **Compare**.
   - If the comparison was executed successfully, the comparison result is displayed as a list (see the following section).
   - If a comparison of the selected interconnections is not possible, a corresponding message is displayed.
4. In order to stop the comparison operation and close the dialog box: Press **Back**.

Representation of the comparison result

The comparison result is displayed in the form of a list in the *FB interconnection comparison* dialog box:



- The symbols in the *Toolbar* serve to show or hide different details and export and print the shown list.

Symbol	Function
	Only show differences • Button can only be activated in expert mode.
	Show blocks
	Show connections
	Show processing sequence
	Show properties for blocks and connections • Function is only available in expert mode.
	Show comments • Function is only available in expert mode.
	Show block parameters • Function is only available in expert mode.
	Show application parameters • Function is only available in expert mode.
	Show all • Function is only available in expert mode.
	Show general information
	Show summary

Symbol	Function
	Activate expert mode • In the expert mode, also non-functional differences are shown.
	Use filter • Only show list entries which contain the text entered in the input field. • Function is only available in expert mode.
	Export shown list as comma-separated list (*.csv)
	Print Preview
	Print list

18.8 Copying an interconnection

In contrast to copying/inserting selected interconnection elements via the clipboard, the function described in this chapter serves to replace the current FB interconnection of a device completely by the FB interconnection of another project device.

Note!

The complete FB interconnection can only be copied between devices of the same device type and version (e.g. 8400 HighLine C V1.0).

A complete interconnection comprises:

- Function blocks (use and parameter values)
- System blocks (application and parameter values)
- Port blocks (use and parameter values)
- Connections
- Comments
- Interconnection layout (arrangement of the modules)
- Port definition of the ports used in the FB interconnection

How to copy the complete interconnection into another project device:

1. Select the application with the FB interconnection to be copied in the *project view*.
2. Select the command **Application data→Copy FB interconnections...**.
3. Go to *project view* and select the application which is to be inserted into the copied FB interconnection.
4. Select the command **Application data→Add FB interconnection...**.
   - The command can only be activated if an FB interconnection has been copied from a device of the same device type and version.
   - After the command has been executed, the module assembly is compared. If there are relevant deviations, the insertion is refused and a corresponding message is displayed.
   - If an insertion is possible, you are asked if the FB interconnection is to be inserted.
5. Confirm the question if the copied FB interconnection is to be inserted with **Yes**.
   - After the insertion, an update of the project is required.

18.9 Exporting/Importing an interconnection

The interconnection existing in the project can be exported to a file for reuse/transfer to other devices.

Note!

The file can only be imported to devices of the same device type and version (e.g. 8400 HighLine C V1.0).

How to export the interconnection from the project to a file:

1. Go to the *Project view* in the *context menu* of the inverter and select the **Export FB interconnection...** command.
2. Enter the memory location and the file name for the interconnection to be exported in the *Export FB interconnection* dialog box.
3. Click **Save** to export the interconnection and close the dialog box.

How to import the interconnection from a file to the project:

1. Go to the *Project view* in the *context menu* of the inverter and select the **Import FB interconnection...** command.
2. Select the file with the interconnection to be imported in the *Import FB interconnection* dialog box.
3. Click **Open** to import the interconnection and close the dialog box.

19 Function library

19.1 Function blocks

This chapter describes the function blocks which are available for the inverter in the FB Editor.

The system blocks are described in the chapter "System blocks". (📖 1788)

Note!

A maximum of 75 function blocks can be used for a function block interconnection. The maximum calculating time is approx. 300 µs.

Overview of function blocks available

Function block	Runtime	Function
L_Absolute_1 L_Absolute_2	2 µs	... converts a bipolar input signal into a unipolar output signal.
L_AddSub_1	3 µs	... adds / subtracts analog input signals.
L_AnalogSwitch_1 L_AnalogSwitch_2 L_AnalogSwitch_3 L_AnalogSwitch_4 L_AnalogSwitch_5	2 µs	... switches between two analog input signals.
L_And_1 L_And_2 L_And_3	2 µs	... ANDs three binary signals.
L_And5_1 L_And5_2	2 µs	... ANDs five binary signals.
L_Arithmetik_1 L_Arithmetik_2 L_Arithmetik_3 L_Arithmetik_4 L_Arithmetik_5	7 µs	... combines two analog signals arithmetically.
L_ArithmetikPhi_1 L_ArithmetikPhi_2 L_ArithmetikPhi_3 L_ArithmetikPhi_4 L_ArithmetikPhi_5 L_ArithmetikPhi_6	7 µs	... combines two angle signals arithmetically. • L_ArithmetikPhi_4 ... L_ArithmetikPhi_6 are available from version 12.00.00.
L_CalcDiameter_1	2 µs	... can calculate the reel diameter from line speed and winding speed. • This FB is available from version 02.00.00.
L_Compare_1 L_Compare_2 L_Compare_3 L_Compare_4 L_Compare_5	5 µs	... compares two analog signals and can be used e.g. to implement a trigger.

Function block	Runtime	Function
L_ComparePhi_1 L_ComparePhi_2 L_ComparePhi_3 L_ComparePhi_4 L_ComparePhi_5	5 µs	... compares two angle signals.
L_ConvActPos_1	0 µs	... filters and scales a dancer position value to an available setting range. • This FB is available from version 12.00.00.
L_ConvAP_1 L_ConvAP_2 L_ConvAP_3	0 µs	... converts an analog value into a position. • These FBs are available from version 02.00.00.
L_ConvBitsToWord_1 L_ConvBitsToWord_2 L_ConvBitsToWord_3	3 µs	... converts 16 bit input values of the type "BOOL" into an output value of the type "WORD".
L_ConvDIntToWords_1 L_ConvDIntToWords_2 L_ConvDIntToWords_3	3 µs	... converts an input value of the type "DINT" into two output values of the type "WORD".
L_ConvPA_1 L_ConvPA_2 L_ConvPA_3	3 µs	... converts a position into an analog value. • These FBs are available from version 02.00.00.
L_ConvPP_1 L_ConvPP_2 L_ConvPP_3	3 µs	... converts a position with dynamic fraction. • These FBs are available from version 02.00.00.
L_ConvUnitsToIncr_1 L_ConvUnitsToIncr_2 L_ConvUnitsToIncr_3	3 µs	... converts a position value provided in the real unit of the machine into an internal 32-bit position value. • These FBs are available from version 12.00.00.
L_ConvW_1 L_ConvW_2 L_ConvW_3 L_ConvW_4	2 µs	... enables parameterisable conversion between analog signal forms. • Division is remainder considered.
L_ConvWordsToDInt_1 L_ConvWordsToDInt_2 L_ConvWordsToDInt_3	3 µs	... converts two inputs values of the type "WORD" into an output value of the type "DINT".
L_ConvWordToBits_1 L_ConvWordToBits_2 L_ConvWordToBits_3	3 µs	... converts an input value of "WORD" type into 16 individual binary signals.
L_ConvX_1 L_ConvX_2 L_ConvX_3	0 µs	... scales an analog value. • These FBs are available from version 02.00.00.
L_Counter_1 L_Counter_2 L_Counter_3	3 µs	... is a digital upcounter and downcounter.
L_Curve_1 L_Curve_2 L_Curve_3	4 µs	... can optionally display a characteristic function or a curve function y=f(x), the X axis being the input signal and the Y axis being the output signal. From version 12.00.00, the FB **L_Curve_3** for winding applications can also be used for creating a tensile force profile depending on the diameter. • L_Curve_2 and L_Curve_3 are available from version 02.00.00.
L_DFlipFlop_1 L_DFlipFlop_2	3 µs	... provides two stable states depending on the input signals.
L_DFRFG_1	3 µs	... synchronises a drive (slave) to the master value of a master drive and then executes a angular synchronism with regard to the digital frequency. • This FB is available from version 02.00.00.

Function block	Runtime	Function
L_DFSET_1	3 µs	... prepares the master value for a slave drive. This FB enables the inverter to follow the master drive true to speed and angle. • This FB is available from version 02.00.00.
L_DigitalDelay_1 L_DigitalDelay_2 L_DigitalDelay_3	2 µs	... delays binary signals.
L_DigitalLogic_1 L_DigitalLogic_2 L_DigitalLogic_3	2 µs	... provides a binary output signal which is generated by the logic combination of three input signals. • L_DigitalLogic_3 is available from version 02.00.00.
L_DigitalLogic5_1 L_DigitalLogic5_2	2 µs	... provides a binary output signal which is generated by the logic combination of five input signals.
L_DT1_1	3 µs	...differentiates signals. The function block can, for instance, be used to apply an acceleration (dv/dt).
L_FixSet_a_1	2 µs	... outputs one of 16 parameterisable analog signals.
L_FixSet_w_1 L_FixSet_w_2	2 µs	... outputs one of 16 parameterisable data words.
L_GainOffset_1 L_GainOffset_2 L_GainOffset_3	3 µs	... can amplify an analog input signal and then add an offset to it. • Gain and offset can be set via FB inputs.
L_GainOffsetP_1 L_GainOffsetP_2 L_GainOffsetP_3	3 µs	... can amplify an analog input signal and then add an offset to it. • Gain and offset can be set via parameters.
L_GainOffsetPhiP_1 L_GainOffsetPhiP_2	3 µs	... can amplify an angle signal and then add an offset to it. • Gain and offset can be set via parameters.
L_GearComp_1	3 µs	... compensates elasticities in the drive train (gearbox compensation). • This FB is available from version 02.00.00.
L_Interpolator_1	5 µs	... can interpolate a position setpoint and/or an analog value e.g. to compensate for larger bus transmission cycles or to continue signal characteristics if data telegrams are missing.
L_JogCtrlExtension_1	5 µs	... can be connected upstream to the L_NSet ramp generator to implement a switch-off positioning at limit switch.
L_Limit_1 L_Limit_2	3 µs	... limits an analog input signal to an adjustable value range.
L_LimitPhi_1 L_LimitPhi_2 L_LimitPhi_3	3 µs	... limits an angle signal to an adjustable value range.
L_MckCtrlInterface_1	5 µs	... provides the application with process inputs for controlling various basic functions of the Motion Control Kernel. ▸ MCKInterface (📖 599)
L_MckStateInterface_1	5 µs	... provides the application with various items of status information of the Motion Control via process outputs. ▸ MCKInterface (📖 599)
L_MFail_1	0 µs	... can be used for implementing a mains failure control. • This FB is available from version 12.00.00.
L_MPot_1	10 µs	... replaces a hardware motor potentiometer as setpoint source.
L_MulDiv_1 L_MulDiv_2	4 µs	... multiplies the analog input signal with a factor. • Not remainder considered.
L_Mux_1	3 µs	... switches one of eight selectable input signals to the output.
L_Negation_1 L_Negation_2	2 µs	... negates an analog input signal.
L_NLim_1 L_NLim_2	3 µs	... can suppress up to three parameterisable blocking zones within a continuous signal characteristic of an analog input signal.

Function block	Runtime	Function
L_Not_1 L_Not_2 L_Not_3 L_Not_4 L_Not_5 L_Not_6 L_Not_7	2 µs	... inverts a digital input signal.
L_NSet_1	50 µs	... contains a ramp generator with comprehensive parameterisation and control options to condition a setpoint signal.
L_Odometer_1	2 µs	... detects positions and calculates distances.
L_OffsetGain_1 L_OffsetGain_2 L_OffsetGain_3	4 µs	... can add an offset to an analog input signal and amplify it afterwards. • Offset and gain can be set via FB inputs.
L_OffsetGainP_1 L_OffsetGainP_2 L_OffsetGainP_3	17 µs	... can add an offset to an analog input signal and amplify it afterwards. • Offset and gain can be set via parameters.
L_OffsetGainPhiP_1 L_OffsetGainPhiP_2	17 µs	... can add an offset to an angle signal and amplify it afterwards. • Offset and gain can be set via parameters.
L_Or_1 L_Or_2 L_Or_3 L_Or_4	2 µs	... ORs three binary signals. • L_Or_4 is available from version 02.00.00.
L_Or5_1 L_Or5_2	2 µs	... ORs five binary signals.
L_PCTRL_1	20 µs	... is a PID inverter and can be used for various control tasks.
L_PhaseDiff_1 L_PhaseDiff_2	2 µs	... generates a position difference for the defined position setpoint from a position value and a speed signal.
L_PhaseIntK_1 L_PhaseIntK_2	5 µs	... integrates a speed to an angle.
L_PhiAdd_1	2 µs	... adds or subtracts angle signals. • This FB is available from version 12.00.00.
L_PhiDiv_1	2 µs	... divides or multiplies angle signals in the power of two format. • This FB is available from version 12.00.00.
L_PhiIntegrator_1	2 µs	... evaluates a speed with a gearbox factor and adds them in an integrator. • This FB is available from version 02.00.00.
L_PosCtrlLin_1 L_PosCtrlLin_2	2 µs	... is a simple linear profile generator. • These FBs are available from version 02.00.00.
L_PosiShaftCtrlInterface_1	5 µs	FB in preparation!
L_ProcessCtrl_1	2 µs	... can realise a dancer position or tension control. • This FB is available from version 02.00.00.
L_PT1_1 L_PT1_2 L_PT1_3	3 µs	... filters and delays analog signals.
L_RLQ_1	3 µs	... links a selected direction of rotation to the QSP function with wire-break protection.
L_RSFlipFlop_1 L_RSFlipFlop_2	3 µs	... saves a binary input information element and resets it on command..
L_SampleHold_1 L_SampleHold_2	3 µs	... saves a value.
L_Sequencer_1	10 µs	... processes a positioning program based on a sequence table. • This FB is available from version 12.00.00.
L_SignalMonitor_a	15 µs	... serves to output analog output signals of other FBs, SBs or LAs.

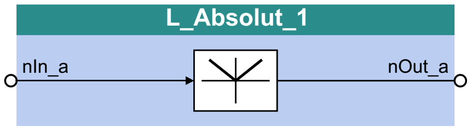
Function block	Runtime	Function
L_SignalMonitor_b	3 µs	... serves to output binary output signals of other FBs, SBs or LAs.
L_SignalSwitch_1 L_SignalSwitch_2 L_SignalSwitch_3 L_SignalSwitch_4	2 µs	... switches between two input signals of the "WORD" data type.
L_SignalSwitch32_1 L_SignalSwitch32_2 L_SignalSwitch32_3	2 µs	... .switches between two input signals of "DINT" data type. • These FBs are available from version 02.00.00.
L_SQrt_1	3 µs	... outputs the square root for a DINT input value.
L_SRFG_1 L_SRFG_2	2 µs	... is a ramp function generator with S-shaped ramps for limiting the temporal rise of analog signals.
L_SwitchPoint_1	3 µs	... provides position switch points, i.e. digital switches the binary statuses (FALSE/TRUE) of which depend on the actual position. • The position switch point positions can be set via FB inputs. • This FB is available from version 02.00.00.
L_SwitchPointPar_1	3 µs	... provides position switch points, i.e. digital switches the binary statuses (FALSE/TRUE) of which depend on the actual position. • The position switch point positions can be set via parameters. • This FB is available from version 12.00.00.
L_Transient_1 L_Transient_2 L_Transient_3 L_Transient_4 L_Transient_5 L_Transient_6 L_Transient_7 L_Transient_8	3 µs	... evaluates digital signal edges and converts them into timed pulses.

Related topics:

- Overview of system blocks available (🕮 1788)
- Working with the FB Editor (🕮 1416)

19.1.1 L_Absolut_1

This FB converts a bipolar input signal into a unipolar output signal.



inputs

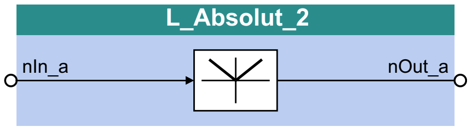
Designator Data type	Information/possible settings
nIn_a INT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.2 L_Absolut_2

This FB converts a bipolar input signal into a unipolar output signal.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

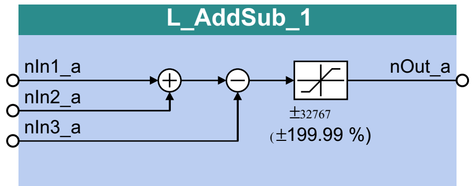
outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.3 L_AddSub_1

This FB has two adding inputs and one subtracting input.

- The value provided at the *nOut_a* output is internally limited to ±32767.



inputs

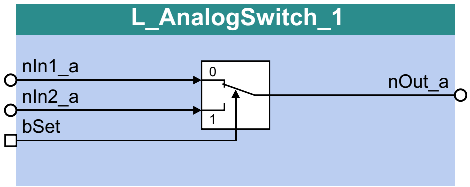
Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1 • This input is added
nIn2_a INT	Input signal 2 • This input is added
nIn3_a INT	Input signal 3 • This input is subtracted

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • *nOut_a = nIn1_a + nIn2_a - nIn3_a* • Internal limitation to ±32767 (±199.99 %)

19.1.4 L_AnalogSwitch_1

This FB changes over between two analog input signals. The change-over is controlled via a boolean input signal.



inputs

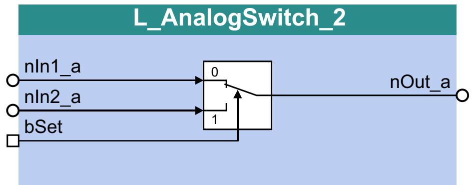
Designator Data type	Information/possible settings	
nIn1_a INT	Input signal 1	
nIn2_a INT	Input signal 2	
bSet BOOL	Selection of the input signal for the output to *nOut_a*	
	FALSE	*nIn1_a*
	TRUE	*nIn2_a*

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.5 L_AnalogSwitch_2

This FB changes over between two analog input signals. The change-over is controlled via a boolean input signal.



inputs

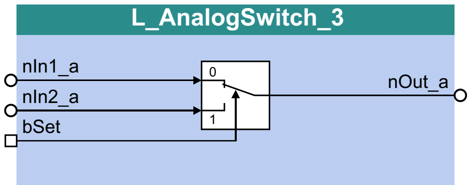
Designator Data type	Information/possible settings	
nIn1_a INT	Input signal 1	
nIn2_a INT	Input signal 2	
bSet BOOL	Selection of the input signal for the output to *nOut_a*	
	FALSE	*nIn1_a*
	TRUE	*nIn2_a*

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.6 L_AnalogSwitch_3

This FB changes over between two analog input signals. The change-over is controlled via a boolean input signal.



inputs

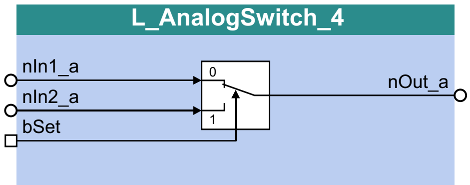
Designator Data type	Information/possible settings	
nIn1_a INT	Input signal 1	
nIn2_a INT	Input signal 2	
bSet BOOL	Selection of the input signal for the output to *nOut_a*	
	FALSE	*nIn1_a*
	TRUE	*nIn2_a*

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.7 L_AnalogSwitch_4

This FB changes over between two analog input signals. The change-over is controlled via a boolean input signal.



inputs

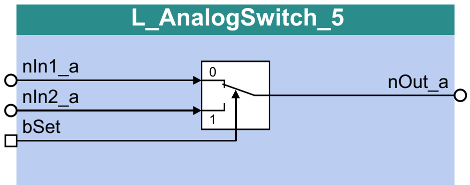
Designator Data type	Information/possible settings	
nIn1_a INT	Input signal 1	
nIn2_a INT	Input signal 2	
bSet BOOL	Selection of the input signal for the output to *nOut_a*	
	FALSE	*nIn1_a*
	TRUE	*nIn2_a*

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.8 L_AnalogSwitch_5

This FB changes over between two analog input signals. The change-over is controlled via a boolean input signal.



inputs

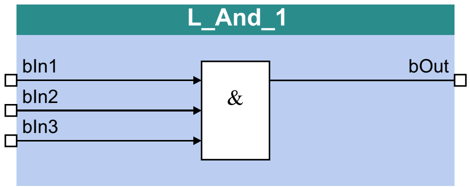
Designator Data type	Information/possible settings	
nIn1_a INT	Input signal 1	
nIn2_a INT	Input signal 2	
bSet BOOL	Selection of the input signal for the output to *nOut_a*	
	FALSE	*nIn1_a*
	TRUE	*nIn2_a*

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.9 L_And_1

This FB implements AND operations for input signals.



inputs

Designator Data type	Information/possible settings
bIn1 bIn2 bIn3 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

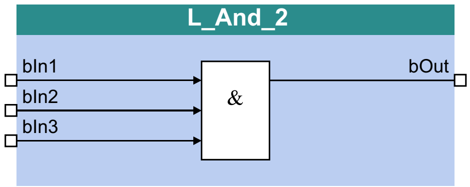
Function

inputs			Output
bIn3	bIn2	bIn1	bOut
FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	
FALSE	TRUE	FALSE	
FALSE	TRUE	TRUE	
TRUE	FALSE	FALSE	
TRUE	FALSE	TRUE	
TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE

[19-1] Truth table of the FB L_And_1

19.1.10 L_And_2

This FB implements AND operations for input signals.



inputs

Designator Data type	Information/possible settings
bIn1 bIn2 bIn3 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

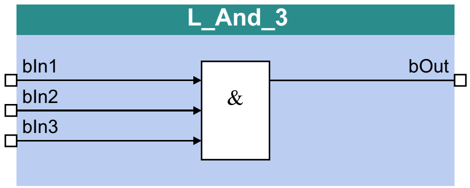
Function

inputs			Output
bIn3	**bIn2**	**bIn1**	**bOut**
FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	
FALSE	TRUE	FALSE	
FALSE	TRUE	TRUE	
TRUE	FALSE	FALSE	
TRUE	FALSE	TRUE	
TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE

[19-2] Truth table of the FB L_And_2

19.1.11 L_And_3

This FB implements AND operations for input signals.



inputs

Designator Data type	Information/possible settings
bIn1 bIn2 bIn3 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

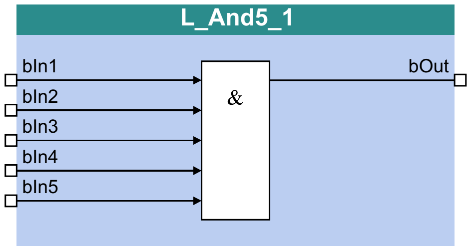
Function

inputs			Output
bIn3	**bIn2**	**bIn1**	**bOut**
FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	
FALSE	TRUE	FALSE	
FALSE	TRUE	TRUE	
TRUE	FALSE	FALSE	
TRUE	FALSE	TRUE	
TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE

[19-3] Truth table of the FB L_And_3

19.1.12 L_And5_1

This FB implements AND operations for input signals.



inputs

Designator Data type	Information/possible settings
bIn1 ... bIn5 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

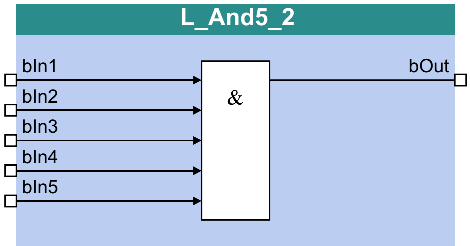
Function

inputs					Output
bIn5	**bIn4**	**bIn3**	**bIn2**	**bIn1**	**bOut**
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	FALSE	FALSE	TRUE	
FALSE	FALSE	FALSE	TRUE	FALSE	
FALSE	FALSE	FALSE	TRUE	TRUE	
FALSE	FALSE	TRUE	FALSE	FALSE	
...					
TRUE	TRUE	TRUE	FALSE	TRUE	
TRUE	TRUE	TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

[19-4] Truth table of the FB L_And5_1

19.1.13 L_And5_2

This FB implements AND operations for input signals.



inputs

Designator Data type	Information/possible settings
bIn1 ... bIn5 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

Function

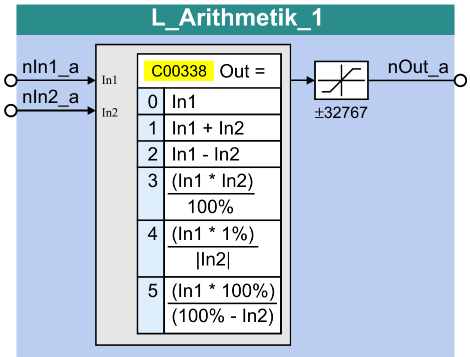
inputs					Output
bIn5	bIn4	bIn3	bIn2	bIn1	bOut
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	FALSE	FALSE	TRUE	
FALSE	FALSE	FALSE	TRUE	FALSE	
FALSE	FALSE	FALSE	TRUE	TRUE	
FALSE	FALSE	TRUE	FALSE	FALSE	
...					
TRUE	TRUE	TRUE	FALSE	TRUE	
TRUE	TRUE	TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

[19-5] Truth table of the FB L_And5_2

19.1.14 L_Arithmetik_1

This FB can implement an arithmetic link between two analog signals.

- The arithmetic function is selected in C00338.
- All internal intermediate results and the value output at the *nOut_a* output are internally limited to ±32767.
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±32767 (±199.99 %)

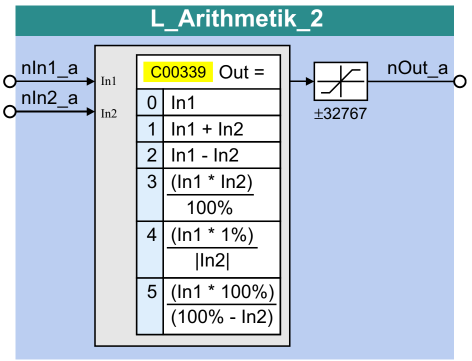
Parameters

Parameters	Possible settings		Information
C00338			Function selection
	0	$nOut\_a = nIn1\_a$	
	1	$nOut\_a = nIn1\_a + nIn2\_a$	
	2	$nOut\_a = nIn1\_a - nIn2\_a$	
	3	$nOut\_a = \frac{nIn1\_a \cdot nIn2\_a}{16384}$	
	4	$nOut\_a = \frac{nIn1\_a}{\lvert nIn2\_a \rvert} \cdot 164$	When the denominator has the value "0", it will be set to "1".
	5	$nOut\_a = \frac{nIn1\_a}{16384 - nIn2\_a} \cdot 16384$	

19.1.15 L_Arithmetik_2

This FB can implement an arithmetic link between two analog signals.

- The arithmetic function is selected in C00339.
- All internal intermediate results and the value output at the *nOut_a* output are internally limited to ±32767.
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±32767 (±199.99 %)

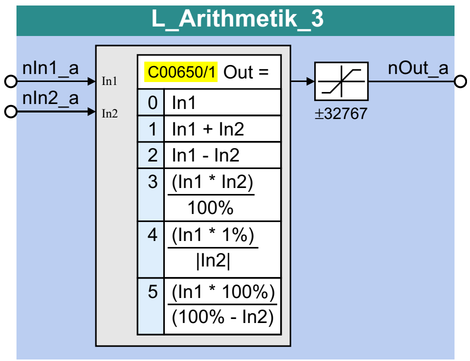
Parameters

Parameters	Possible settings		Information
C00339			Function selection
	0	$nOut\_a = nIn1\_a$	
	1	$nOut\_a = nIn1\_a + nIn2\_a$	
	2	$nOut\_a = nIn1\_a - nIn2\_a$	
	3	$nOut\_a = \frac{nIn1\_a \cdot nIn2\_a}{16384}$	
	4	$nOut\_a = \frac{nIn1\_a}{\lvert nIn2\_a \rvert} \cdot 164$	When the denominator has the value "0", it will be set to "1".
	5	$nOut\_a = \frac{nIn1\_a}{16384 - nIn2\_a} \cdot 16384$	

19.1.16 L_Arithmetik_3

This FB can implement an arithmetic link between two analog signals.

- The arithmetic function is selected in C00650/1.
- All internal intermediate results and the value output at the *nOut_a* output are internally limited to ±32767.
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±32767 (±199.99 %)

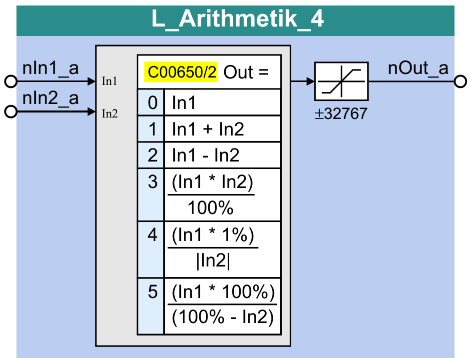
Parameters

Parameters	Possible settings		Information
C00650/1			Function selection
	0	$nOut\_a = nIn1\_a$	
	1	$nOut\_a = nIn1\_a + nIn2\_a$	
	2	$nOut\_a = nIn1\_a - nIn2\_a$	
	3	$nOut\_a = \frac{nIn1\_a \cdot nIn2\_a}{16384}$	
	4	$nOut\_a = \frac{nIn1\_a}{\lvert nIn2\_a \rvert} \cdot 164$	When the denominator has the value "0", it will be set to "1".
	5	$nOut\_a = \frac{nIn1\_a}{16384 - nIn2\_a} \cdot 16384$	

19.1.17 L_Arithmetik_4

This FB can implement an arithmetic link between two analog signals.

- The arithmetic function is selected in C00650/2.
- All internal intermediate results and the value output at the *nOut_a* output are internally limited to ±32767.
- Division is not remainder considered.



inputs

Designator (Data type)	Information/possible settings
nIn1_a (INT)	Input signal 1
nIn2_a (INT)	Input signal 2

outputs

Designator (Data type)	Value/meaning
nOut_a (INT)	Output signal • Internal limitation to ±32767 (±199.99 %)

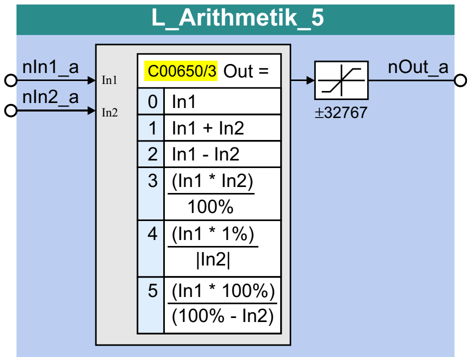
Parameters

Parameters	Possible settings		Information
C00650/2			Function selection
	0	$nOut\_a = nIn1\_a$	
	1	$nOut\_a = nIn1\_a + nIn2\_a$	
	2	$nOut\_a = nIn1\_a - nIn2\_a$	
	3	$nOut\_a = \frac{nIn1\_a \cdot nIn2\_a}{16384}$	
	4	$nOut\_a = \frac{nIn1\_a}{\lvert nIn2\_a \rvert} \cdot 164$	When the denominator has the value "0", it will be set to "1".
	5	$nOut\_a = \frac{nIn1\_a}{16384 - nIn2\_a} \cdot 16384$	

19.1.18 L_Arithmetik_5

This FB can implement an arithmetic link between two analog signals.

- The arithmetic function is selected in C00650/3.
- All internal intermediate results and the value output at the *nOut_a* output are internally limited to ±32767.
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±32767 (±199.99 %)

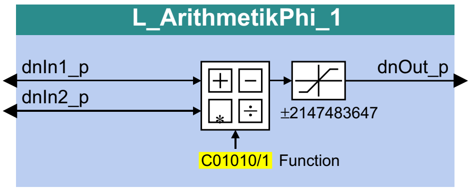
Parameters

Parameters	Possible settings		Information
C00650/3			Function selection
	0	$nOut\_a = nIn1\_a$	
	1	$nOut\_a = nIn1\_a + nIn2\_a$	
	2	$nOut\_a = nIn1\_a - nIn2\_a$	
	3	$nOut\_a = \frac{nIn1\_a \cdot nIn2\_a}{16384}$	
	4	$nOut\_a = \frac{nIn1\_a}{\lvert nIn2\_a \rvert} \cdot 164$	When the denominator has the value "0", it will be set to "1".
	5	$nOut\_a = \frac{nIn1\_a}{16384 - nIn2\_a} \cdot 16384$	

19.1.19 L_ArithmetikPhi_1

This FB can implement an arithmetic link between two angle signals.

- The arithmetic function is selected in C01010/1.
- All internal intermediate results and the value provided at the *dnOut_p* output are internally limited to ±2147483647 (±2^{31} - 1).
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647

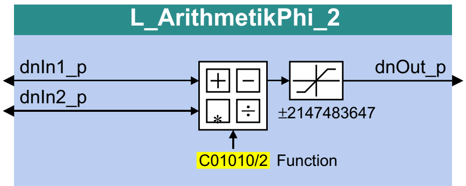
Parameters

Parameters	Possible settings		Information
C01010/1			Function selection
	0	dnOut_p = dnIn1_p	
	1	dnOut_p = dnIn1_p + dnIn2_p	
	2	dnOut_p = dnIn1_p - dnIn2_p	
	3	dnOut_p = dnIn1_p * dnIn2_p	
	4	dnOut_p = dnIn1_p / dnIn2_p	

19.1.20 L_ArithmetikPhi_2

This FB can implement an arithmetic link between two angle signals.

- The arithmetic function is selected in C01010/2.
- All internal intermediate results and the value provided at the *dnOut_p* output are internally limited to ±2147483647 (±2^{31} - 1).
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647

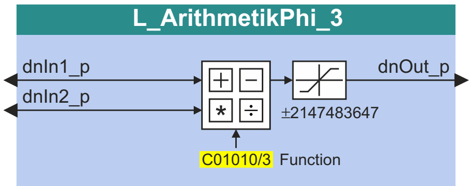
Parameters

Parameters	Possible settings		Information
C01010/2			Function selection
	0	dnOut_p = dnIn1_p	
	1	dnOut_p = dnIn1_p + dnIn2_p	
	2	dnOut_p = dnIn1_p - dnIn2_p	
	3	dnOut_p = dnIn1_p * dnIn2_p	
	4	dnOut_p = dnIn1_p / dnIn2_p	

19.1.21 L_ArithmetikPhi_3

This FB can implement an arithmetic link between two angle signals.

- The arithmetic function is selected in C01010/3.
- All internal intermediate results and the value provided at the *dnOut_p* output are internally limited to ±2147483647 (±2^{31} - 1).
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647

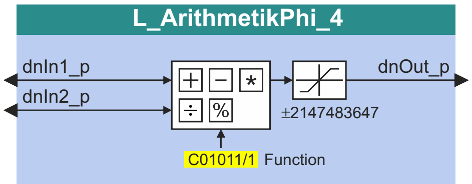
Parameters

Parameters	Possible settings		Information
C01010/3			Function selection
	0	dnOut_p = dnIn1_p	
	1	dnOut_p = dnIn1_p + dnIn2_p	
	2	dnOut_p = dnIn1_p - dnIn2_p	
	3	dnOut_p = dnIn1_p * dnIn2_p	
	4	dnOut_p = dnIn1_p / dnIn2_p	

19.1.22 L_ArithmetikPhi_4

This FB can implement an arithmetic link between two angle signals.

- The arithmetic function is selected in C01011/1.
- All internal intermediate results and the value provided at the *dnOut_p* output are internally limited to ±2147483647 ($\pm 2^{31}$ - 1).
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647

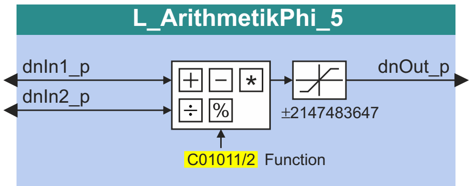
Parameters

Parameters	Possible settings		Information
C01011/1			Function selection
	0	dnOut_p = dnIn1_p	
	1	dnOut_p = dnIn1_p + dnIn2_p	
	2	dnOut_p = dnIn1_p - dnIn2_p	
	3	dnOut_p = dnIn1_p * dnIn2_p	
	4	dnOut_p = dnIn1_p / dnIn2_p	
	5	dnOut_p = dnIn1_p % dnIn2_p	From version 16.00.00
	6	dnOut_p = dnIn1_p + dnIn2_p	From version 16.00.00 Without limitation of the output signal
	7	dnOut_p = dnIn1_p - dnIn2_p	From version 16.00.00 Without limitation of the output signal

19.1.23 L_ArithmetikPhi_5

This FB can implement an arithmetic link between two angle signals.

- The arithmetic function is selected in C01011/2.
- All internal intermediate results and the value provided at the *dnOut_p* output are internally limited to ±2147483647 ($\pm 2^{31}$ - 1).
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647

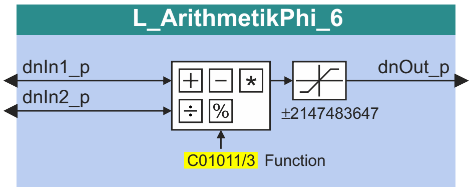
Parameters

Parameters	Possible settings		Information
C01011/2			Function selection
	0	dnOut_p = dnIn1_p	
	1	dnOut_p = dnIn1_p + dnIn2_p	
	2	dnOut_p = dnIn1_p - dnIn2_p	
	3	dnOut_p = dnIn1_p * dnIn2_p	
	4	dnOut_p = dnIn1_p / dnIn2_p	
	5	dnOut_p = dnIn1_p % dnIn2_p	From version 16.00.00
	6	dnOut_p = dnIn1_p + dnIn2_p	From version 16.00.00 Without limitation of the output signal
	7	dnOut_p = dnIn1_p - dnIn2_p	From version 16.00.00 Without limitation of the output signal

19.1.24 L_ArithmetikPhi_6

This FB can implement an arithmetic link between two angle signals.

- The arithmetic function is selected in C01011/3.
- All internal intermediate results and the value provided at the *dnOut_p* output are internally limited to ±2147483647 (±2^{31} - 1).
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

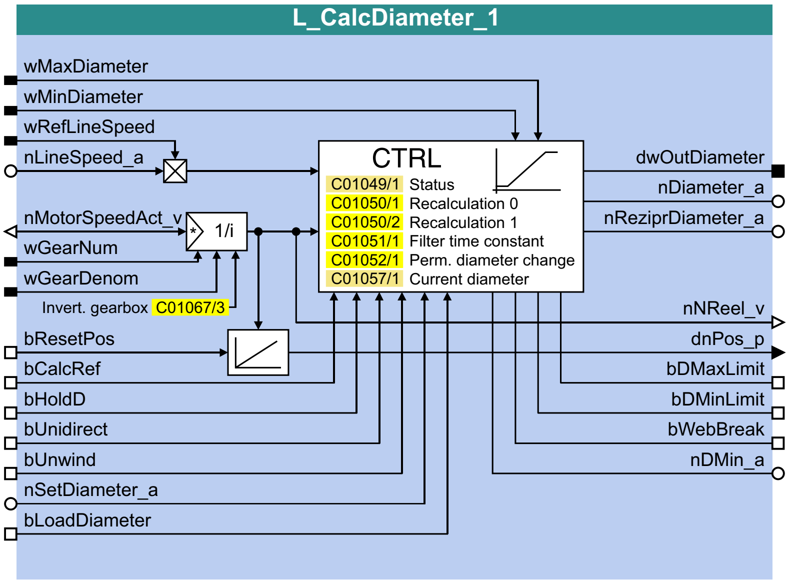
Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647

Parameters

Parameters	Possible settings		Information
C01011/3			Function selection
	0	dnOut_p = dnIn1_p	
	1	dnOut_p = dnIn1_p + dnIn2_p	
	2	dnOut_p = dnIn1_p - dnIn2_p	
	3	dnOut_p = dnIn1_p * dnIn2_p	
	4	dnOut_p = dnIn1_p / dnIn2_p	
	5	dnOut_p = dnIn1_p % dnIn2_p	From version 16.00.00
	6	dnOut_p = dnIn1_p + dnIn2_p	From version 16.00.00 Without limitation of the output signal
	7	dnOut_p = dnIn1_p - dnIn2_p	From version 16.00.00 Without limitation of the output signal

19.1.25 L_CalcDiameter_1

This FB serves to calculate the reel diameter from line speed and winding speed.



inputs

Designator Data type	Information/possible settings	
wMaxDiameter WORD	Maximum diameter in [mm] • Internal limitation to 10000 mm (≡ 10 m)	
wMinDiameter WORD	Minimum diameter in [mm] • Internal limitation to 10000 mm (≡ 10 m)	
wRefLineSpeed WORD	Maximum material speed in [0.1 m/min] • Scaling: 2500 ≡ 250.0 m/min • Internal limitation to 3000 m/min	
nLineSpeed_a INT	Material speed setpoint • Scaling: 16384 ≡ maximum material speed (*wRefLineSpeed*)	
nMotorSpeedAct_v INT	Current motor speed in [increments/ms] • Scaling: 16384 ≡ 15000 rpm	
wGearNum WORD	Gearbox factor (numerator) • Internal limitation to -32767 ... -1 / 1 ... 32767	
wGearDenom WORD	Gearbox factor (denominator) • Internal limitation to 1 ... 32767	
bResetPos BOOL	Reset angle of rotation *dnPos_p*	
	TRUE	The angle of rotation *dnPos_p* is reset.
bCalcRef BOOL	Selection of the calculation cycle	
	FALSE	Use diameter recalculation 0 (C01050/1).
	TRUE	Use diameter recalculation 1 (C01050/2).
bHoldD BOOL	Hold last diameter value	
	TRUE	The diameter value output last is not overwritten with new values.
bUnidirect BOOL	Enable of only one change of direction and activation of the web break monitoring	
	TRUE	Only the change of direction defined via *bUnwind* is enabled.

Designator Data type	Information/possible settings	
bUnwind BOOL	Selection of the enabled change or direction • Only when *bUnidirect* = TRUE	
	FALSE	Clockwise rotation (Cw)
	TRUE	Counter-clockwise rotation (Ccw)
nSetDiameter_a DINT	Selection of an initial value/external diameter signal • Scaling: 16384 ≡ maximum diameter (*wMaxDiameter*) • The value is accepted by setting *bLoadDiameter* to TRUE.	
bLoadDiameter BOOL	Accept initial value *nSetDiameter_a* • This input has a higher priority than the *bHoldD*input.	
	TRUE	Accept the value at *nSetDiameter_a*.

outputs

Designator Data type	Value/meaning	
dwOutDiameter DWORD	Current diameter in [µm] • Internal limitation to 10 m • Unfiltered	
nDiameter_a INT	Current diameter in [%] • 100 % ≡ Maximum diameter (*wMaxDiameter*) • filtered via PT1 element (filter time constant can be set in C01051/1)	
nReziprDiameter_a INT	Reciprocal value of the current diameter in [%] • 100 % ≡ Minimum diameter (*wMinDiameter*)	
nNReel_v INT	Current winding speed in [increments/ms] • Scaling: 16384 ≡ 15000 rpm	
bDMaxLimit BOOL	Limit value monitoring	
	TRUE	Upper limit value *wMaxDiameter* reached.
bDMinLimit BOOL	Limit value monitoring	
	TRUE	Lower limit value *wMinDiameter* reached.
bWebBreak BOOL	Web break monitoring	
	TRUE	Web break after passing the monitor window set in C01052/1.
nDMin_a INT	Minimum diameter in [%] • 100 % ≡ Maximum diameter (*wMaxDiameter*)	

Parameters

Parameters	Possible settings		Information
C01049/1			Status
	-10	Line speed overflow	
	-1	*wMinDiameter* > *wMaxDiameter*	
	0	OK - diameter has been recalculated	
	10	Diameter has been initialised with minimum diameter (*wMinDiameter)*	
	20	Diameter is loaded (*bLoadDiameter* active)	
	30	Diameter is held (*bHoldD* active)	

Parameters	Possible settings			Information
C01050/1	0.001	Rev.	2.000	Diameter recalculation 0 • Number of revolutions after which a diameter calculation is executed. • Setting is effective when *bCalcRef* = FALSE. • Initialisation: 1.000
C01050/2	0.001	Rev.	2.000	Diameter recalculation 1 • Number of revolutions after which a diameter calculation is executed. • Setting is effective when *bCalcRef* = TRUE. • Initialisation: 0.100
C01051/1	0.010	s	3.000	Filter time constant for calculated diameter values • Initialisation: 1.000 s
C01052/1	0.00	%	100.00	Permissible diameter change in opposite direction (web break monitoring) • Only relevant when *bUnidirect* = TRUE. • Referring to the maximum diameter (*wMaxDiameter*). • Initialisation: 10.00 %
C01057/1	0.000	mm	10000.000	Current diameter • Read only
C01067/3				Invert. gearbox nMotorSpeedAct_v
	0	Not inverted		
	1	Inverted		
	2	Automatically from MCK		

19.1.25.1 Set initial value

An initial value or an external diameter signal can be defined at the *nSetDiameter_a* input.

- this value is accepted if *bLoadDiameter* is set to TRUE.
- In case of acceptance, the diameter calculation will be reset and the filter for diameter calculation is loaded with the defined initial value.

19.1.25.2 Calculate diameter

For reel diameter calculation, the material speed and the current winding speed are integrated cyclically.

- The material speed is defined via *nLineSpeed_a*.
- The winding speed is calculated from the *nMotorSpeedAct_v* motor speed and the current *wGearNum*/*wGearDenom* gearbox factor.
- C01050/1...2 can be used to define two different calculation cycles. The setting to the used is selected via *bCalcRef*.
- At the end of each integration interval, a new diameter value results from the division of the integrator values.
- An absolute value generation and smoothing via a first order low pass is carried out. The time constant of this filter can be set in C01051/1. The *dwOutDiameter* output is not filtered.

19.1.25.3 Select change direction, web break monitoring

By setting *bUnidirect* to TRUE, you only enable one change direction for diameter calculation and simultaneously activate the web break monitoring.

- The permissible change direction can be defined via *bUnwind*.
- A recalculated value is only accepted if it exceeds the value saved last in the permissible direction.

After a web break, the diameter values calculated successively mostly run very quickly in the opposite direction.

- C01052/1 serves to define the maximally permissible deviation in the opposite direction. If it is exceeded, the *bWebBreak* output is set to TRUE.
- If *bUnidirect* is set to FALSE, both change directions are enabled and the internal memory is always overwritten with the recalculated value which switches off the web break monitoring.

19.1.25.4 Holding the current value

By setting *bHoldD* to TRUE, you fix the diameter value *nDiameter_a* output last.

- The diameter calculation is reset and the filter for diameter calculation is loaded with the internally held diameter value.

19.1.25.5 Limit value monitoring

wMinDiameter and *wMaxDiameter* serve to define the limit values for the given diameter values. If they are reached/exceeded, the corresponding ouput (*bDMaxLimit*/*bDminLimit*) is set to TRUE.

- The unfiltered diameter value *dwOutDiameter* and the filtered diameter value *nDiameter_a* are limited.
- The limit values are entered in [mm].
- The hysteresis for resetting *bDMaxLimit*/*bDminLimit* is permanently set to 1 % of *wMaxDiameter*.
- For parameter setting of further FBs, the value in *wMinDiameter* with regard to *wMaxDiameter* is provided via the *nDMin_a* output.

19.1.25.6 Converting diameter in 1/D

For evaluating the material speed with 1/D (speed-controlled winder), the reciprocal value of the reel diameter is provided via *nReziprDiameter_a*.

- This value refers to the limitation value *wMinDiameter*.

19.1.25.7 Physical state variables

The *dwOutDiameter* output provides the current diameter as physical quantity (1 LSB ≡ 1 m).

nNReel_v outputs the winding speed which results from the *nMotorSpeedAct_v* motor speed and the current (*wGearNum*/*wGearDenom*) gearbox factor.

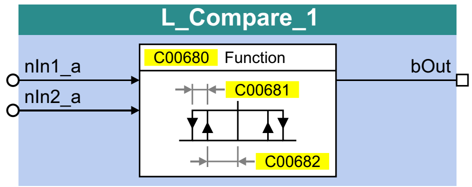
For supporting further functions (e.g. a traversing control), the winding speed is integrated additionally.

- The angle of rotation is output via *dnPos_p*.
- By setting *bResetPos* to TRUE, the angle of rotation is reset.

19.1.26 L_Compare_1

This FB compares two analog signals and can be used e.g. to implement a trigger.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

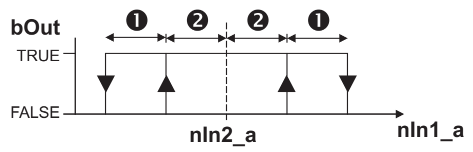
Parameters

Parameters	Possible settings			Information
C00680				Function selection
	1	nIn1 = nIn2		
	2	nIn1 > nIn2		
	3	nIn1 < nIn2		
	4	|nIn1| = |nIn2|		
	5	|nIn1| > |nIn2|		
	6	|nIn1| < |nIn2|		
C00681	0.00	%	100.00	Hysteresis • Lenze setting: 0.50 %
C00682	0.00	%	100.00	Window • Lenze setting: 2.00 %

19.1.26.1 Function 1: nIn1 = nIn2

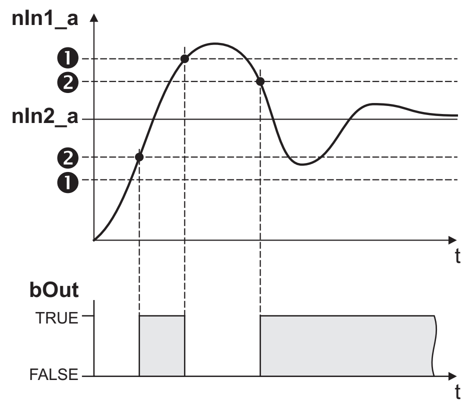
This function compares two signals with regard to equality. It can, for instance, provide the comparison "actual speed equals setpoint speed" ($n_{act} = n_{set}$).

- Use C00682 to set the window within which the equality is to apply.
- Use C00681 to set a hysteresis if the input signals are not stable and the output oscillates.



❶ C00681: Hysteresis
❷ C00682: Window

[19-6] Function 1: Switching performance

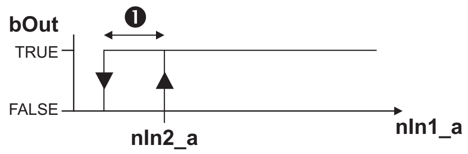


❶ C00681: Hysteresis
❷ C00682: Window

[19-7] Function 1: Example

19.1.26.2 Function 2: nIn1 > nIn2

This function serves, for instance, to implement the comparison "actual speed is higher than a limit value" ($n_{act} > n_x$) for one direction of rotation.

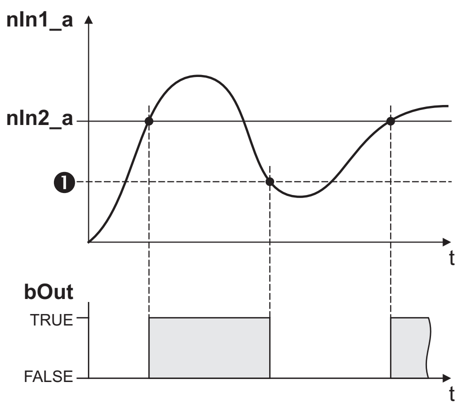


❶ C00681: Hysteresis

[19-8] Function 2: Switching performance

Functional sequence

1. If the value at *nIn1_a* exceeds the value *nIn2_a*, *bOut* changes from FALSE to TRUE.
2. Only if the signal at *nIn1_a* falls below the value of *nIn2_a - hysteresis* again, *bOut* changes back from TRUE to FALSE.

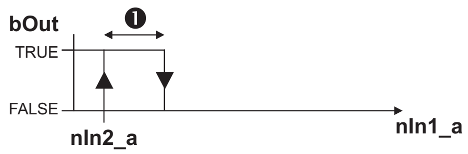


❶ C00681: Hysteresis

[19-9] Function 2: Example

19.1.26.3 Function 3: nIn1 < nIn2

This function serves, for instance, to implement the comparison "actual speed is lower than a limit value" ($n_{act} < n_x$) for one direction of rotation.

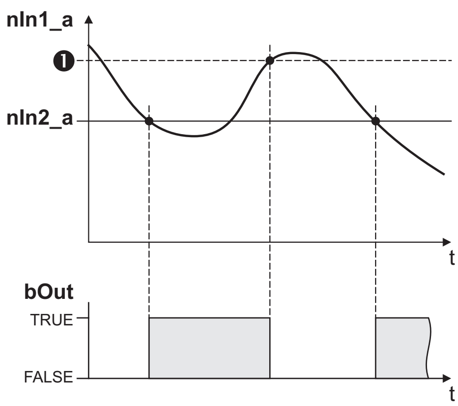


❶ C00681: Hysteresis

[19-10] Function 3: Switching performance

Functional sequence

1. If the value at *nIn1_a* falls below the value at *nIn2_a*, *bOut* changes from FALSE to TRUE.
2. Only if the signal at *nIn1_a* exceeds the value of *nIn2_a - hysteresis* again, *bOut* changes back from TRUE to FALSE.



❶ C00681: Hysteresis

[19-11] Function 3: Example

19.1.26.4 Function 4: |nIn1| = |nIn2|

This function serves to implement e.g. the comparison "n_{act} = 0". This function is similar to function 1. However, the amount is generated by the input signals before signal processing (without sign).

▶ Function 1: nIn1 = nIn2

19.1.26.5 Function 5: |nIn1| > |nIn2|

This function serves to implement e.g. the comparison "$|n_{act}| > |n_x|$" irrespective of the direction of rotation. This function is similar to function 2. However, the amount is generated by the input signals before signal processing (without sign).

▶ Function 2: nIn1 > nIn2

19.1.26.6 Function 6: |nIn1| < |nIn2|

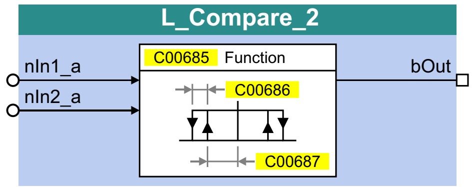
This function serves to implement the comparison "$|n_{act}| < |n_x|$" independent of the direction of rotation. This function is similar to function 3. However, the amount is generated by the input signals before signal processing (without sign).

▶ Function 3: nIn1 < nIn2

19.1.27 L_Compare_2

This FB compares two analog signals and can be used e.g. to implement a trigger.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

Parameters

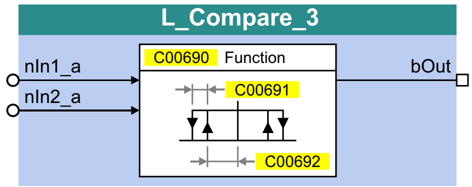
Parameters	Possible settings			Information
C00685				Function selection
	1	nIn1 = nIn2		
	2	nIn1 > nIn2		
	3	nIn1 < nIn2		
	4	|nIn1| = |nIn2|		
	5	|nIn1| > |nIn2|		
	6	|nIn1| < |nIn2|		
C00686	0.00	%	100.00	Hysteresis • Lenze setting: 0.50 %
C00687	0.00	%	100.00	Window • Lenze setting: 2.00 %

For a detailed functional description see L_Compare_1.

19.1.28 L_Compare_3

This FB compares two analog signals and can be used e.g. to implement a trigger.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

Parameters

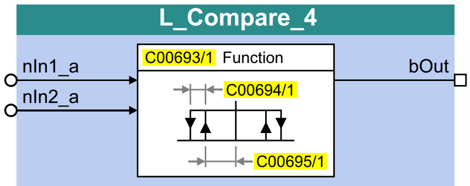
Parameters	Possible settings			Information
C00690				Function selection
	1	nIn1 = nIn2		
	2	nIn1 > nIn2		
	3	nIn1 < nIn2		
	4	|nIn1| = |nIn2|		
	5	|nIn1| > |nIn2|		
	6	|nIn1| < |nIn2|		
C00691	0.00	%	100.00	Hysteresis • Lenze setting: 0.50 %
C00692	0.00	%	100.00	Window • Lenze setting: 2.00 %

For a detailed functional description see L_Compare_1.

19.1.29 L_Compare_4

This FB compares two analog signals and can be used e.g. to implement a trigger.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

Parameters

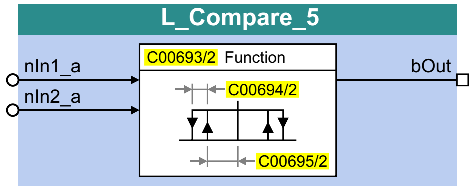
Parameters	Possible settings			Information
C00693/1				Function selection
	1	nIn1 = nIn2		
	2	nIn1 > nIn2		
	3	nIn1 < nIn2		
	4	|nIn1| = |nIn2|		
	5	|nIn1| > |nIn2|		
	6	|nIn1| < |nIn2|		
C00694/1	0.00	%	100.00	Hysteresis • Lenze setting: 0.00 %
C00695/1	0.00	%	100.00	Window • Lenze setting: 0.00 %

For a detailed functional description see L_Compare_1.

19.1.30 L_Compare_5

This FB compares two analog signals and can be used e.g. to implement a trigger.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
nIn1_a INT	Input signal 1
nIn2_a INT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

Parameters

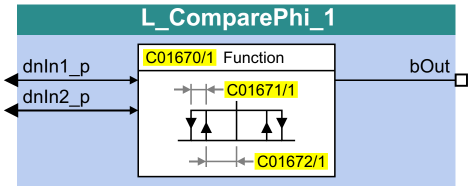
Parameters	Possible settings			Information
C00693/2				Function selection
	1	nIn1 = nIn2		
	2	nIn1 > nIn2		
	3	nIn1 < nIn2		
	4	|nIn1| = |nIn2|		
	5	|nIn1| > |nIn2|		
	6	|nIn1| < |nIn2|		
C00694/2	0.00	%	100.00	Hysteresis • Lenze setting: 0.00 %
C00695/2	0.00	%	100.00	Window • Lenze setting: 0.00 %

For a detailed functional description see L_Compare_1.

19.1.31 L_ComparePhi_1

This FB compares two angle signals.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

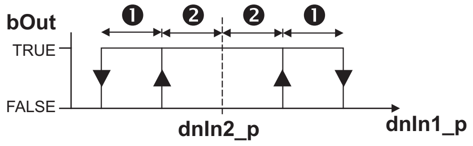
Parameters

Parameters	Possible settings			Information
C01670/1				Function selection
	1	dnIn1 = dnIn2		
	2	dnIn1 > dnIn2		
	3	dnIn1 < dnIn2		
	4	|dnIn1| = |dnIn2|		
	5	|dnIn1| > |dnIn2|		
	6	|dnIn1| < |dnIn2|		
C01671/1	0	Incr.	1073741824	Hysteresis • Lenze setting: 0 incr.
C01672/1	0	Incr.	1073741824	Window • Lenze setting: 0 incr.

19.1.31.1 Function 1: dnIn1 = dnIn2

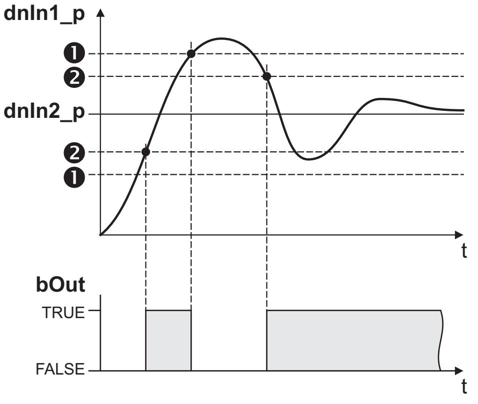
This function compares two signals with regard to equality. It can, for instance, provide the comparison "actual speed equals setpoint speed" ($n_{act} = n_{set}$).

- Use C01672 to set the window within which the equality is to apply.
- Use C01671 to set a hysteresis if the input signals are not stable and the output oscillates.



❶ C001671: Hysteresis
❷ C001672: Window

[19-12] Function 1: Switching performance

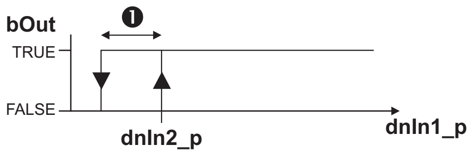


❶ C001671: Hysteresis
❷ C001672: Window

[19-13] Function 1: Example

19.1.31.2 Function 2: dnIn1 > dnIn2

This function serves, for instance, to implement the comparison "actual speed is higher than a limit value" ($n_{act} > n_x$) for one direction of rotation.

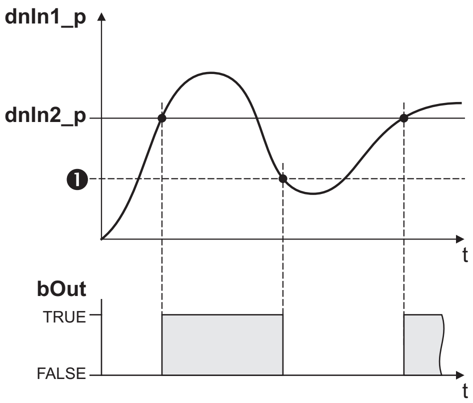


❶ C001671: Hysteresis

[19-14] Function 2: Switching performance

Functional sequence

1. If the value at *dnIn1_p* exceeds the value at *dnIn2_p*, *bOut* changes from FALSE to TRUE.
2. Only if the signal at *dnIn1_p* falls below the value of *dnIn2_p - hysteresis* again, *bOut* changes back from TRUE to FALSE.

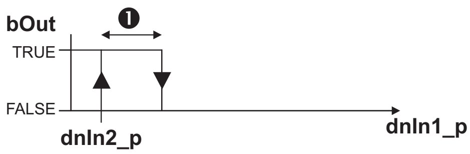


❶ C001671: Hysteresis

[19-15] Function 2: Example

19.1.31.3 Function 3: dnIn1 < dnIn2

This function serves, for instance, to implement the comparison "actual speed is lower than a limit value" ($n_{act} < n_x$) for one direction of rotation.

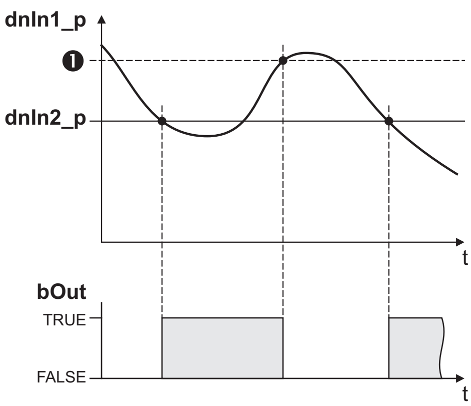


❶ C001671: Hysteresis

[19-16] Function 3: Switching performance

Functional sequence

1. If the value at *dnIn1_p* falls below the value at *dnIn2_p*, *bOut* changes from FALSE to TRUE.
2. Only if the signal at *dnIn1_p* exceeds the value of *dnIn2_p* - *hysteresis* again, *bOut* changes back from TRUE to FALSE.



❶ C001671: Hysteresis

[19-17] Function 3: Example

19.1.31.4 Function 4: |dnIn1| = |dnIn2|

This function serves to implement e.g. the comparison "n_{act} = 0". This function is similar to function 1. However, the amount is generated by the input signals before signal processing (without sign).

Function 1: dnIn1 = dnIn2

19.1.31.5 Function 5: |dnIn1| > |dnIn2|

This function serves to implement e.g. the comparison "$|n_{act}| > |n_x|$" irrespective of the direction of rotation. This function is similar to function 2. However, the amount is generated by the input signals before signal processing (without sign).

Function 2: dnIn1 > dnIn2

19.1.31.6 Function 6: |dnIn1| < |dnIn2|

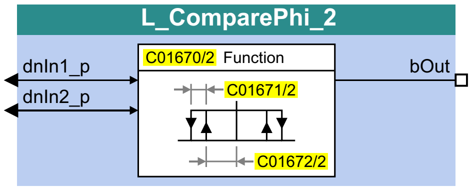
This function serves to implement the comparison "$|n_{act}| < |n_x|$" independent of the direction of rotation. This function is similar to function 3. However, the amount is generated by the input signals before signal processing (without sign).

Function 3: dnIn1 < dnIn2

19.1.32 L_ComparePhi_2

This FB compares two angle signals.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

Parameters

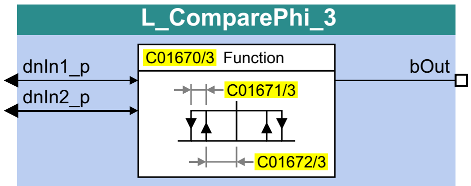
Parameters	Possible settings			Information
C01670/2				Function selection
	1	dnIn1 = dnIn2		
	2	dnIn1 > dnIn2		
	3	dnIn1 < dnIn2		
	4	|dnIn1| = |dnIn2|		
	5	|dnIn1| > |dnIn2|		
	6	|dnIn1| < |dnIn2|		
C01671/2	0	Incr.	1073741824	Hysteresis • Lenze setting: 0 incr.
C01672/2	0	Incr.	1073741824	Window • Lenze setting: 0 incr.

For a detailed functional description see L_ComparePhi_1.

19.1.33 L_ComparePhi_3

This FB compares two angle signals.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

Parameters

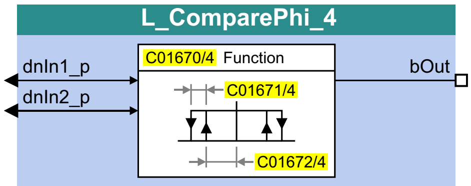
Parameters	Possible settings			Information
C01670/3				Function selection
	1	dnIn1 = dnIn2		
	2	dnIn1 > dnIn2		
	3	dnIn1 < dnIn2		
	4	|dnIn1| = |dnIn2|		
	5	|dnIn1| > |dnIn2|		
	6	|dnIn1| < |dnIn2|		
C01671/3	0	Incr.	1073741824	Hysteresis • Lenze setting: 0 incr.
C01672/3	0	Incr.	1073741824	Window • Lenze setting: 0 incr.

For a detailed functional description see L_ComparePhi_1.

19.1.34 L_ComparePhi_4

This FB compares two angle signals.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

Parameters

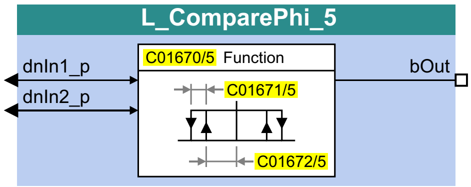
Parameters	Possible settings			Information
C01670/4				Function selection
	1	dnIn1 = dnIn2		
	2	dnIn1 > dnIn2		
	3	dnIn1 < dnIn2		
	4	|dnIn1| = |dnIn2|		
	5	|dnIn1| > |dnIn2|		
	6	|dnIn1| < |dnIn2|		
C01671/4	0	Incr.	1073741824	Hysteresis • Lenze setting: 0 incr.
C01672/4	0	Incr.	1073741824	Window • Lenze setting: 0 incr.

For a detailed functional description see L_ComparePhi_1.

19.1.35 L_ComparePhi_5

This FB compares two angle signals.

- Comparison operation, hysteresis and window size can be parameterised.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2

outputs

Designator Data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

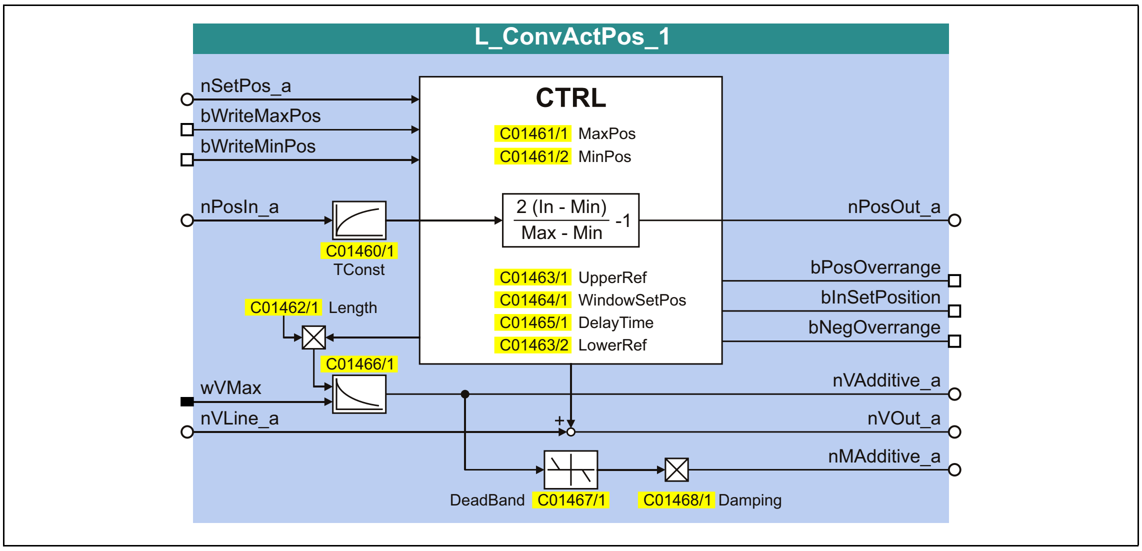
Parameters

Parameters	Possible settings			Information
C01670/5				Function selection
	1	dnIn1 = dnIn2		
	2	dnIn1 > dnIn2		
	3	dnIn1 < dnIn2		
	4	|dnIn1| = |dnIn2|		
	5	|dnIn1| > |dnIn2|		
	6	|dnIn1| < |dnIn2|		
C01671/5	0	Incr.	1073741824	Hysteresis • Lenze setting: 0 incr.
C01672/5	0	Incr.	1073741824	Window • Lenze setting: 0 incr.

For a detailed functional description see L_ComparePhi_1.

19.1.36 L_ConvActPos_1

This FB serves to filter an actual dancer position value and scale it to the available setting range. The setting range is defined by parameterisable limit positions which can optionally be accepted automatically by a "teach-in" function.



inputs

Designator Data type	Information/possible settings
nSetPos_a INT	Current position setpoint • Scaling: 16384 ≡ 100 %
bWriteMaxPos BOOL	"Teach-In" function for the upper limit position TRUE: Actual position is saved as upper limit position in C01461/1.
bWriteMinPos BOOL	"Teach-In" function for the lower limit position TRUE: Actual position is saved as lower limit position in C01461/2.
nPosIn_a INT	Actual position • Scaling: 16384 ≡ 100 %
wVMax WORD	Maximum material speed in [0.1 m/min] • Scaling: 2500 ≡ 250.0 m/min • Internal limitation to 3000 m/min
nVLine_a INT	Current path velocity • Scaling: 16384 ≡ Maximum material speed (*wVMax*)

outputs

Designator Data type	Value/meaning
nPosOut_a INT	Converted actual position value • Scaling: 16384 ≡ 100 % ≡ upper limit position (C01461/1) -16384 ≡ -100 % ≡ lower limit position (C01461/2)
bPosOverrange BOOL	Monitoring of upper home position TRUE: Upper home position (C01463/1) reached.
bInSetPosition BOOL	Monitoring of setpoint TRUE: Setpoint *nSetPos_a* reached.

Designator Data type	Value/meaning	
bNegOverrange BOOL	Monitoring of lower home position	
	TRUE	Lower home position (C01463/2) reached.
nVAdditive_a INT	Material speed caused by the dancer motion • Scaling: 16384 ≡ Maximum material speed (*wVMax*)	
nVOut_a INT	Current circumferential speed of the reel when the dancer hub is entered • Scaling: 16384 ≡ Maximum material speed (*wVMax*)	
nMAdditive_a INT	Additional torque for damping the dancer position control • Scaling: 16384 ≡ 100 %	

Parameters

Parameters	Possible settings			Information
C01460/1	0	ms	1000	TConst • Time constant for filtering the actual position value. • Lenze setting: 10 ms
C01461/1		%		Upper limit position • Lenze setting: 100.00 %
C01461/2		%		Lower limit position • Lenze setting: -100.00 %
C01462/1		mm		dwLength • Storage volumen of the dancer (dancer hub). • Lenze setting: 0 mm
C01463/1		%		Upper home position • Lenze setting: 90.00 %
C01463/2		%		Lower home position • Lenze setting: -90.00 %
C01464/1		%		nWindowSetPos • Tolerance zone for monitoring the dancer position. • Lenze setting: 10.00 %
C01465/1		s		wDelayTime • Delay time for monitoring the dancer position. • Lenze setting: 0.100 s
C01466/1		ms		wTConstVAdd • Time constant for filtering the additional speed. • Lenze setting: 10 ms
C01467/1		%		nDeadBand • Dead band area for feedforward control torque (*nMAdditive_a*). • Lenze setting: 1.00 %
C01468/1				wDamping • Damping factor for feedforward control torque (*nMAdditive_a*). • Lenze setting: 0.00

19.1.36.1 Determining limit positions

Option 1: Parameterising the limit positions

- Enter the actual value where the material stored in the dancer reaches its minimum length as upper limit position in C01461/1.
- Enterh the actual value where the material stored in the dancer reaches its maximum length as lower limit position in C01461/2.
- If the dancer detection provides an inverse signal, the lower limit position can be greater than the upper limit position.

Option 2: Defining the limit positions using the "teach-in" function

Alternatively to the direct setting of the limit values in the two parameters mentioned above, it is possible to bring the dancer into the respective limit position and then transfer the current value into the corresponding parameter by setting *bWriteMaxPos* bzw. *bWriteMinPos* to TRUE ("teach-in" function).

19.1.36.2 Monitoring of the dancer position

When the actual position value *nPosIn_a* reaches the current position setpoint *nSetPos_a*, the *bInSetPosition* output is set to TRUE.

- A tolerance zone can be set in C01464/1 for reaching the setpoint. when the actual dancer value leaves this tolerance zone, the *bInSetPosition* output will only be reset to FALSE after the delay time set in C01465/1 has elapsed (switch off delayed).
- In C01463/1 and C01463/2, reference values for the limit position monitoring can be set. If the actual dancer value reaches the set reference value towards the limit position, the corresponding (*bPosOverrange* output or *bNegOverrange*) output is set to TRUE (switch-off delayed) after the delay time set in C01465/1 has elapsed.

19.1.36.3 Circumferential speed of the reel

If the storage of the dancer is set in C01462/1, the additional speed of the material resulting from the motion of the dancer is provided at the *nVAdditive_a* output. When being added to the *nVLine_a* material speed, it results in the current circumferential speed of the reel via the *nVOut_a* output.

- The storage results from e.g. twice the distance between the two limit positions multiplied by the number of material wraps.
- The line speed is determined via *wVMax* and *nVLine_a*.
- In order to suppress the natural "noise" of the actual value signal with regard to the effect on the additional speed, the additional speed passes through a filter. The time constant of the filter can be set in C01466/1.

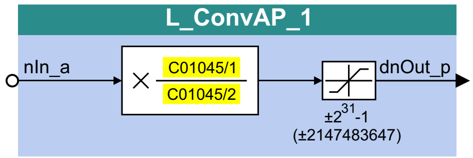
19.1.36.4 Additional torque for damping the dancer control

If the storage of the dancer is set in C01462/1, an additional feedforward control torque *nMAdditive_a* can be generated for damping the dancer control.

- The feedforward control torque is derived from the negated additional speed (double dancer speed) evaluated with the damping factor set in C01468/1.
- In order to suppress the natural "noise" of the actual value signal with regard to the effect on the additional torque, the additional speed passes through a dead band component in addition to the filter. The dead band area can be set in C01467/1.

19.1.37 L_ConvAP_1

This FB converts an analog value into a position.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input value • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Parameters

Parameters	Possible settings			Information
C01045/1	-32767		32767	Meters • Lenze setting: 1
C01045/2	-32767		32767	Denominator • Lenze setting: 1

Function

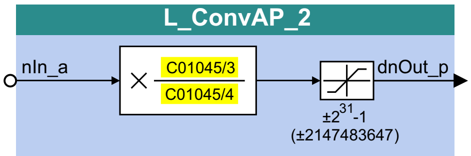
Note!

Conversion is remainder considered.

$$\text{dnOut\_p} = \text{nIn\_a} \cdot \frac{\text{C01045/1}}{\text{C01045/2}}$$

19.1.38 L_ConvAP_2

This FB converts an analog value into a position.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input value • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Parameters

Parameters	Possible settings			Information
C01045/3	-32767		32767	Meters • Lenze setting: 1
C01045/4	-32767		32767	Denominator • Lenze setting: 1

Function

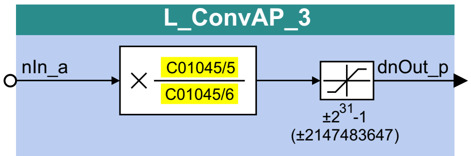
Note!

Conversion is remainder considered.

$$dnOut\_p = nIn\_a \cdot \frac{C01045/3}{C01045/4}$$

19.1.39 L_ConvAP_3

This FB converts an analog value into a position.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input value • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Parameters

Parameters	Possible settings			Information
C01045/5	-32767		32767	Meters • Lenze setting: 1
C01045/6	-32767		32767	Denominator • Lenze setting: 1

Function

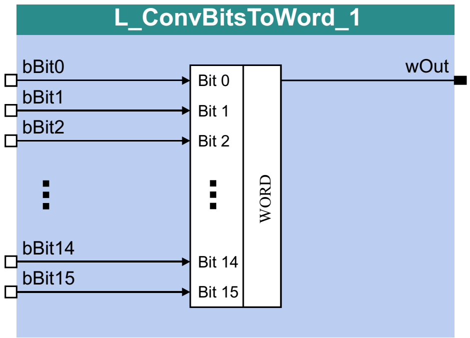
Note!

Conversion is remainder considered.

$$\text{dnOut\_p} = \text{nIn\_a} \cdot \frac{\text{C01045/5}}{\text{C01045/6}}$$

19.1.40 L_ConvBitsToWord_1

This FB converts 16 bit input values of the type "BOOL" into an output value of the type "WORD".



inputs

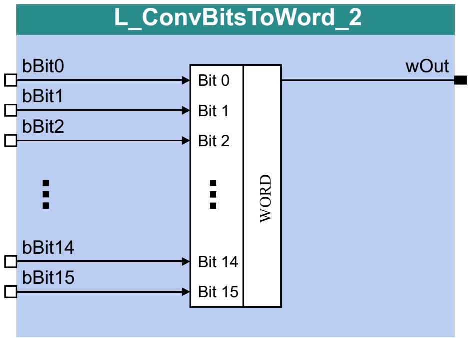
Designator Data type	Information/possible settings
bBit0 ... bBit15 BOOL	Input signal

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

19.1.41 L_ConvBitsToWord_2

This FB converts 16 bit input values of the type "BOOL" into an output value of the type "WORD".



inputs

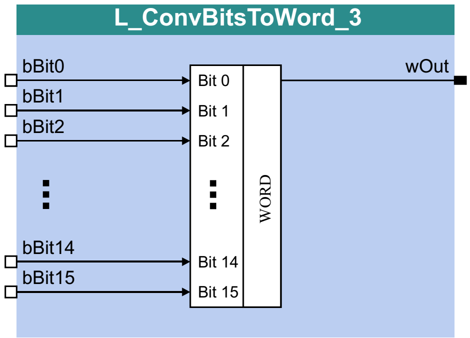
Designator Data type	Information/possible settings
bBit0 ... bBit15 BOOL	Input signal

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

19.1.42 L_ConvBitsToWord_3

This FB converts 16 bit input values of the type "BOOL" into an output value of the type "WORD".



inputs

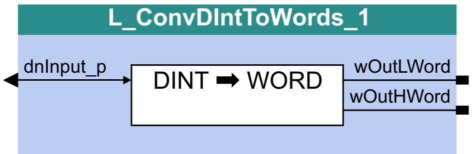
Designator Data type	Information/possible settings
bBit0 ... bBit15 BOOL	Input signal

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

19.1.43 L_ConvDIntToWords_1

This FB converts an input value of the type "DINT" into two output values of the type "WORD".



inputs

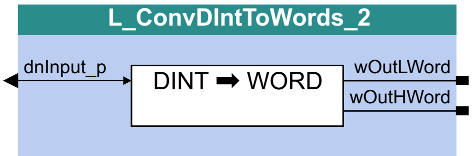
Designator Data type	Information/possible settings
dnInput_p DINT	Input signal

outputs

Designator Data type	Value/meaning
wOutLWord WORD	Output signal Low Word
wOutHWord WORD	Output signal High Word

19.1.44 L_ConvDIntToWords_2

This FB converts an input value of the type "DINT" into two output values of the type "WORD".



inputs

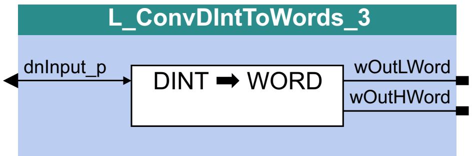
Designator Data type	Information/possible settings
dnInput_p DINT	Input signal

outputs

Designator Data type	Value/meaning
wOutLWord WORD	Output signal Low Word
wOutHWord WORD	Output signal High Word

19.1.45 L_ConvDIntToWords_3

This FB converts an input value of the type "DINT" into two output values of the type "WORD".



inputs

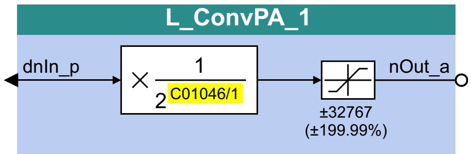
Designator Data type	Information/possible settings
dnInput_p DINT	Input signal

outputs

Designator Data type	Value/meaning
wOutLWord WORD	Output signal Low Word
wOutHWord WORD	Output signal High Word

19.1.46 L_ConvPA_1

This FB converts a position into an analog value.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Parameters

Parameters	Possible settings			Information
C01046/1	0		31	Division factor • Lenze setting: 1

Function

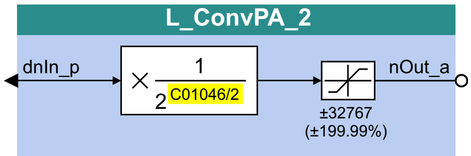
Note!

Conversion is remainder considered.

$$nOut\_a = dnIn\_p \cdot \frac{1}{2^{C01046/1}}$$

19.1.47 L_ConvPA_2

This FB converts a position into an analog value.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Parameters

Parameters	Possible settings			Information
C01046/2	0		31	Division factor • Lenze setting: 1

Function

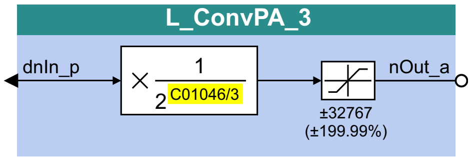
Note!

Conversion is remainder considered.

$$nOut\_a = dnIn\_p \cdot \frac{1}{2^{C01046/2}}$$

19.1.48 L_ConvPA_3

This FB converts a position into an analog value.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Parameters

Parameters	Possible settings			Information
C01046/3	0		31	Division factor • Lenze setting: 1

Function

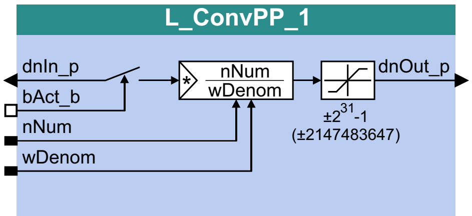
Note!

Conversion is remainder considered.

$$nOut\_a = dnIn\_p \cdot \frac{1}{2^{C01046/3}}$$

19.1.49 L_ConvPP_1

This FB converts a position with dynamic fraction.



inputs

Designator / Data type	Information/possible settings
dnIn_p (DINT)	Input signal
bAct_b (BOOL)	Conversion mode FALSE: $dnOut\_p = Remainder \cdot \frac{nNum}{wDenom}$ TRUE: $dnOut\_p = dnIn\_p \cdot \frac{nNum}{wDenom}$
nNum (INT)	Factor (numerator) • Internal limitation to -32767 ... -1 / 1 ... 32767
wDenom (WORD)	Factor (denominator) • Internal limitation to 1 ... 32767

outputs

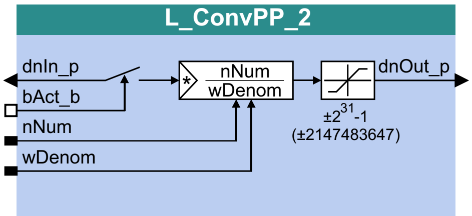
Designator / Data type	Value/meaning
dnOut_p (DINT)	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Note!

Conversion is remainder considered.

19.1.50 L_ConvPP_2

This FB converts a position with dynamic fraction.



inputs

Designator / Data type	Information/possible settings	
dnIn_p DINT	Input signal	
bAct_b BOOL	Conversion mode	
	FALSE	$dnOut\_p = Remainder \cdot \frac{nNum}{wDenom}$
	TRUE	$dnOut\_p = dnIn\_p \cdot \frac{nNum}{wDenom}$
nNum INT	Factor (numerator) • Internal limitation to -32767 ... -1 / 1 ... 32767	
wDenom WORD	Factor (denominator) • Internal limitation to 1 ... 32767	

outputs

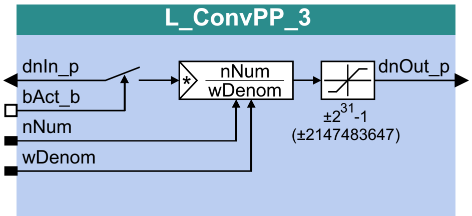
Designator / Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Note!

Conversion is remainder considered.

19.1.51 L_ConvPP_3

This FB converts a position with dynamic fraction.



inputs

Designator Data type	Information/possible settings	
dnIn_p DINT	Input signal	
bAct_b BOOL	Conversion mode	
	FALSE	$\text{dnOut\_p} = \text{Remainder} \cdot \frac{\text{nNum}}{\text{wDenom}}$
	TRUE	$\text{dnOut\_p} = \text{dnIn\_p} \cdot \frac{\text{nNum}}{\text{wDenom}}$
nNum INT	Factor (numerator) • Internal limitation to -32767 ... -1 / 1 ... 32767	
wDenom WORD	Factor (denominator) • Internal limitation to 1 ... 32767	

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

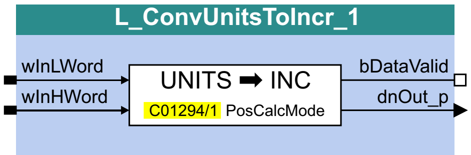
Note!

Conversion is remainder considered.

19.1.52 L_ConvUnitsToIncr_1

Taking into account the machine parameters, this FB converts a position value provided in the real unit of the machine into an internal 32-bit position value.

- The conversion mode has to be set in C01294/1. In the Lenze setting, no conversion takes place due to compatibility reasons.
- Conversion and provision of the result to *dnOut_p* do not take place in real time! Hence, the *bDataValid* output signalises when the conversion has been completed and the *dnOut_p* output value is consistent for transfer to following processes.



inputs

Designator Data type	Information/possible settings
wInLWord / wInHWord WORD	Input signal Low Word / High Word

outputs

Designator Data type	Value/meaning	
bDataValid BOOL	FALSE	Conversion is active, *dnOut_p* is not valid.
	TRUE	Conversion is completed, *dnOut_p* is valid (consistent).
dnOut_p DINT	Result of conversion in [increments]	

Parameters

Parameters	Possible settings		Information
C01294/1			Mode: Position calculation
	0	dnOut_p=HW+LW	No conversion (Lenze setting) **Note!** If a cycle length is set in C01201/1, a calculation is made under the following conditions: • Position specification is higher than or equal to the cycle length. • Position specification is negative. For a correct positioning process, wait until *bDataValid* = TRUE.
	1	16 bits: LW=+/-32767	*wInLWord* = ±32767 [Unit]
	2	16 bits: HW=+/-; LW=0..65535	*wInLWord* = 0 ... 65535 [Unit] *wInHWord* = sign (0 = positive; ≠0 = negative)
	3	32 bits: HW_LW=+/-214748_3647	*wInHWord* and *wInLWord* = ±214748.3647 [Unit]

Conversion formula for modes 1 ... 3

$$\text{dnOut\_p [incr.]} = \text{Input value [Unit]} \cdot \frac{\text{C01202/1}}{\text{C01202/2}} \cdot \frac{\text{C01203/2}}{\text{C01203/1}} \cdot \frac{65536\,\text{[Incr./rev.]}}{\text{C01204}}$$

C01202/1: Gearbox factor (numerator)
C01202/2: Gearbox factor (denominator)
C01203/1: Encoder gearbox factor (numerator)
C01203/2: Encoder gearbox factor (denominator)
C01204: Feed constant

[19-18] Conversion formula for [Unit] ➞ [increments]

Example

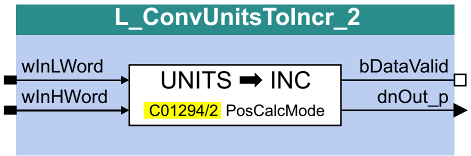
- All gearbox factors = 1
- Feed constant = 360°/revolution

<table>
<tr><th colspan="2">Conversion mode (C01294)</th><th>wInHWord</th><th>wInLWord</th><th>Input value [Unit]</th><th>dnOut_p [Increments</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">dnOut_p=HW+LW
(no conversion)</td><td colspan="2">32-bit input value*</td><td rowspan="2">-32000</td><td rowspan="2">-32000</td></tr>
<tr><td>65535</td><td>33536</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">16 bits: LW=+/-32767</td><td>No meaning</td><td>16-bit input value*</td><td rowspan="2">-32000</td><td rowspan="2">-5825422</td></tr>
<tr><td>65535</td><td>33536</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">16 bits: HW=+/-; LW=0..65535</td><td>Sign</td><td>16-bit input value</td><td rowspan="2">-33536</td><td rowspan="2">-6105042</td></tr>
<tr><td>65535 ≡ negative</td><td>33536</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">32 bits: HW_LW=+/-214748_3647</td><td colspan="2">32-bit input value**</td><td rowspan="2">-3.2000</td><td rowspan="2">-582</td></tr>
<tr><td>65535</td><td>33536</td></tr>
<tr><td colspan="6">* Two's complement   ** Two's complement with four decimal positions</td></tr>
</table>

19.1.53 L_ConvUnitsToIncr_2

Taking into account the machine parameters, this FB converts a position value provided in the real unit of the machine into an internal 32-bit position value.

- The conversion mode has to be set in C01294/2. In the Lenze setting, no conversion takes place due to compatibility reasons.
- Conversion and provision of the result to *dnOut_p* do not take place in real time! Hence, the *bDataValid* output signalises when the conversion has been completed and the *dnOut_p* output value is consistent for transfer to following processes.



inputs

Designator Data type	Information/possible settings
wInLWord / wInHWord WORD	Input signal Low Word / High Word

outputs

Designator Data type	Value/meaning	
bDataValid BOOL	FALSE	Conversion is active, *dnOut_p* is not valid.
	TRUE	Conversion is completed, *dnOut_p* is valid (consistent).
dnOut_p DINT	Result of conversion in [increments]	

Parameters

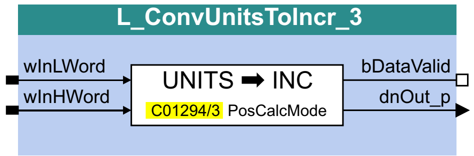
Parameters	Possible settings		Information
C01294/2			Mode: Position calculation
	0	dnOut_p=HW+LW	No conversion (Lenze setting) **Note!** If a cycle length is set in C01201/1, a calculation is made under the following conditions: • Position specification is higher than or equal to the cycle length. • Position specification is negative. For a correct positioning process, wait until *bDataValid* = TRUE.
	1	16 bits: LW=+/-32767	*wInLWord* = ±32767 [Unit]
	2	16 bits: HW=+/-; LW=0..65535	*wInLWord* = 0 ... 65535 [Unit] *wInHWord* = sign (0 = positive; ≠0 = negative)
	3	32 bits: HW_LW=+/-214748_3647	*wInHWord* and *wInLWord* = ±214748.3647 [Unit]

For a detailed functional description see L_ConvUnitsToIncr_1.

19.1.54 L_ConvUnitsToIncr_3

Taking into account the machine parameters, this FB converts a position value provided in the real unit of the machine into an internal 32-bit position value.

- The conversion mode has to be set in C01294/3. In the Lenze setting, no conversion takes place due to compatibility reasons.
- Conversion and provision of the result to *dnOut_p* do not take place in real time! Hence, the *bDataValid* output signalises when the conversion has been completed and the *dnOut_p* output value is consistent for transfer to following processes.



inputs

Designator Data type	Information/possible settings
wInLWord / wInHWord WORD	Input signal Low Word / High Word

outputs

Designator Data type	Value/meaning	
bDataValid BOOL	FALSE	Conversion is active, *dnOut_p* is not valid.
	TRUE	Conversion is completed, *dnOut_p* is valid (consistent).
dnOut_p DINT	Result of conversion in [increments]	

Parameters

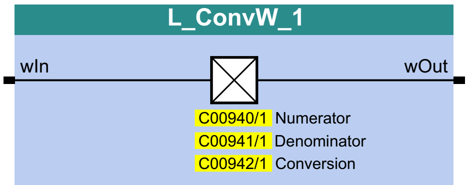
Parameters	Possible settings		Information
C01294/3			Mode: Position calculation
	0	dnOut_p=HW+LW	No conversion (Lenze setting) **Note!** If a cycle length is set in C01201/1, a calculation is made under the following conditions: • Position specification is higher than or equal to the cycle length. • Position specification is negative. For a correct positioning process, wait until *bDataValid* = TRUE.
	1	16 bits: LW=+/-32767	*wInLWord* = ±32767 [Unit]
	2	16 bits: HW=+/-; LW=0..65535	*wInLWord* = 0 ... 65535 [Unit] *wInHWord* = sign (0 = positive; ≠0 = negative)
	3	32 bits: HW_LW=+/-214748_3647	*wInHWord* and *wInLWord* = ±214748.3647 [Unit]

For a detailed functional description see L_ConvUnitsToIncr_1.

19.1.55 L_ConvW_1

This FB serves to convert analog signal forms. The following conversions per parameter can be selected:

- Input signal is passed through without conversion
- [%] → [incr/ms]
- [incr/ms] → [%]
- Conversion via parameterisable factors



inputs

Designator Data type	Information/possible settings
wIn WORD	Input signal

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

Parameters

Parameters	Possible settings			Information
C00940/1	-32767		32767	Numerator • Lenze setting: 1
C00941/1	-32767		32767	Denominator • Lenze setting: 1
C00942/1	0	*wOut* = *wIn* (no conversion)		Selection of the conversion
	1	[%] → [incr/ms]		
	2	[incr/ms] → [%]		
	3	*wOut* = *wIn[signed]* * C00940 / C00941		
	4	*wOut* = *wIn[unsigned]* * C00940 / C00941		

Conversion formula for selection 1

Note!

Division is remainder considered.

$$\text{wOut [incr/ms]} = \frac{\text{wIn [\%]} \cdot \text{C00011 [rpm]} \cdot \text{65536 [incr/rev.]}}{\text{100 \%} \cdot \text{60 [s/min]} \cdot \text{1000 [ms/s]}}$$

[19-19] Conversion formula for selection 1: [%] ➞ [incr/ms]

Conversion formula for selection 2

$$\text{wOut [\%]} = \frac{\text{wIn [incr/ms]} \cdot \text{100 \%} \cdot \text{60 [s/min]} \cdot \text{1000 [ms/s]}}{\text{C00011 [rpm]} \cdot \text{65536 [incr/rev.]}}$$

[19-20] Conversion formula for selection 2: [incr/ms] ➞ [%]

Conversion formula for selection 3

Note!

In the internal processing, the double number display range is taken into account.

$$\mathrm{wOut} = \mathrm{wIn} \cdot \frac{\mathrm{C00940}}{\mathrm{C00941}}$$

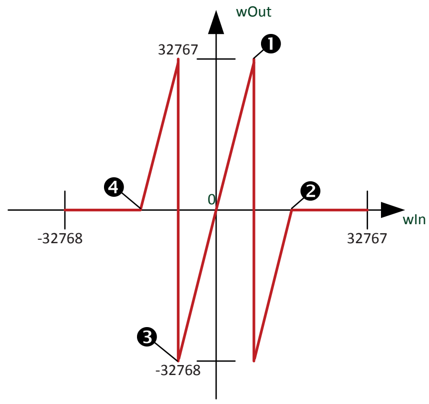
[19-21] Conversion formula for selection 3: Parameterisable factors with the evaluation of the *wIn* input variable as signed value

Use this setting to scale analog signals (e.g. *L_MPot_1.nOut_a* // output variable of the motor potentiometer).

Note!

In the internal processing, the double number display range is taken into account. This causes a one-time overflow when the analog display area is reached for the first time.

- In order to avoid this effect, you can use the e.g. the L_OffsetGain_x function block for scaling analog values.



Selected numerator: C00940 = 4
Selected denominator: C00941 = 1
❶ Overflow of positive number range
❷ Limitation to -1
❸ Overflow of negative number range
❹ Limitation to -1

[19-22] Selection 3: Example

Conversion formula for selection 4

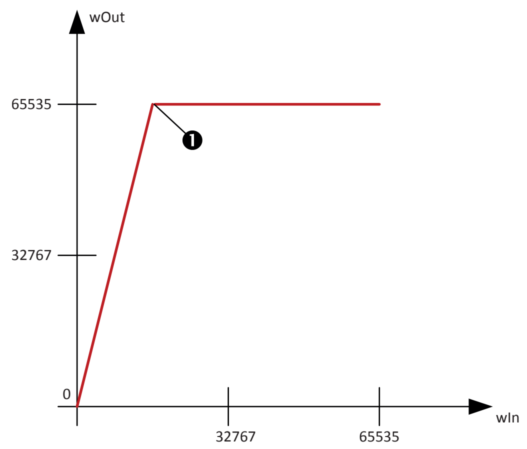
$$\mathrm{wOut} = \mathrm{wIn} \cdot \frac{\mathrm{C00940}}{\mathrm{C00941}}$$

[19-23] Conversion formula for selection 4: Parameterisable factors with the evaluation of the *wIn* input variable as unsigned value

Use this setting to scale unsigned signals (e.g. *LS_ParFree.wC471_1* // output variable of the free parameter 471/1).

Note!

The output value is limited to the number range and no overflow takes place.



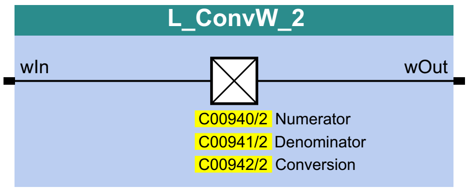
Selected numerator: C00940 = 4
Selected denominator: C00941 = 1
❶ Limitation to -65535

[19-24] Selection 4: Example

19.1.56 L_ConvW_2

This FB serves to convert analog signal forms. The following conversions per parameter can be selected:

- Input signal is passed through without conversion
- [%] ➞ [incr/ms]
- [incr/ms] ➞ [%]
- Conversion via parameterisable factors



inputs

Designator Data type	Information/possible settings
wIn WORD	Input signal

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

Parameters

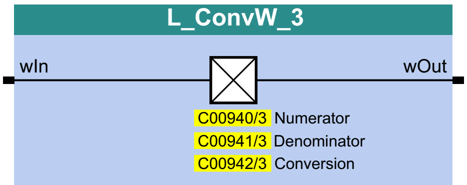
Parameters	Possible settings			Information
C00940/2	-32767		32767	Numerator • Lenze setting: 1
C00941/2	-32767		32767	Denominator • Lenze setting: 1
C00942/2	0	*wOut* = *wIn* (no conversion)		Selection of the conversion
	1	[%] ➞ [incr/ms]		
	2	[incr/ms] ➞ [%]		
	3	*wOut* = *wIn[signed]* * C00940 / C00941		
	4	*wOut* = *wIn[unsigned]* * C00940 / C00941		

For conversion formulae see L_ConvW_1.

19.1.57 L_ConvW_3

This FB serves to convert analog signal forms. The following conversions per parameter can be selected:

- Input signal is passed through without conversion
- [%] ➞ [incr/ms]
- [incr/ms] ➞ [%]
- Conversion via parameterisable factors



inputs

Designator Data type	Information/possible settings
wIn WORD	Input signal

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

Parameters

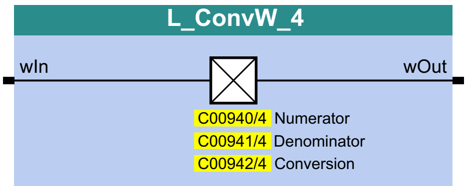
Parameters	Possible settings			Information
C00940/3	-32767		32767	Numerator • Lenze setting: 1
C00941/3	-32767		32767	Denominator • Lenze setting: 1
C00942/3	0	*wOut* = *wIn* (no conversion)		Selection of the conversion
	1	[%] ➞ [incr/ms]		
	2	[incr/ms] ➞ [%]		
	3	*wOut* = *wIn[signed]* * C00940 / C00941		
	4	*wOut* = *wIn[unsigned]* * C00940 / C00941		

For conversion formulae see L_ConvW_1.

19.1.58 L_ConvW_4

This FB serves to convert analog signal forms. The following conversions per parameter can be selected:

- Input signal is passed through without conversion
- [%] → [incr/ms]
- [incr/ms] → [%]
- Conversion via parameterisable factors



inputs

Designator Data type	Information/possible settings
wIn WORD	Input signal

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

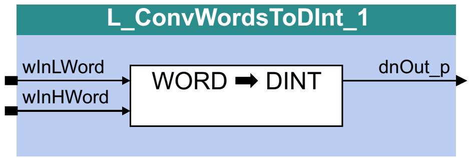
Parameters

Parameters	Possible settings			Information
C00940/4	-32767		32767	Numerator • Lenze setting: 1
C00941/4	-32767		32767	Denominator • Lenze setting: 1
C00942/4	0	*wOut = wIn* (no conversion)		Selection of the conversion
	1	[%] → [incr/ms]		
	2	[incr/ms] → [%]		
	3	*wOut = wIn[signed] ** C00940 / C00941		
	4	*wOut = wIn[unsigned] ** C00940 / C00941		

For conversion formulae see L_ConvW_1.

19.1.59 L_ConvWordsToDInt_1

This FB converts two input values of the type "WORD" into one output value of the type "DINT".



inputs

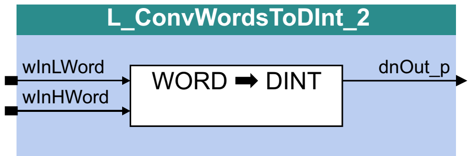
Designator Data type	Information/possible settings
wInLWord WORD	Input signal Low Word
wInHWord WORD	Input signal High Word

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal

19.1.60 L_ConvWordsToDInt_2

This FB converts two input values of the type "WORD" into one output value of the type "DINT".



inputs

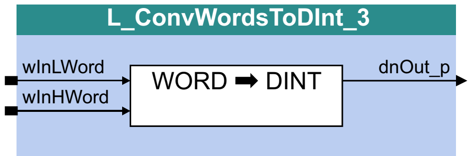
Designator Data type	Information/possible settings
wInLWord WORD	Input signal Low Word
wInHWord WORD	Input signal High Word

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal

19.1.61 L_ConvWordsToDInt_3

This FB converts two input values of the type "WORD" into one output value of the type "DINT".



inputs

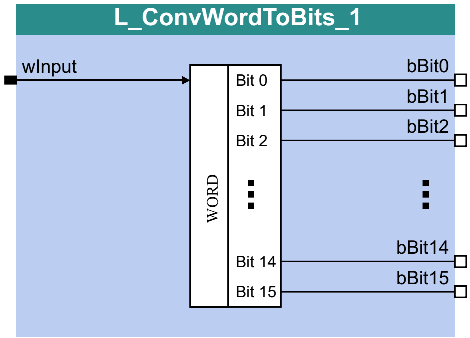
Designator Data type	Information/possible settings
wInLWord WORD	Input signal Low Word
wInHWord WORD	Input signal High Word

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal

19.1.62 L_ConvWordToBits_1

This FB converts an input value of "WORD" type into 16 individual binary signals.



inputs

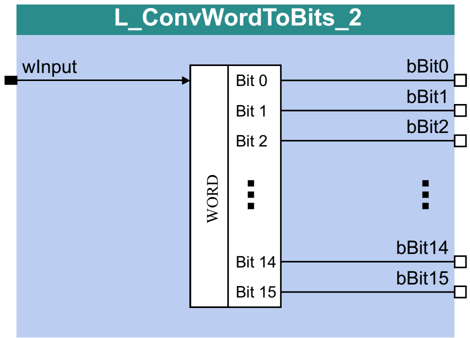
Designator Data type	Information/possible settings
wInput WORD	Input signal

outputs

Designator Data type	Value/meaning
bBit0 ... bBit15 BOOL	Output signal

19.1.63 L_ConvWordToBits_2

This FB converts an input value of "WORD" type into 16 individual binary signals.



inputs

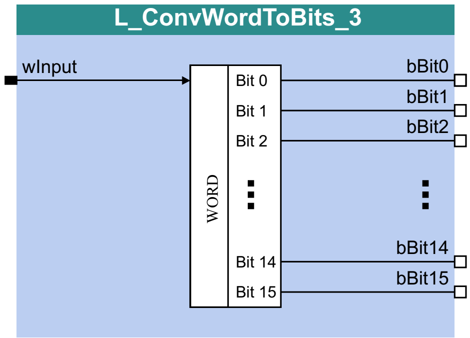
Designator Data type	Information/possible settings
wInput WORD	Input signal

outputs

Designator Data type	Value/meaning
bBit0 ... bBit15 BOOL	Output signal

19.1.64 L_ConvWordToBits_3

This FB converts an input value of "WORD" type into 16 individual binary signals.



inputs

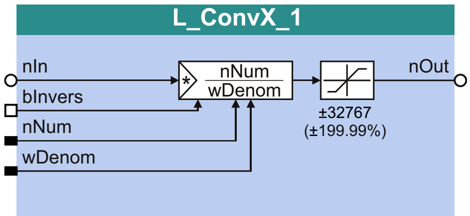
Designator Data type	Information/possible settings
wInput WORD	Input signal

outputs

Designator Data type	Value/meaning
bBit0 ... bBit15 BOOL	Output signal

19.1.65 L_ConvX_1

This FB scales an analog value.



inputs

Designator Data type	Information/possible settings
nIn INT	Input signal
bInvers BOOL	Invert sign of the output signal FALSE: $\text{nOut} = \text{nIn} \cdot \frac{\text{nNum}}{\text{wDenom}}$ TRUE: $\text{nOut} = \text{nIn} \cdot \frac{\text{nNum}}{\text{wDenom}} \cdot (-1)$
nNum INT	Factor (numerator) • Internal limitation to -32767 ... -1 / 1 ... 32767
wDenom WORD	Factor (denominator) • Internal limitation to 1 ... 32767

outputs

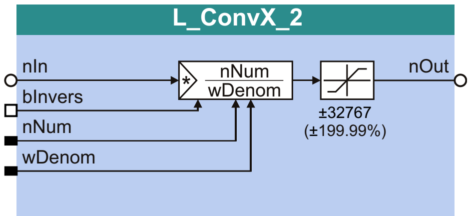
Designator Data type	Value/meaning
nOut INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Note!

Conversion is remainder considered.

19.1.66 L_ConvX_2

This FB scales an analog value.



inputs

Designator Data type	Information/possible settings	
nIn INT	Input signal	
bInvers BOOL	Invert sign of the output signal	
	FALSE	$\mathrm{nOut} = \mathrm{nIn} \cdot \frac{\mathrm{nNum}}{\mathrm{wDenom}}$
	TRUE	$\mathrm{nOut} = \mathrm{nIn} \cdot \frac{\mathrm{nNum}}{\mathrm{wDenom}} \cdot (-1)$
nNum INT	Factor (numerator) • Internal limitation to -32767 ... -1 / 1 ... 32767	
wDenom WORD	Factor (denominator) • Internal limitation to 1 ... 32767	

outputs

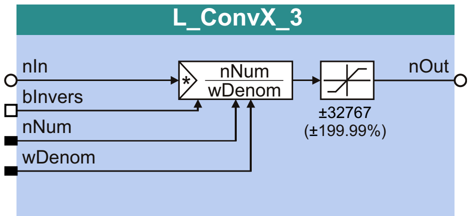
Designator Data type	Value/meaning
nOut INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Note!

Conversion is remainder considered.

19.1.67 L_ConvX_3

This FB scales an analog value.



inputs

Designator Data type	Information/possible settings
nIn INT	Input signal
bInvers BOOL	Invert sign of the output signal FALSE: $nOut = nIn \cdot \frac{nNum}{wDenom}$ TRUE: $nOut = nIn \cdot \frac{nNum}{wDenom} \cdot (-1)$
nNum INT	Factor (numerator) • Internal limitation to -32767 ... -1 / 1 ... 32767
wDenom WORD	Factor (denominator) • Internal limitation to 1 ... 32767

outputs

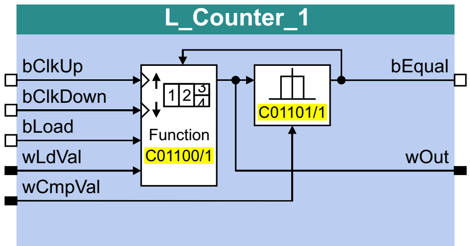
Designator Data type	Value/meaning
nOut INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Note!

Conversion is remainder considered.

19.1.68 L_Counter_1

This FB is a digital upcounter and downcounter with a parameterisable comparison operation.



inputs

Designator / Data type		Information/possible settings	
bClkUp	BOOL	Clock input • With each edge, the module counts up by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.	
bClkDown	BOOL	Clock input • With each edge, the module counts down by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.	
bLoad	BOOL	Load input • The input has the highest priority.	
		TRUE	Accept starting value *wLdVal*.
wLdVal	WORD	Starting value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.	
wCmpVal	WORD	Comparison value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.	

outputs

Designator / Data type		Value/meaning	
bEqual	BOOL	Status signal "Comparison statement is true" • The TRUE output is active in the Lenze setting if the current counter content is greater than or equal to the comparison value *wCmpVal*.	
		TRUE	The statement of the comparison mode selected in C01101/1 is true.
wOut	WORD	Counter content • Internal limitation to ± 32767 • The most significant bit determines the sign!	

Parameters

Parameters	Possible settings		Information
C01100/1			Function selection • Lenze setting: Normal counting
	0	Normal counting	
	1	Auto reset	
	2	Manual reset	

Parameters	Possible settings		Information
C01101/1			Selection of comparison operation • Lenze setting: Counter content ≥ comparison value
	0	Counter content ≥ comparison value	
	1	Counter content ≤ comparison value	
	2	Counter content = comparison value	

General function

- Every FALSE/TRUE edge at the *bClkUp* input causes the block to count upwards by "1".
- Every FALSE/TRUE edge at the *bClkDown* input causes the block to count downwards by "1".

Function "Normal counting"

If the statement of the comparison mode selected in C01101/1 is true, the *bCompare* output is set to TRUE.

Function "Auto reset"

If the statement of the comparison mode selected in C01101/1 is true, the *bCompare* output is set to TRUE for 1 ms and the counter is reset to the *wLdVal* starting value.

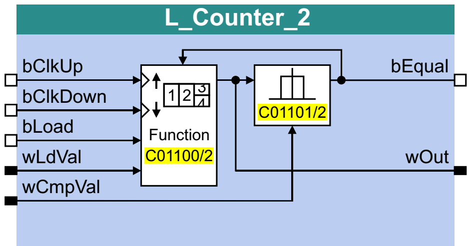
Function "Manual reset"

If the statement of the comparison mode selected in C01101/1 is true, the *bCompare* output is set to TRUE and the counter stops.

- Edges at *bClkUp* and *bClkDown* are ignored.
- The counter must be reset via the *bLoad* input.

19.1.69 L_Counter_2

This FB is a digital upcounter and downcounter with a parameterisable comparison operation.



inputs

Designator Data type	Information/possible settings	
bClkUp BOOL	Clock input • With each edge, the module counts up by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.	
bClkDown BOOL	Clock input • With each edge, the module counts down by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.	
bLoad BOOL	Load input • The input has the highest priority.	
	TRUE	Accept starting value *wLdVal*.
wLdVal WORD	Starting value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.	
wCmpVal WORD	Comparison value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.	

outputs

Designator Data type	Value/meaning	
bEqual BOOL	Status signal "Comparison statement is true" • The TRUE output is active in the Lenze setting if the current counter content is greater than or equal to the comparison value *wCmpVal*.	
	TRUE	The statement of the comparison mode selected in C01101/2 is true.
wOut WORD	Counter content • Internal limitation to ± 32767 • The most significant bit determines the sign!	

Parameters

Parameters	Possible settings		Information
C01100/2			Function selection • Lenze setting: Normal counting
	0	Normal counting	
	1	Auto reset	
	2	Manual reset	
C01101/2			Selection of comparison operation • Lenze setting: Counter content ≥ comparison value
	0	Counter content ≥ comparison value	
	1	Counter content ≤ comparison value	
	2	Counter content = comparison value	

General function

- Every FALSE/TRUE edge at the *bClkUp* input causes the block to count upwards by "1".
- Every FALSE/TRUE edge at the *bClkDown* input causes the block to count downwards by "1".

Function "Normal counting"

If the statement of the comparison mode selected in C01101/2 is true, the *bCompare* output is set to TRUE.

Function "Auto reset"

If the statement of the comparison mode selected in C01101/2 is true, the *bCompare* output is set to TRUE for 1 ms and the counter is reset to the *wLdVal* starting value.

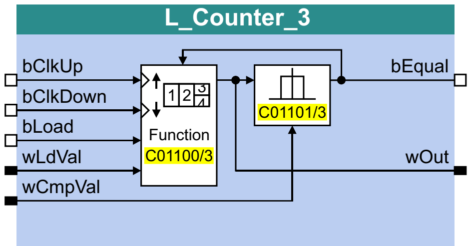
Function "Manual reset"

If the statement of the comparison mode selected in C01101/2 is true, the *bCompare* output is set to TRUE and the counter stops.

- Edges at *bClkUp* and *bClkDown* are ignored.
- The counter must be reset via the *bLoad* input.

19.1.70 L_Counter_3

This FB is a digital upcounter and downcounter with a parameterisable comparison operation.



inputs

Designator Data type	Information/possible settings
bClkUp BOOL	Clock input • With each edge, the module counts up by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.
bClkDown BOOL	Clock input • With each edge, the module counts down by "1". • Only FALSE-TRUE edges are evaluated. Note: The static state "1" is not permissible at this input.
bLoad BOOL	Load input • The input has the highest priority. TRUE: Accept starting value *wLdVal*.
wLdVal WORD	Starting value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.
wCmpVal WORD	Comparison value • Assigned value is internally interpreted as "INT" data type (-32767 ... +32767), i.e. the most significant bit determines the sign.

outputs

Designator Data type	Value/meaning
bEqual BOOL	Status signal "Comparison statement is true" • The TRUE output is active in the Lenze setting if the current counter content is greater than or equal to the comparison value *wCmpVal*. TRUE: The statement of the comparison mode selected in C01101/3 is true.
wOut WORD	Counter content • Internal limitation to ± 32767 • The most significant bit determines the sign!

Parameters

Parameters	Possible settings		Information
C01100/3			Function selection • Lenze setting: Normal counting
	0	Normal counting	
	1	Auto reset	
	2	Manual reset	
C01101/3			Selection of comparison operation • Lenze setting: Counter content ≥ comparison value
	0	Counter content ≥ comparison value	
	1	Counter content ≤ comparison value	
	2	Counter content = comparison value	

General function

- Every FALSE/TRUE edge at the *bClkUp* input causes the block to count upwards by "1".
- Every FALSE/TRUE edge at the *bClkDown* input causes the block to count downwards by "1".

Function "Normal counting"

If the statement of the comparison mode selected in C01101/3 is true, the *bCompare* output is set to TRUE.

Function "Auto reset"

If the statement of the comparison mode selected in C01101/3 is true, the *bCompare* output is set to TRUE for 1 ms and the counter is reset to the *wLdVal* starting value.

Function "Manual reset"

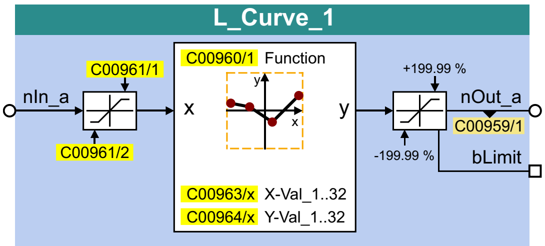
If the statement of the comparison mode selected in C01101/3 is true, the *bCompare* output is set to TRUE and the counter stops.

- Edges at *bClkUp* and *bClkDown* are ignored.
- The counter must be reset via the *bLoad* input.

19.1.71 L_Curve_1

Alternatively, this FB represents a characteristic curve or cam function y=f(x), whereby the input signal of the X-axis and the output signal of the Y-axis correspond.

- A limitation of the input value can be set in C00961/1 (upper limit) and C00961/2 (lower limit).
- The output value is internally limited to ±199.99 %. If a limitation of the output value is active, *bLimit* is set to TRUE.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input value (x) • ±16384 ≡ ±100 % • A limitation of the input value can be set in C00961/1 (upper limit) and C00961/2 (lower limit).

outputs

Designator Data type	Value/meaning	
nOut_a INT	Output value (y) • ±16384 ≡ ±100 % • Internal limitation to ±32767 (±199.99 %)	
bLimit BOOL	"Limitation active" status signal	
	TRUE	The output value is limited to ±199.99 %.

Parameters

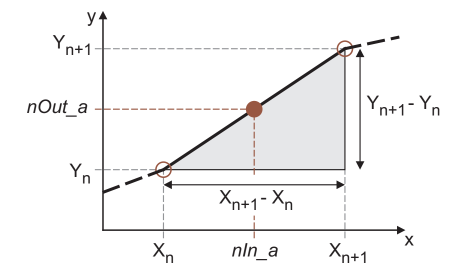
Parameters	Possible settings			Information
C00959/1 From version 02.00.00	-199.99	%	199.99	Output value • Read only
C00960/1				Function selection
	0	Out = 0		FB deactivated (Lenze setting)
	1	Out = In		
	2	Out = f(In)		In preparation!
	3	Out = f(table)		▸ Function 3: nOut_a = f(table)
C00961/1	-199.99	%	199.99	Upper limit for input value • Lenze setting: 199.99 %
C00961/2	-199.99	%	199.99	Lower limit for input value • Lenze setting: -199.99 %

Parameters	Possible settings			Information
C00963/1...32	-32767		32767	X values for characteristic function • Subcodes 1 ... 32 correspond to interpolation point values X1 ... X32. • ±16384 ≡ ±100 % • Lenze setting: 0 ▶ Function 3: nOut_a = f(table)
C00964/1...32	-32767		32767	Y values for characteristic function • Subcodes 1 ... 32 correspond to interpolation point values Y1 ... Y32. • ±16384 ≡ ±100 % • Lenze setting: 0 ▶ Function 3: nOut_a = f(table)

19.1.71.1 Function 3: nOut_a = f(table)

If function 3 has been selected in C00960, the output value is calculated according to a characteristic.

- The characteristic may comprise up to 32 interpolation points which are defined by parameters.
- If the *nIn_a* input value is equal to one of the X interpolation points, the corresponding Y interpolation point will be output to *nOut_a*.
- If the *nIn_a* input value lies between two X interpolation points, the *nOut_a* output value will be interpolated linearly:



As per the equation,

$$\frac{X_{n+1} - X_n}{Y_{n+1} - Y_n} = \frac{nIn\_a - X_n}{nOut\_a - Y_n}$$

output value *nOut_a* for *nIn_a* is calculated as follows:

$$nOut\_a = \frac{(nIn\_a - X_n) \cdot (Y_{n+1} - Y_n)}{X_{n+1} - X_n} + Y_n$$

[19-25] Linear interpolation between two interpolation points

Selection of the characteristic

The max. 32 interpolation points of the characteristic are selected via the subcodes of C00963 and C00964.

- The values are to be set as raw values (±16384 ≡ ±100 %).
- The same subcodes of C00963 and C00964 correspond to one pair of variates/interpolation point (x_n, y_n).

	1	2	...	32
X	C00963/**1**	C00963/**2**	C00963/...	C00963/**32**
Y	C00964/**1**	C00964/**2**	C00964/...	C00964/**32**

- The first pair of variates (C00963/1 and C00964/1) is always valid.
- The X values of the characteristic must be entered in ascending order (X1 < X2 < ... < X32).
- An interruption of the ascending order of the X values represents the end of the characteristic (in our example: X5 < X4):

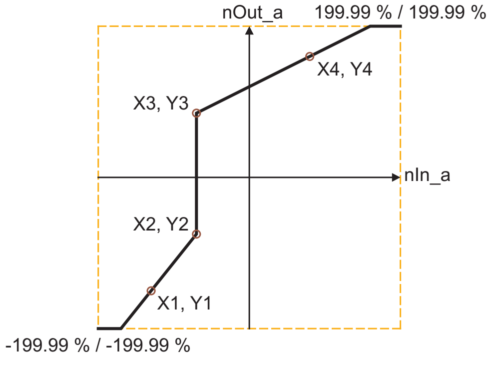
	1	2	3	4	5	6	7	...	32
X	-22938 (-140 %)	-14746 (-90 %)	11469 (70 %)	23757 (145 %)	0	0	0	0	0
Y	-22938 (-140 %)	-6554 (-40 %)	18842 (115 %)	26215 (160 %)	0	0	0	0	0

- Starting at the last valid pairs of variates, extrapolation is performed on both sides until the end of the X value range (±199.99 %).

Special cases

- Step changes/discontinuities can be displayed by using the same parameter setting for two consecutive X values. A step change is only valid if it has a valid pair of variates. Otherwise, extrapolation continues, starting at the last valid pair of variates.

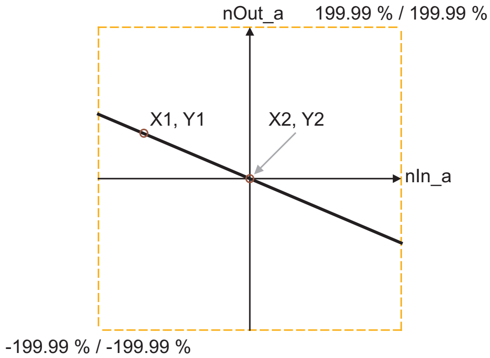
	1	2	3	4	5	6	7	...	32
X	-21300 (-130 %)	-11469 (-70 %)	-11469 (-70 %)	13107 (80 %)	0	0	0	0	0
Y	-24576 (-150 %)	-12288 (-75 %)	13927 (85 %)	26215 (160 %)	0	0	0	0	0



[19-26] Example: Characteristic with step change

- If only one interpolation point, X1, in the negative range is defined, X2 exists with the "0" Lenze setting being the valid interpolation point (X2 > X1), and a line is drawn through both interpolation points:

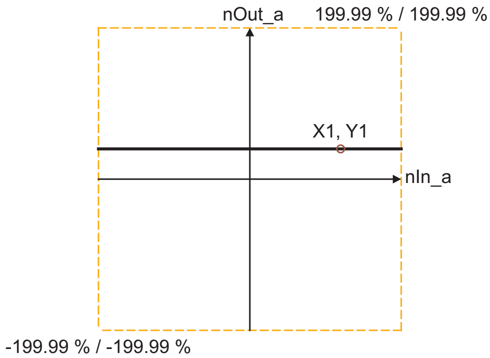
	1	2	3	4	5	6	7	...	32
X	-21300 (-130 %)	0 (0 %)	0	0	0	0	0	0	0
Y	6554 (40 %)	0 (0 %)	0	0	0	0	0	0	0



[19-27] Example: Characteristic with only one defined interpolation point and X1 < 0

- If only one interpolation point, X1, is defined in the positive range, a line is extrapolated:

	1	2	3	4	5	6	7	...	32
X	19661 (120 %)	0	0	0	0	0	0	0	0
Y	6554 (40 %)	0	0	0	0	0	0	0	0

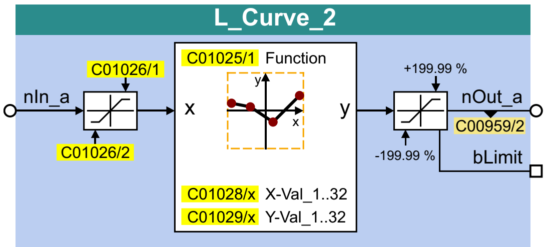


[19-28] Example: Characteristic with only one defined interpolation point and X1 > 0

19.1.72 L_Curve_2

Alternatively, this FB represents a characteristic curve or cam function y=f(x), whereby the input signal of the X-axis and the output signal of the Y-axis correspond.

- A limitation of the input value can be set in C01026/1 (upper limit) and C01026/2 (lower limit).
- The output value is internally limited to ±199.99 %. If a limitation of the output value is active, *bLimit* is set to TRUE.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input value (x) • ±16384 ≡ ±100 % • A limitation of the input value can be set in C01026/1 (upper limit) and C01026/2 (lower limit).

outputs

Designator Data type	Value/meaning	
nOut_a INT	Output value (y) • ±16384 ≡ ±100 % • Internal limitation to ±32767 (±199.99 %)	
bLimit BOOL	"Limitation active" status signal	
	TRUE	The output value is limited to ±199.99 %.

Parameters

Parameters	Possible settings			Information
C00959/2	-199.99	%	199.99	Output value • Read only
C001025/1				Function selection
	0	Out = 0		FB deactivated (Lenze setting)
	1	Out = In		
	2	Out = f(In)		In preparation!
	3	Out = f(table)		See description of **L_Curve_1**: ▸ Function 3: nOut_a = f(table)
C01026/1	-199.99	%	199.99	Upper limit for input value • Lenze setting: 199.99 %
C01026/2	-199.99	%	199.99	Lower limit for input value • Lenze setting: -199.99 %

Parameters	Possible settings			Information
C01028/1...32	-32767		32767	X values for characteristic function • Subcodes 1 ... 32 correspond to interpolation point values X1 ... X32. • ±16384 ≡ ±100 % • Lenze setting: 0 See description of **L_Curve_1**: Function 3: nOut_a = f(table)
C01029/1...32	-32767		32767	Y values for characteristic function • Subcodes 1 ... 32 correspond to interpolation point values Y1 ... Y32. • ±16384 ≡ ±100 % • Lenze setting: 0 See description of **L_Curve_1**: Function 3: nOut_a = f(table)

19.1.73 L_Curve_3

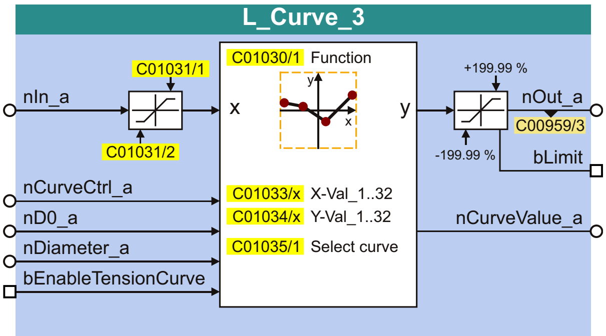
Alternatively, this FB represents a characteristic curve or cam function y=f(x), whereby the input signal of the X-axis and the output signal of the Y-axis correspond.

- A limitation of the input value can be set in C01031/1 (upper limit) and C01031/2 (lower limit).
- The output value is internally limited to ±199.99 %. If a limitation of the output value is active, *bLimit* is set to TRUE.

Tip!

From version 12.00.00 onwards the FB **L_Curve_3** can be used to create a tensiel force dependingt on the diameter for winding applications. For this purpose, the FB has been extended by further inputs/outputs and parameters.

▸ Use of the L_Curve_3 for tensile force characteristic



inputs

Designator Data type	Information/possible settings	
nIn_a INT	Input value (x) • ±16384 ≡ ±100 % • A limitation of the input value can be set in C01031/1 (upper limit) and C01031/2 (lower limit).	
	When using the FB for tensile force characteristic (from V12.00.00): Maximum tensile force setpoint in [%]	
The following inputs are only relevant when the FB is used for tensile force characteristic:		
nCurveCtrl_a INT From version 12.00.00	Slope of the characteristic (torque characteristic) in [%] • ±16384 ≡ ±100 % • Internal limitation to -100 ... +200 % • With 100 %, there is a constant tensile force curve and thus a torque proportionally rising with D.	
nD0_a INT From version 12.00.00	Initial point of the characteristic • 16384 ≡ 100 % ≡ maximum diameter	
nDiameter_a INT From version 12.00.00	Current diameter in [%] • 16384 ≡ 100 % ≡ maximum diameter	
bEnableTensionCurve BOOL From version 12.00.00	Activate tensile force setpoint	
	FALSE	Tensile force setpoint is not influenced.
	TRUE	Tensile force setpoint is multiplied by the characteristic value.

outputs

Designator Data type	Value/meaning	
nOut_a INT	Output value (y) • ±16384 ≡ ±100 % • Internal limitation to ±32767 (±199.99 %)	
	When using the FB for tensile force characteristic (from V12.00.00): Current tensile force setpoint in [%] • 16384 ≡ 100 %	
bLimit BOOL	"Limitation active" status signal	
	TRUE	The output value is limited to ±199.99 %.
nCurveValue_a INT From version 12.00.00	Current characteristic value of the tensile force characteristic in [%]	

Parameters

Parameters	Possible settings			Information
C00959/3	-199.99	%	199.99	Output value • Read only
C01030/1				Function selection
	0	Out = 0		FB deactivated (Lenze setting)
	1	Out = In		
	2	Out = f(In)		In preparation!
	3	Out = f(table)		See description of **L_Curve_1**: ▸ Function 3: nOut_a = f(table)
	4	Out = f(characteristic)		▸ Use of the L_Curve_3 for tensile force characteristic
C01031/1	-199.99	%	199.99	Upper limit for input value • Lenze setting: 199.99 %
C01031/2	-199.99	%	199.99	Lower limit for input value • Lenze setting: -199.99 %
C01033/1...32	-32767		32767	X values for characteristic function • Subcodes 1 ... 32 correspond to interpolation point values X1 ... X32. • ±16384 ≡ ±100 % • Lenze setting: 0 See description of **L_Curve_1**: ▸ Function 3: nOut_a = f(table)
C01034/1...32	-32767		32767	Y values for characteristic function • Subcodes 1 ... 32 correspond to interpolation point values Y1 ... Y32. • ±16384 ≡ ±100 % • Lenze setting: 0 See description of **L_Curve_1**: ▸ Function 3: nOut_a = f(table)

Parameters	Possible settings		Information
C01035/1 (from V12.00.00)			Selection of the tensile force profile ▸ Use of the L_Curve_3 for tensile force characteristic
	0	Linear tensile force profile	Lenze setting
	1	Linear torque profile	
	2	Tensile force profile according to characteristic	

19.1.73.1 Use of the L_Curve_3 for tensile force characteristic

In case of centre winders, the drive torque is transmitted from the centre via the individual layers up to the winder surface. In case of smooth materials and high diameter areas, it is mostly required to reduce the tensile force depending on the diameter.

The FB **L_Curve_3** can be used to generate a tensile force profile depending on the diameter. For this purpose, the selection "4: Out = f(characteristic)" has to be set in C01030/1.

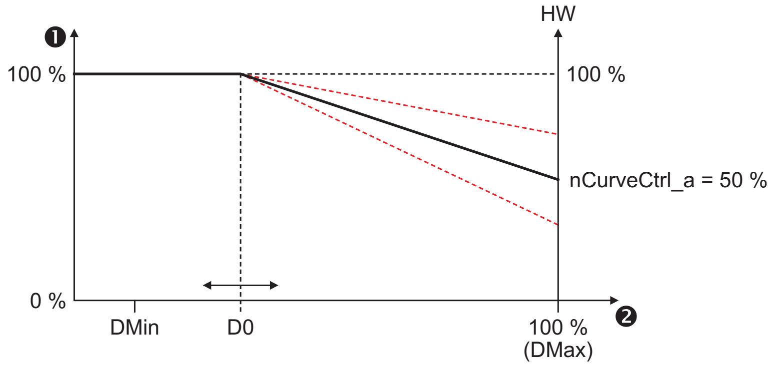
The characteristic is marked by an initial range with constant evaluation (100 %) and a second range where the tensile force is adapted to the diameter.

The part of the characteristic that depends on the diameter can be generated for a linear tensile force profile, a linear torque profile or based on a specified characteristic. The respective selection is made in C01035/1.

Linear tensile force profile

Characteristic for a linear tensile force profile (C01035/1 = 1)

Characteristic
❶ Tensile force
❷ Diameter



Function for D > D0:

$$100\,\% - \frac{D - D0}{100\,\% - D0} \cdot (100\,\% - HW)$$

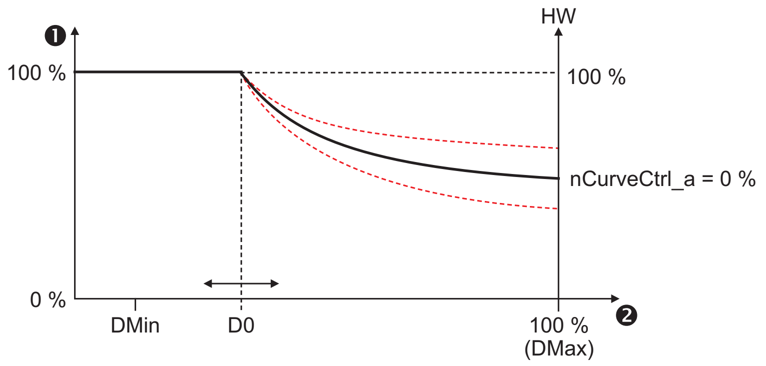
D	Current diameter (*nDiameter_a*)
D0	Initial point for diameter dependency (*nD0_a*)
HW	Tensile force valuation (*nCurveCtrl_a*)

- *nD0_a* serves to define at which diameter the tensile force decrease is to start.

Linear torque profile

Characteristic for a linear torque profile (C01035/1 = 2)

Characteristic
❶ Tensile force
❷ Diameter



Function for D > D0:

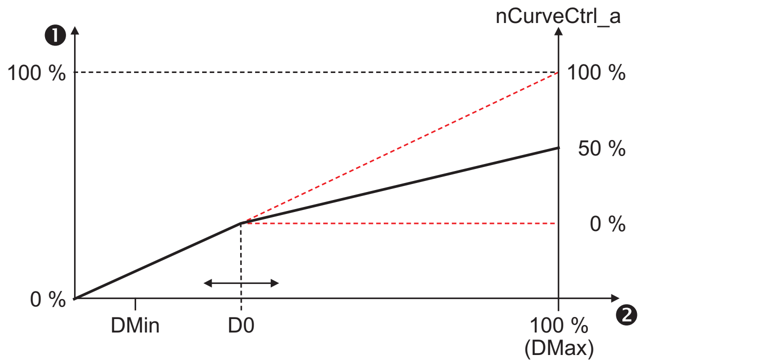
$$HW + \frac{(100\,\% - HW) \cdot D0}{D}$$

D	Current diameter (*nDiameter_a*)
D0	Initial point for diameter dependency (*nD0_a*)
HW	Tensile force valuation (*nCurveCtrl_a*)

- *nD0_a* serves to define at which diameter the tensile force decrease is to start.
- Since in this characteristic selection, the linear torque profile is in the focus, define in this case the slope of the torque curve via *dnCurveCtrl_n*:
  - When *nCurveCtrl_a* = 100 %, the tensile force remains unchanged.
  - When *nCurveCtrl_a* = 0 %, the tensile force is reduced up to D0/DMax.

Example: Steepness of the torque curve

Characteristic
❶ Torque
❷ Diameter



D	Current diameter (*nDiameter_a*)
D0	Initial point for diameter dependency (*nD0_a*)

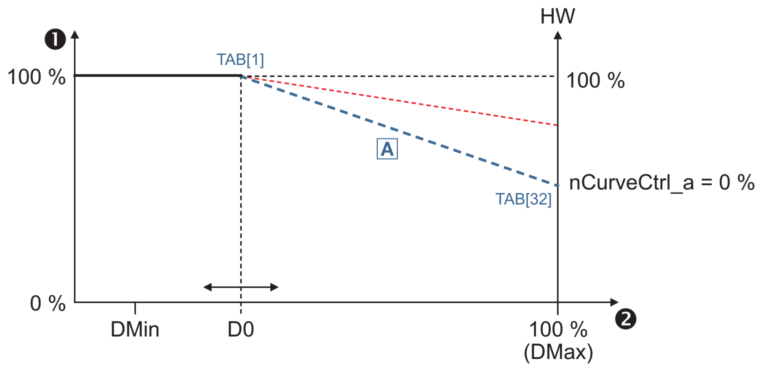
Tensile force profile according to characteristic

The characteristic (A) can be parameterised in C01033/1...32.

- The characteristic comprises 32 values, regularly distributed over the diameter range of D0 ... DMax.
- The values are to be set as raw values (±16384 ≡ ±100 %).

Tensile force profile according to defined characteristic (C01035/3 = 6)

Characteristic
❶ Tensile force
❷ Diameter



Function for D > D0:

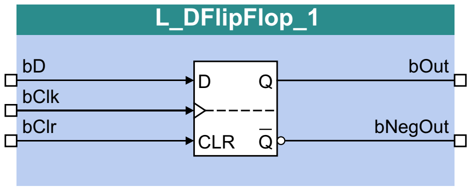
$$100\,\% - \frac{(100\,\% - TAB[D\text{-}D0]) \cdot (100\,\% - HW)}{100\,\%}$$

A	Table function (e.g. line)
TAB[D-D0]	Table value [1...32]for D-D0
D0	Initial point for diameter dependency (*nD0_a*)
HW	Tensile force valuation (*nCurveCtrl_a*)

- *nD0_a* serves to define the start of the table range (D0).
- *nCurveCtrl_a* serves to change the slope of the characteristic.
  - When *nCurveCtrl_a* = 0 %, the slope of the characteristic remains unchanged.

19.1.74 L_DFlipFlop_1

The FB saves binary signals (DFlipFlop) in a clock-controlled way.



inputs

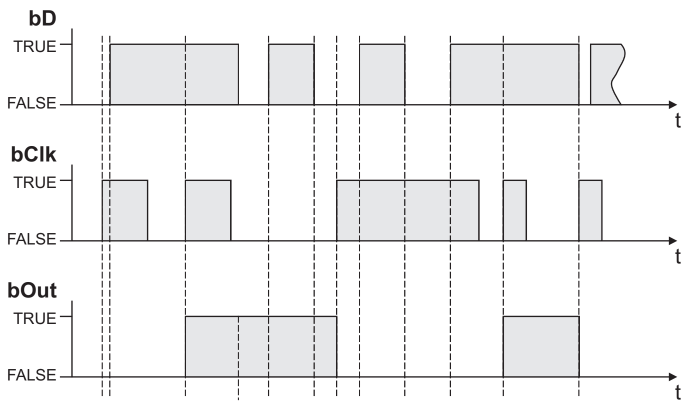
Designator Data type	Information/possible settings	
bD BOOL	Data input	
bClk BOOL	Clock input • Only FALSE/TRUE edges are evaluated	
bClr BOOL	Reset input	
	TRUE	• The *bOut* output is set to FALSE. • The *bNegOut* output is set to TRUE.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal
bNegOut BOOL	Output signal, inverted

Function

If the *bClr* input = FALSE, a signal edge at the *bClk* input switches the static input signal *bD* to the *bOut* output, where it is retained:

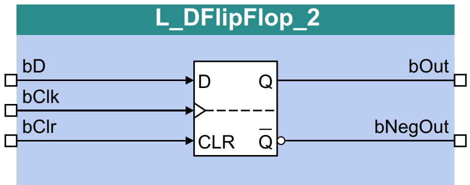


If the *bClr* input = TRUE:

- Due to the priority *bClr* > *bClk*, *bD* the *bOut* output signal can be set any time to the *FALSE* status by the *bClr* input signal = TRUE.
- The output signal is kept in this status independent of the other input signals.

19.1.75 L_DFlipFlop_2

The FB saves binary signals (DFlipFlop) in a clock-controlled way.



inputs

Designator Data type	Information/possible settings	
bD BOOL	Data input	
bClk BOOL	Clock input • Only FALSE/TRUE edges are evaluated	
bClr BOOL	Reset input	
	TRUE	• The *bOut* output is set to FALSE. • The *bNegOut* output is set to TRUE.

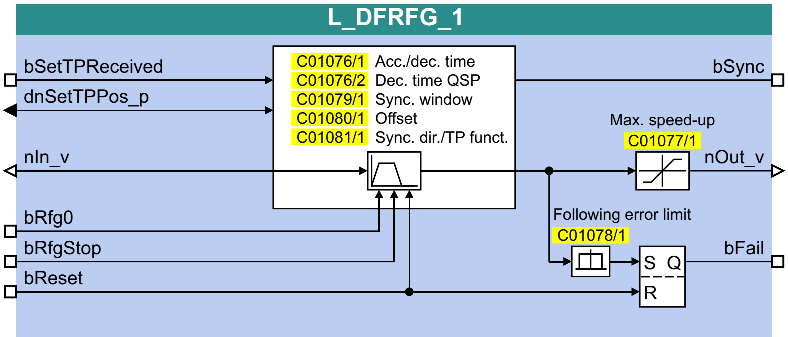
outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal
bNegOut BOOL	Output signal, inverted

For a detailed functional description see L_DFlipFlop_1.

19.1.76 L_DFRFG_1

This FB synchronises a drive (slave) to the master value of a master drive and then executes a angular synchronism with regard to the digital frequency.



inputs

Designator Data type	Information/possible settings	
bSetTPReceived BOOL	Input for status signal "Touch probe detected"	
	FALSE↗TRUE	Touch probe or zero pulse received.
dnSetTPPos_p DINT	Input for accepting the position measured via touch probe	
nIn_v INT	Speed setpoint of the master • Scaling: 16384 ≡ 15000 rpm	
bRfg0 BOOL	Ramp down ramp function generator to "0" (quick stop function) • This input can, for instance, be connected to the status signal "quick stop active".	
	TRUE	Ramp down the ramp function generator to "0" • The drive is brought to a standstill with the deceleration time set in C01076/2.
bRfgStop BOOL	Stop ramp function generator	
	TRUE	Stop ramp function generator • The last status is output to *nOut_v*. • The speed/angle setpoint *nIn_v* is saved. • After *bStop* is reset to FALSE, the setpoint angle is approached via the ramp function generator.
bReset BOOL	Reset added angle setpoint	
	TRUE	The internally added angle setpoint is reset and the ramp function generator is activated. If the status signal *bFail* is set, it will also be reset.
	TRUE↘FALSE	Detect speed/angle setpoint.

outputs

Designator Data type	Value/meaning	
bSync BOOL	Status signal "Drive is running synchronously"	
	TRUE	The drive is running synchronously, the speed input id directly applied to the speed output.
nOut_v INT	Speed/angle setpoint • Scaling: 16384 ≡ 15000 rpm	
bFail BOOL	Status signal "Angular difference exceeded"	
	TRUE	The angular difference set in C01078/1 has been exceeded.

Parameters

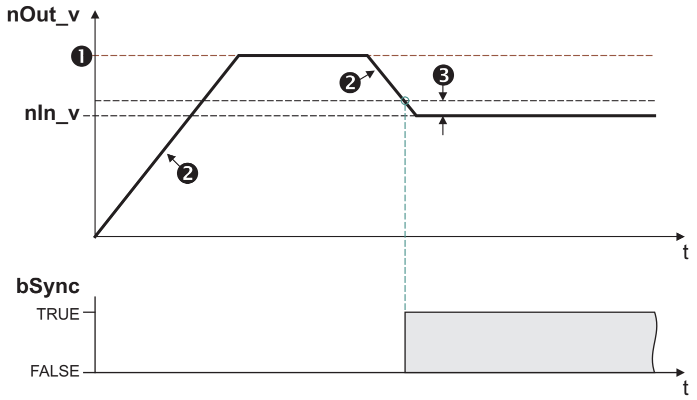
Parameters	Possible settings			Information
C01076/1	0.000	s	999.900	Acceleration and deceleration time • Relating to the reference speed (constant acceleration and deceleration) set in C00011. • It is a maximum value which limits the acceleration of the ramp function generator. • Lenze setting: 1.000 s
C01076/2	0.000	s	999.900	Deceleration time for quick stop of the slave drive • Relating to the reference speed (constant acceleration and deceleration) set in C00011. • It is a maximum value which limits the deceleration of the ramp function generator. • Lenze setting: 0.000 s
C01077/1	1	rpm	15000	Max. speed-up • Lenze setting: 3000 rpm
C01078/1	10	Incr.	2000000000	Following error limit • Scaling: A revolution is displayed with 65536 increments or steps. • Lenze setting: 2000000000 incr.
C01079/1	0	Incr.	65535	Synchronisation window (position) • If the difference between the master and the slave is lower than the synchronisation window, the *bSync* status signal is set to TRUE. • Lenze setting: 100 incr.
C01080/1	-2147483647	Incr.	2147483647	Offset • Angular offset for the internal actual position of the FB • Scaling: A revolution is displayed with 65536 increments or steps. • Lenze setting: 0 incr.

Parameters	Possible settings		Information
C01081/1			Sync. direction/TP function • Selection of the direction of rotation for synchronising • With high offset values and low input speed, the drive may change its direction of rotation. In order to prevent this, a direction of rotation can be permanently defined via this parameter.
	1	cw/ccw - without TP (Lenze setting)	
	2	cw - without TP	
	3	ccw - without TP	
	4	cw/ccw - with TP	
	5	cw - with TP	
	6	ccw - with TP	

19.1.76.1 Ramp function generator (profile generator)

The ramp function generator accelerates the slave drive to the input master speed.

- Synchronising starts when the FB receives a touch probe edge from the master axis (e.g. a conveying belt) at the *bSetTPReceived* input.
- When the slave reaches the setpoint speed of the master and the made up path difference is lower than the synchronisation window set in C01079/1, the *bSync* output is set to TRUE. At the same time, the FB switches the profile generator to inactive and outputs the *nIn_v* input speed to *nOut_v*.

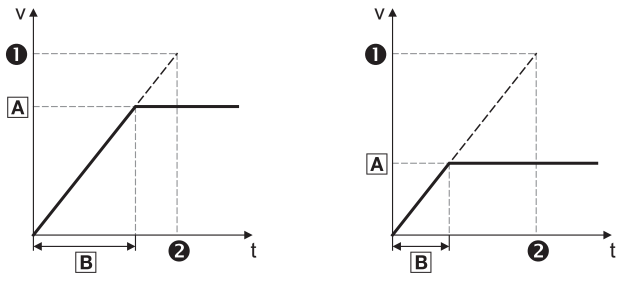


❶ C01077/1: Maximum speed
❷ C01076/1: Acceleration/deceleration time
❸ C01079/1: Synchronisation window

[19-29] Example: Synchronisation

- C01077/1 serves to set a maximum speed which limits the speed of the slave.

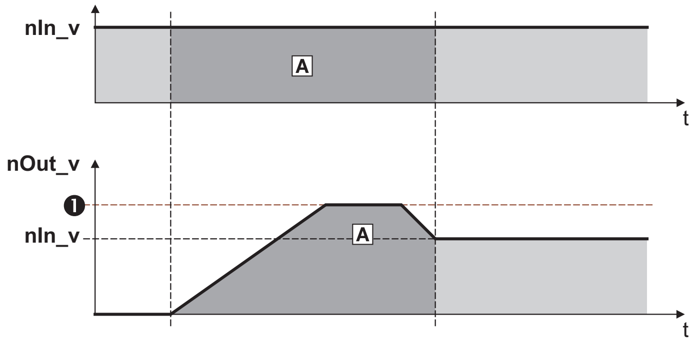
- The acceleration or deceleration in the synchronous point results from the acceleration/deceleration time set in C01076/1.
  - Reference for the acceleration/deceleration time is the reference speed (C00011):



❶ C00011: Reference speed ❷ C01076/1: Acceleration/deceleration time	A Effective target speed B Required time

[19-30] Connection between acceleration time and acceleration

- Based on the input speed of the master axis, a setpoint angle is calculated from the starting time onwards which leads the actual angle of the slave.
- Dependent on the master speed and the settings for acceleration and offset, the FB may travel oversynchronously for reducing the angular difference, i.e. *nOut_v* is higher than *nIn_v*:



❶ C01077/1: Maximum speed
A Identical surface below the speed profile that represents the covered path.

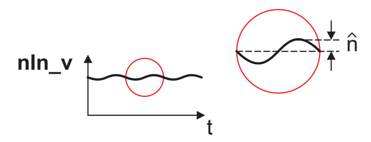
[19-31] Speed/time diagram

Note!

- Set the maximum speed in C01077/1 higher than the master speed to be expected. The speed is selected on the motor side and is independent of the reference speed (C00011). The higher the difference between maximum speed and master speed, the less time to the synchronous time is required.
- In case of a heavily oscillating input speed it may occur that directly after setting the *bSync* status signal to TRUE the FB still executes slight angle corrections.
- Generally avoid acceleration or deceleration processes of the master axis while the slave axes are synchronising.

Tip!

Set the synchronisation window in C01079/1 higher than the amplitude of beat available on the input signal:

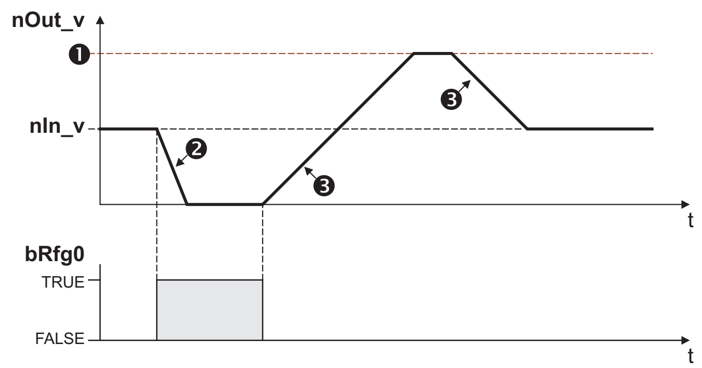


[19-32] Example: Beat on the input signal

19.1.76.2 Quick stop

The quick stop function removes the drive from the interconnection and leads it to standstill.

- Quick stop is activated by setting *bRfg0* to TRUE.
- C01076/2 is used to set the delay time in [s] which refers as the acceleration/deceleration time to the reference speed (C00011).
- The angle setpoint created internally on the basis of the input signal *nIn_v* is saved.
- After *bRfg0* is reset to FALSE, the angle setpoint is approached via the ramp function generator again.



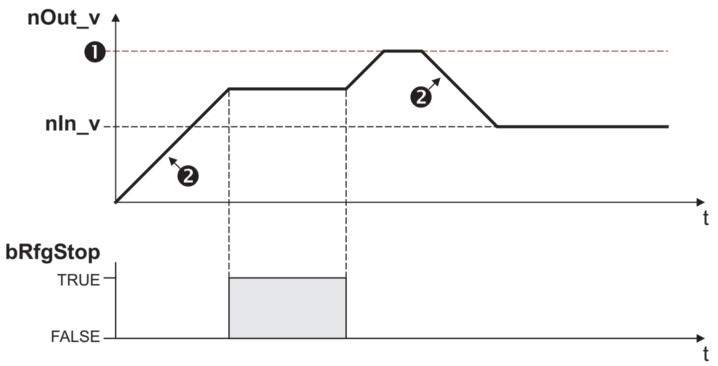
❶ C01077/1: Maximum speed
❷ C01076/2: Deceleration time for quick stop
❸ C01076/1: Acceleration/deceleration time

[19-33] Example: Quick stop function (QSP)

19.1.76.3 Ramp function generator stop

The stop function holds the status of the ramp function generator during operation.

- The stop function is activated by setting *bRfgStop* to TRUE.
- The last status is output to *nOut_v* (16384 ≡ 15000 rpm).
- The angle setpoint created internally on the basis of the input signal *nIn_v* is saved, thus the monitoring function of the angular difference between the created setpoint angle and actual angle cannot be activated.
- After *bRfgStop* is reset to FALSE, the angle setpoint is approached via the ramp function generator again.



❶ C01077/1: Maximum speed
❷ C01076/1: Acceleration/deceleration time

[19-34] Example: Stop function

19.1.76.4 Reset angle setpoint

By setting *bReset* to TRUE, the internally created angle setpoint is set to "0" and the ramp function generator is activated.

19.1.76.5 Detecting the angular difference

In C01078/1 , a limit value for monitoring the angular difference between the created setpoint and actual angles can be set.

- Scaling: A revolution is displayed with 65536 increments or steps.
- The ramp function generator can accept a angular difference of up to ±2140000000 increments (≈32000 revolutions).
- If monitoring responds, the status signal *bFail* is set to TRUE.
- If the internally added angle setpoint is reset by setting *bReset* to TRUE, the *bFail* status signal is reset to FALSE.

19.1.76.6 Offset setting

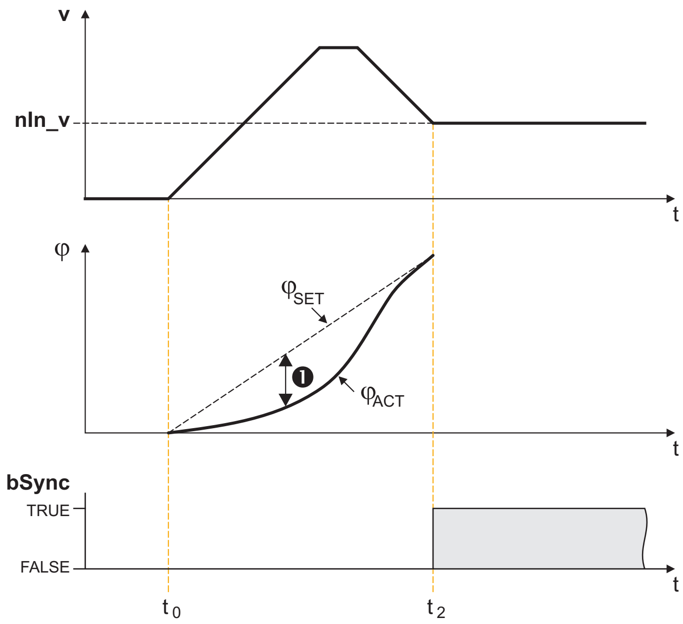
If the time to synchronicity is too long or the slave drive must not be operated oversynchronously, either the acceleration can be adapted or the touch probe initiator can be shifted. Since this is only possible to a limited extent due to construction conditions, in this case the better solution is selecting a virtual angular offset.

- In order to reach angle synchronicity, the difference between created setpoint and actual angle must be zero:

$$\varphi_{DIFF} = \varphi_{SET} - \varphi_{ACT} = 0$$

Behaviour without offset

If the actual angle starts with an offset = 0, a higher difference between setpoint angle and actual angle develops:

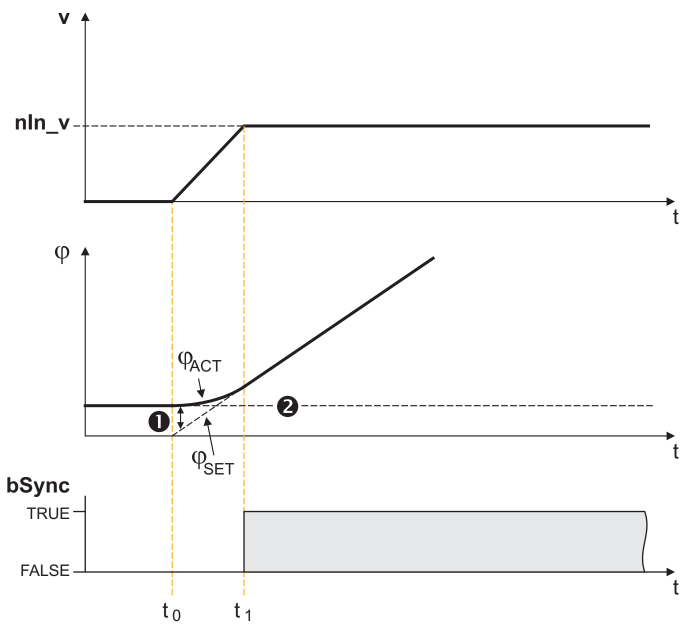


❶ Difference between setpoint angle and actual angle (φ_{DIFF})
t_0 = touch probe event
t_2 = synchronicity reached

[19-35] Example: Without offset selection (offset = 0)

Behaviour with offset

By selecting an offset, the actual position is preloaded with a non-zero value. This reduces the distance and the time to synchronicity.



❶ Difference between setpoint angle and actual angle (φ_{DIFF})
❷ Offset [inc]
t_0 = touch probe event
t_1 = synchronicity achieved

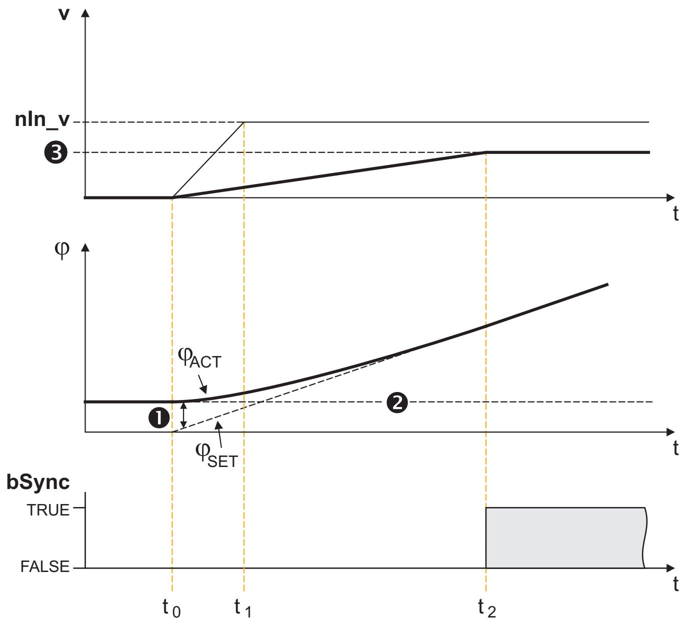
[19-36] Example 1: With offset selection

- When a positive offset is selected, the sign of the distance between setpoint and actual value changes. This distance is reduced due to the acceleration of the slave from the input of the touch probe pulse. Thus, the time to synchronicity is shorter.
- The offset refers to the master value selection and is scaled with an encoder revolution (≡ 65536 increments).
- The offset can be determined empirically, but it is sensible to calculate the required acceleration distance and select this value as offset:

$$\varphi_{ACC} = \frac{1}{2} \cdot \frac{(v_{SET})^2}{C00011\ [rpm]} \cdot C01076/1\ [ms] \cdot \frac{16384\ [incr./ms]}{15000\ [rpm]}$$

φ_{ACC} = acceleration distance in [increments]
v_{SET} = setpoint speed in [rpm]
C00011 = reference speed of the motor in [rpm]
C01076/1 = acceleration/deceleration time of the ramp generator in [ms]

The acceleration calculated from the acceleration/deceleration time (C01076/1) is a maximum acceleration that will not be reached if the input speed is reduced and the FB has to reach its target with a too high offset.



❶ Difference between setpoint angle and actual angle (φ_{DIFF})
❷ Offset [inc]
❸ Reduced input speed
t_0 = touch probe event
t_1 = reaching synchronicity at original input speed
t_2 = synchronicity reached

[19-37] Example 2: With offset selection and reduced input speed

Note!

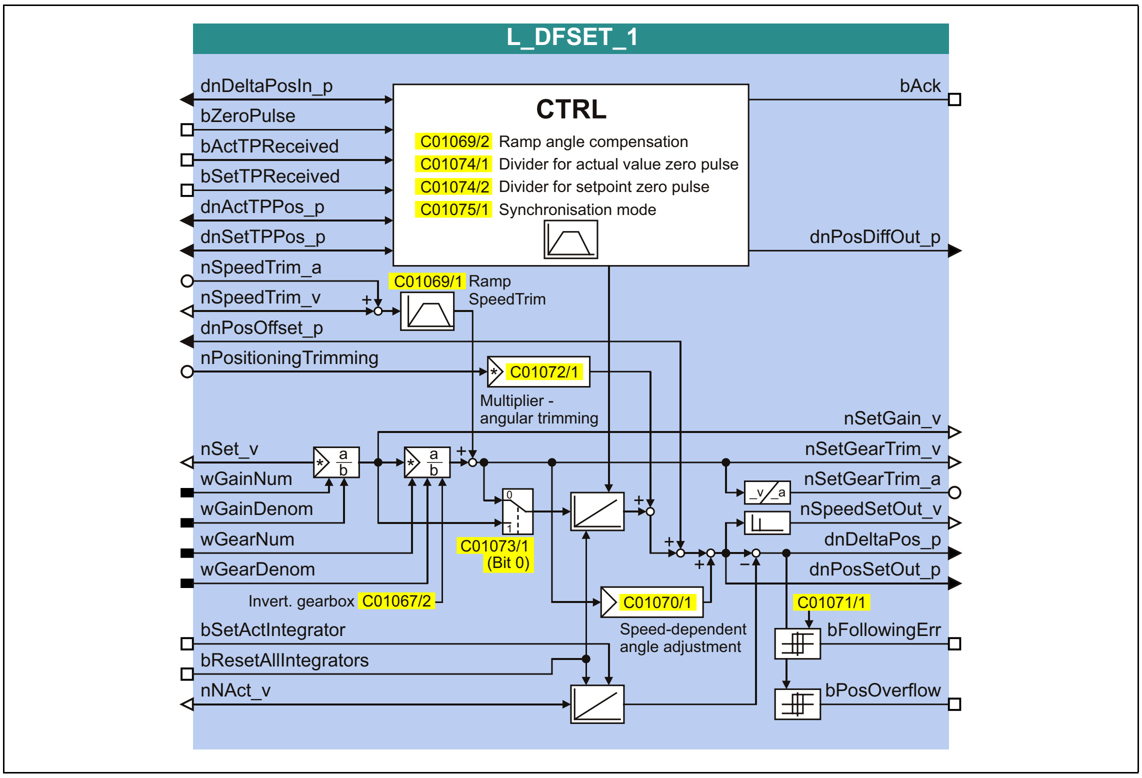
With a very low input speed, the acceleration can also get negative which causes the slave drive to change its direction of rotation.

In order to prevent a change of the direction of rotation, C01069/1 serves to select a permanent direction of rotation.

19.1.77 L_DFSET_1

This FB prepares the master value for a slave drive. This FB enables the inverter to follow the master drive true to speed and angle.

- You can select values for the stretch factor and gearbox factor of the slave.
- A speed trimming or angular trimming is possible.



inputs

Designator / Data type		Information/possible settings
dnDeltaPosIn_p	DINT	Input for considering a following error for the mark correction • The value of this input is used for the following error compensation if the *nNAct_v* input is not assigned.
bZeroPulse	BOOL	Enable of the zero pulse-/touch probe synchronisation for the modes 2, 10 ... 13 • The mode is selected in C01075/1. FALSE↗TRUE TRUE: Synchronisation enabled.
bActTPReceived	BOOL	Input for detecting an actual value touch probe or zero pulse FALSE↗TRUE: Actual value touch probe or zero pulse received.
bSetTPReceived	BOOL	Input for status signal "Touch probe detected" FALSE↗TRUE: Touch probe or zero pulse received.
dnActTPPos_p	DINT	Input for accepting the position measured via touch probe at *bActTPReceived*
dnSetTPPos_p	DINT	Input for accepting the position measured via touch probe at *bSetTPReceived*
nSpeedTrim_a	INT	Speed trimming in [%] • Scaling: 16384 ≡ 100 %

Designator Data type	Information/possible settings
nSpeedTrim_v INT	Speed trimming in [increments/ms] • Scaling: 16384 ≡ 15000 rpm • Speed trimming via this input is more precise.
dnPosOffset_p DINT	Angular offset in [increments] • Scaling: A motor revolution is mapped with 65536 increments or steps.
nPositionTrimming INT	Angular trimming in [increments] • Scaling: A motor revolution is mapped with 65536 increments or steps. • When analog values are selected: 100 % ≡ 1/4 revolution ≡ 16384 increments • The setting range can be extended via C01072/1.
nSet_v INT	Speed setpoint • Scaling: 16384 ≡ 15000 rpm
wGainNum WORD	Stretch factor (numerator)
wGainDenom WORD	Stretch factor (denominator)
wGearNum WORD	Gearbox factor (numerator)
wGearDenom WORD	Gearbox factor (denominator)
bSetActIntegrator BOOL	Equalise angle integrators (current position = set position) • This input has a higher priority than the *bResetAllIntegrators* input. FALSE↗TRUE: Set following error *dnDeltaPos_p* = 0. TRUE: Equalise angle integrators.
bResetAllIntegrators BOOL	Reset angle integrators TRUE: Positional deviation, *dnPosSetOut_p* and *dnDeltaPos_p* are set to "0".
nNAct_v INT	Actual value in [increments/ms] for calculating the actual position • Scaling: 16384 ≡ 15000 rpm • If this input is connected or assigned to an output, the following error created from integration of this input and difference with the set position is used for the following error compensation in case of mark correction.

outputs

Designator Data type	Value/meaning
bAck BOOL	Status signal of the internal angle correction (or control signal for external angle correction) TRUE: Status signal "Synchronising is executed" • When the external angle correction is deactivated (C01073/1 - bit 2 = "0") and a ramp for the angle compensation is set in C01069/2. FALSE↗TRUE: Control signal "Start external angle correction" • When the external angle correction is activated (C01073/1 - bit 2 = "1") and the angular offset is measured.
dnPosDiffOut_p DINT	Angular offset in [increments] • Difference between two positions measured by means of touch probe.
nSetGain_v INT	Speed setpoint in [increments/ms] evaluated with stretch factor • Scaling: 16384 ≡ 15000 rpm
nSetGearTrim_v INT	Speed setpoint in [increments/ms] evaluated with stretch factor and gearbox factor • Scaling: 16384 ≡ 15000 rpm
nSetGearTrim_a INT	Speed setpoint in [%] evaluated with stretch factor and gearbox factor • Scaling: 16384 ≡ 100 % reference speed (C00011)

Designator Data type	Value/meaning	
nSpeedSetOut_v INT	Speed setpoint in [increments/ms] • Scaling: 16384 ≡ 15000 rpm • Derived from angle setpoint *dnPosSetOut_p*. • Can change abruptly if e.g. the angular offset changes abruptly.	
dnDeltaPos_p DINT	Following error between setpoint position and actual position in [increments] • This output can be used for display purposes.	
dnPosSetOut_p DINT	Angle setpoint in [increments] • Scaling: A revolution is displayed with 65536 increments or steps.	
bFollowingErr BOOL	Status signal "Following error"	
	TRUE	Following error occurred. • The limit value set in C01071/1 has been exceeded.
bPosOverflow BOOL	Status signal "angle controller overflow" • An angle controller overflow occurs from a limit value of 2130706432 increments.	
	TRUE	Angle controller overflow occurred.

Parameters

Parameters	Possible settings			Information
C01067/2				Invert. gearbox nSet_v
	0	Not inverted		
	1	Inverted		
	2	Automatically from MCK		
C01069/1	0	Incr./ms	32767	Ramp SpeedTrim • Lenze setting: 10 incr./ms • From version 12.00.00 onwards, this setting has a resolution increased by the factor 100: A value of 32767 is internally evaluated with 327.67. Hence, the unit is [incr./100 ms].
C01069/2	0	Incr./ms	32767	Ramp angle compensation • The increments for the angle compensation in case of mark correction are specified in the Lenze setting with downstream 3rd-order polynomial. • Lenze setting: 100 incr./ms
C01070/1	-134217728	Incr.	134217728	Speed-dependent angle adjustment • Scaling: A revolution is displayed with 65536 increments or steps. • Lenze setting: 0 incr.
C01071/1	10	Incr.	2147483647	Following error limit • Scaling: A motor revolution is mapped with 65536 increments or steps. • Lenze setting: 32768 incr.
C01072/1	-20000		20000	Multiplier - angular trimming • Lenze setting: 1

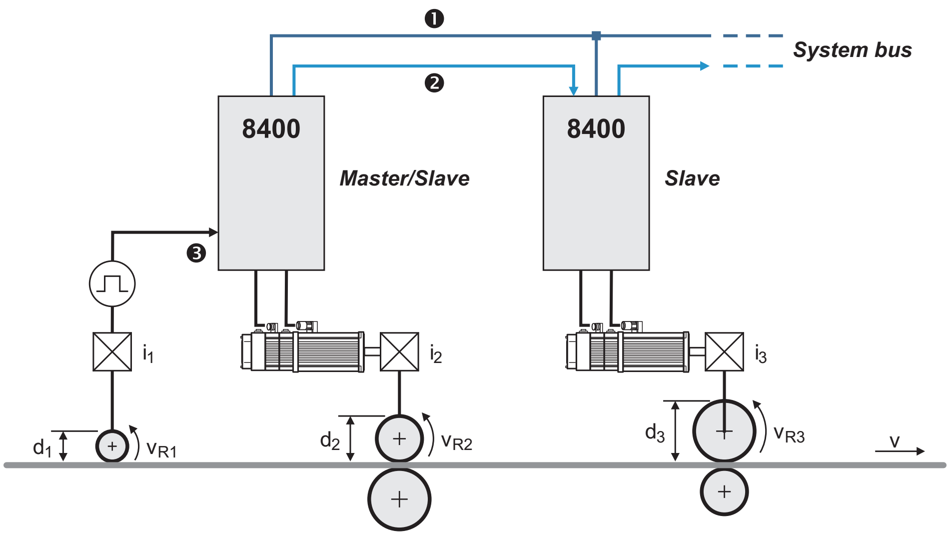
Parameters	Possible settings			Information
C01073/1	Setting is bit coded:			setting • Lenze setting: 0x02
	Bit 0	Setpoint angle without gearbox factor		0 = Evaluation of the setpoint angle with gearbox factor. 1 = Evaluation of the setpoint angle without gearbox factor.
	Bit 1	Angle correction with polynomial		0 = Angle correction without downstream polynomial. 1 = Angle correction with downstream polynomial.
	Bit 2	External angle correction		0 = Internal angle correction 1 = External angle correction • From version 13.00.00
	Bit 3 ... Bit 7	Reserved		
C01074/1	0		16384	Divisor for actual value zero pulse • Number of signals to be received at least until synchronisation gets active. • Initialisation: 0 (≡ 1)
	Note: The settings 0 and 1 are functionally identical. If 0 or 1 is set, at least one pulse is required in order that the synchronisation gets active.			
C01074/2	0		16384	Divisor for setpoint zero pulse • Number of signals to be received at least until synchronisation gets active. • Initialisation: 0 (≡ 1)
	Note: The settings 0 and 1 are functionally identical. If 0 or 1 is set, at least one pulse is required in order that the synchronisation gets active.			

Parameters	Possible settings		Information
C01075/1			Synchronisation mode - mark correction
	0	Synchronisation inactive	Lenze setting
	1	Permanent synchronisation without enable via the *bZeroPulse* input • Correction is executed over the shortest possible path.	The synchronisation only takes place after 2 setpoint pulses have been received at the *bSetTPReceived* input. (required pulse order: set-actual-set)
	2	Permanent synchronisation only with enable via the *bZeroPulse* input • When *bZeroPulse* is set to TRUE, a permanent zero pulse/touch probe synchronisation is executed. • Correction is executed over the shortest possible path.	
	10	One-time synchronisation • An angular difference is compensated over the shortest possible path.	
	11	One-time synchronisation • An angle difference is compensated in CW direction.	
	12	One-time synchronisation • An angular difference is compensated in CCW direction.	
	13	One-time synchronisation • An angular difference is compensated over the shortest possible path.	Synchronisation takes place after the first setpoint pulse has been received at the *bSetTPReceived* input and/or actual value pulse at the *bActTPReceived* input. (Required clock sequence: Set/actual or actual/set)

19.1.77.1 Master value rail/slave cascade

If there is only one master speed that is transmitted to all slave drives involved, e.g. via system bus (CAN), it is called master value rail.

If a slave drive takes over the master value generation for the following slave, it is called slave cascade.



❶ Master value rail
❷ Slave cascade
❸ Master value encoder

[19-38] Example: Master value rail/slave cascade

19.1.77.2 Setpoint conditioning with stretch factor and gearbox factor

Stretch factor

The stretch factor is required for the "speed synchronism via master value cascade" mode. It defines the ratio the slave drive is to be running with regard to its master value.

- The stretch factor evaluates the setpoint at the *nSet_v* input.
- The stretch factor must be selected via the *wGainNum* and *wGainDenom* inputs in the form of numerators and denominators.
- The result is provided at the *nSetGain_v* output.
  - Scaling: 16384 ≡ 15000 rpm

$$\mathrm{nSetGain\_v} = \mathrm{nSet\_v} \cdot \frac{\mathrm{wGainNum}}{\mathrm{wGainDenom}}$$

- If the stretch factor is 1 and the gearbox factors are selected correctly, the circumferential speeds of the rolls for master and slave 1 are identical in the example shown in the illustration [19-38].

Gearbox factor

The gearbox factor defines the gearbox ratio of the drive. Enter the ratio of the drive.

- The gearbox factor evaluates the setpoint at the *nSet_v* input multiplied by the stretch factor.
- The gearbox factor has to be selected via the inputs *wGearNum* and *wGearDenom* in the form of numerators and denominators.
- The bit 0 in C01073/1 serves to simply deactivate the evaluation with the gearbox factor (does not apply to the speed outputs *nSetGain_v*, *nSetGearTrim_v* and *nSetGearTrim_a*).
- The result is provided at the outputs *nSetGearTrim_v* and *nSetGearTrim_a*.
  - Scaling *nSetGearTrim_v*: 16384 ≡ 15000 rpm
  - Scaling *nSetGearTrim_a*: 16384 ≡ 100 % reference speed (C00011)

$$\text{nSetGearTrim\_v} = \text{nSet\_v} \cdot \text{Stretch factor} \cdot \frac{\text{wGearNum}}{\text{wGearDenom}}$$

$$\text{nSetGearTrim\_v} = \text{nSet\_v} \cdot \frac{\text{wGainNum}}{\text{wGainDenom}} \cdot \frac{\text{wGearNum}}{\text{wGearDenom}}$$

$$\text{nSetGearTrim\_a} = \text{nSet\_v} \cdot \frac{\text{C00011}}{15000} \cdot \frac{\text{wGainNum}}{\text{wGainDenom}} \cdot \frac{\text{wGearNum}}{\text{wGearDenom}}$$

Note!

The intermediate results and the result of the evaluation are limited to ±32767 increments (16 bits).

19.1.77.3 Processing correction values

Speed trimming

Speed trimming enables the connection of correction values, e.g. from a higher-level control loop. This permits an acceleration or deceleration of the drive.

The correction value can either be transmitted as an analog value via the *nSpeedTrim_a* input or for exact speed trimming as a speed value via the *nSpeedTrim_v* input.

- Scaling *nSpeedTrim_a*: 16384 ≡ 100 % reference speed (C00011)
- Scaling *nSpeedTrim_v*: 16384 ≡ 15000 rpm

Speed variations of the trimming values are specified via the SpeedTrim ramp (C01069/1).

Angular trimming

Angular trimming enables the rotor position to be put forward or back with regard to the setpoint (the drive is leading or lagging).

The correction value must has to be defined in [increments] via the *nPositionTrimming* input. It is internally added to the angle setpoint.

- Scaling: A motor revolution is mapped with 65536 increments or steps.
- The maximum angular trimming amounts to ±½ motor revolution (±32767 increments).
- When analog values are selected: 100 % ≡ ¼ motor revolution ≡ 16384 increments.
- The setting range can be extended with the multiplier to be set in C01072/1.

Angular offset

The *dnPosOffset_p* input serves to define a permanent angular offset for the setpoint of the drive.

- An adjustment takes place via the following error (*dnDeltaPos_p*) output and via the speed change at the *nSpeedSetOut_v* output.
- Example: A angular offset of 90° is to be set on the load side with a gearbox ratio of 3.8147:

$$\text{dnPosOffset\_p} = \frac{90^\circ}{360^\circ} \cdot 65536 \left[\frac{\text{Increments}}{\text{Motor revolution}}\right] \cdot 3.8147 = 62500\ [\text{increments}]$$

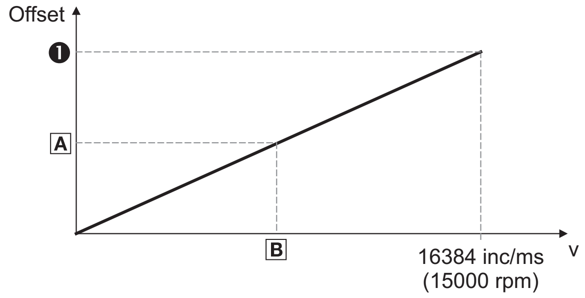
Stop!

Change the angular offset only step by step as the FB does not contain a ram generator for preventing torque impulses.

Speed-proportional angle adjustment

The speed-proportional angle adjustment enables a leading or lagging of the angle with increasing speed and serves to compensate dead times and transmission times.

The correction value selected in C01070/1. in [increments] refers to a speed of 15000 rpm (linear relationship):



❶ C01070/1: Speed-proportional angle adjustment
A Effective offset
B Current speed

[19-39] Connection between offset/speed

19.1.77.4 Synchronising slave drive to master drive

If an angular synchronism of the drives is required (e.g. in printing units), the slave drives must be positioned to the master position since the FB **L_DFSET** as a master value can only receive and process one relative signal. The FB **L_DFSET** receives the master value via a real-time capable MotionBus as the system bus (CAN).

Tip!

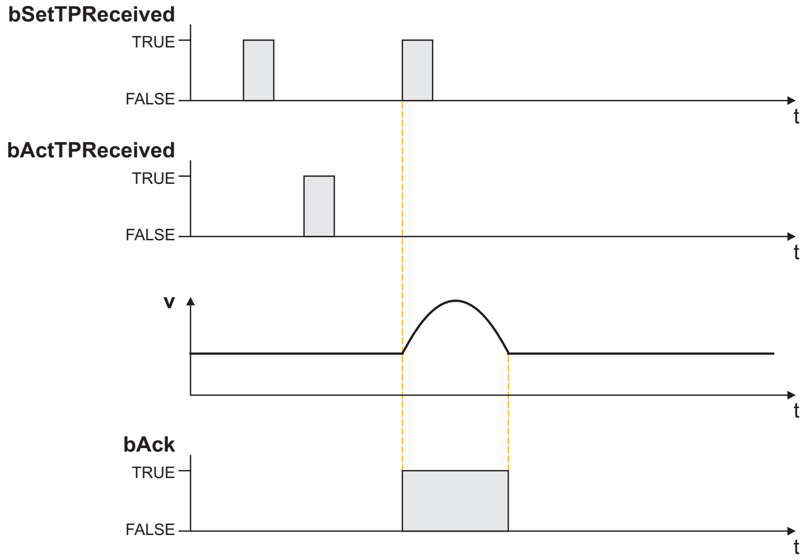
The 8400 TopLine serves to realise very easily a master value cascade via the cross communication with the Axis bus of the drives.

For synchronisation purposes, the slave drive is informed cyclically or once about its drive position by a pulse generated by the master drive. This pulse is either the zero pulse of the master feedback or the edge of a touch probe sensor. Moreover, a pulse must be generated by the slave drive. Only if the FB **L_DFSET** receives both signals, it can execute a synchronisation.

Synchronisation mode - mark correction

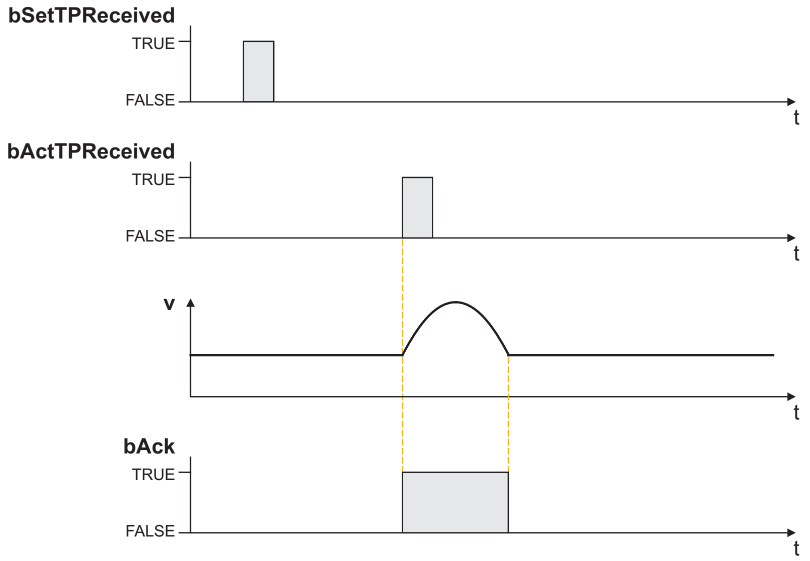
For synchronisation, the following modes are available in C01075/1 :

Mode		Info
0	Synchronisation inactive	Lenze setting
1	Permanent synchronisation without enable via the *bZeroPulse* input • Correction is executed over the shortest possible path.	The synchronisation only takes place after 2 setpoint pulses have been received at the *bSetTPReceived* input. (required pulse order: set-actual-set) • See illustration [19-40]. (📖 1596)
2	Permanent synchronisation only with enable via the *bZeroPulse* input • When *bZeroPulse* is set to TRUE, a permanent zero pulse/touch probe synchronisation is executed. • Correction is executed over the shortest possible path.	
10	One-time synchronisation only with enable via the *bZeroPulse* input • An angular difference is compensated over the shortest possible path.	
11	One-time synchronisation only with enable via the *bZeroPulse* input • An angle difference is compensated in CW direction.	
12	One-time synchronisation only with enable via the *bZeroPulse* input • An angular difference is compensated in CCW direction.	
13	One-time synchronisation only with enable via the *bZeroPulse* input • An angular difference is compensated over the shortest possible path.	Synchronisation takes place after the first setpoint pulse has been received at the *bSetTPReceived* input and/or actual value pulse at the *bActTPReceived* input. (Required clock sequence: Set/actual or actual/set) • See illustration [19-41]. (📖 1596)



Synchronisation mode (C01075/1) = 1, 2, 10, 11, 12
Divisor for actual value zero pulse (C01074/1) = 0 or 1
Divisor for setpoint zero pulse (C01074/2) = 0 or 1

[19-40] Synchronisation process in the modes 1, 2, 10, 11, 12



Synchronisation mode (C01075/1) = 13
Divisor for actual value zero pulse (C01074/1) = 0 or 1
Divisor for setpoint zero pulse (C01074/2) = 0 or 1

[19-41] Synchronisation process in mode 13

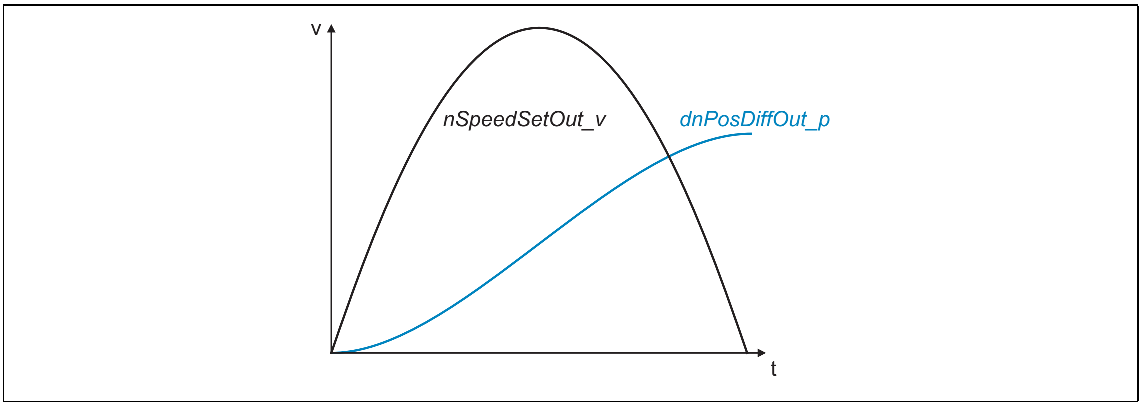
Compensation process

At the input of the second set or actual touch probe pulse, the difference between master and slave position is detected which is then provided via the *nSpeedSetOut_v* speed output and equivalently as position at .

In C01073/1, the following bit coded settings can be made for the compensation procedure:

setting		Info
Bit 1	Angle correction with polynomial	If this bit is set (Lenze setting),the compensating movement is rounded by polynomial. • The angle error to be compensated is limited to ±1073741823 increments due to the system. This applies in general when the **L_DFSET_1** carries out the compensation. • The maximum compensating speed can be set in C1069/2. When "0" is set in C1069/2 if activated in C01075/1, the angle error is measured and output to *dnPosDiffOut_p*.
Bit 2	External angle correction From version 13.00.00	If this bit is set, the FB **L_DFSET_1** does not compensate any angle errors. • Every time the angle errors have been measured, the angular difference is output to *dnPosDiffOut_p*. • In addition, a FALSE/TRUE edge is output to *bAck*. This signal serves to trigger an external profile generator which compensates the angular offset via a speed profile additionally.

The following illustration shows a compensating process after the marks have been detected. The speed is added here as a parabola to the line speed. (The scaling in this diagram does not correspond to reality)

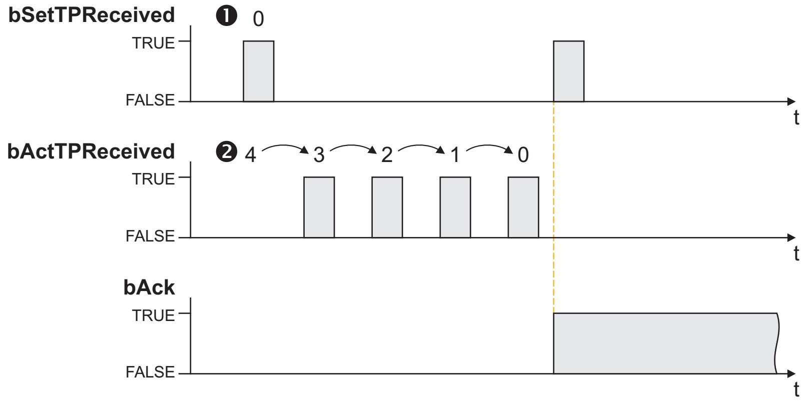


[19-42] Example: Compensation process

19.1.77.5 Masking out touch probe signals

When passing-through material is used, e.g. printed foil, touch probe initiators may respond several times per cycle. For suppressing such "interference pulses", count values can be selected in C01074/1 and C01074/2 that are decremented when the touch probe pulse has been received. Only when the counter content is "0", the synchronisation will be enabled. Please note that the settings 0 and 1 are functionally identical. If 0 or 1 is set, at least one pulse is required to activate the synchronisation.

Synchronisation mode = 1, 2, 10, 11, 12



❶ Divisor for setpoint zero pulse (C01074/2) = 0 or 1
❷ Divisor for actual value zero pulse (C01074/1) = 4

[19-43] Synchronisation process in the modes 1, 2, 10, 11, 12 with a masking out of the touch probe pulses

19.1.77.6 Process monitoring functions

Following error

The *bFollowingErr* status output is set to TRUE if the drive cannot follow its setpoint angle.

- Possible causes:
  - The centrifugal mass is too high for the set acceleration or deceleration time.
  - The torque limit has been reached (load torque > drive torque).
- Remedy: Unload drive or increase torque limit at the servo controller (if the power limits of the inverter have not yet been reached).

The following error is derived from the angular difference of the setpoint angle integrator minus the actual angle integrator. The comparison value (following error limit) can be set in C01071/1.

Angle controller overflow (bPosOverflow = TRUE)

The *bPosOverflow* status output is set to TRUE if the angular difference that can be displayed device-internally has been exceeded. Home positions get lost here.

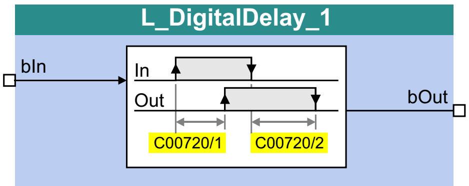
Tip!

If an error response is to be triggered in case of a following error and/or angle controller overflow, connect the corresponding status output with a free *bSetError* input of the SB LS_SetError_2 and parameterise the requested error response for this input in C00581.

19.1.78 L_DigitalDelay_1

This FB applies a time delay to binary signals.

- ON and OFF-deceleration can be parameterised separately.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input signal

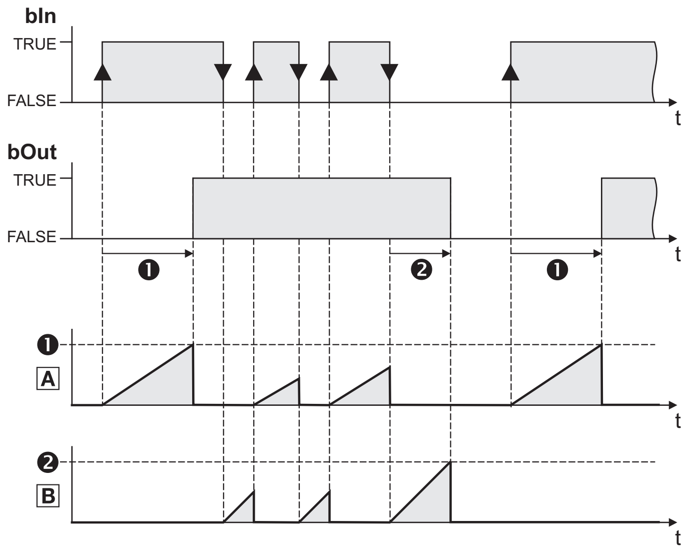
outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal (time-delayed input signal)

Parameters

Parameters	Possible settings			Information
C00720/1	0.000	s	3600.000	ON-deceleration • Lenze setting: 0.000 s
C00720/2	0.000	s	3600.000	OFF-deceleration • Lenze setting: 0.000 s

Function



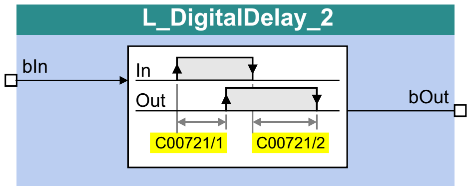
❶ ON delay
❷ OFF delay
[A] Internal timing element for ON delay (retriggerable)
[B] Internal timing element for OFF delay (retriggerable)

1. A FALSE-TRUE edge at b*In* starts the internal timing element for the ON delay.
2. After the defined ON delay, the input signal *bIn* is output at *bOut*.
3. A TRUE-FALSE edge at b*In* starts the internal timing element for the OFF delay.
4. After the defined OFF delay, the input signal *bIn* is output at *bOut*.

19.1.79 L_DigitalDelay_2

This FB applies a time delay to binary signals.

- ON and OFF-deceleration can be parameterised separately.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal (time-delayed input signal)

Parameters

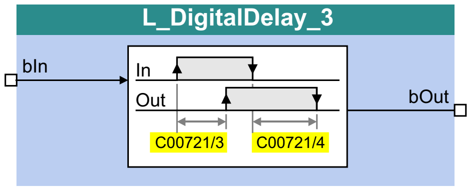
Parameters	Possible settings			Information
C00721/1	0.000	s	3600.000	ON-deceleration • Lenze setting: 0.000 s
C00721/2	0.000	s	3600.000	OFF-deceleration • Lenze setting: 0.000 s

For a detailed description see L_DigitalDelay_1.

19.1.80 L_DigitalDelay_3

This FB applies a time delay to binary signals.

- ON and OFF-deceleration can be parameterised separately.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal (time-delayed input signal)

Parameters

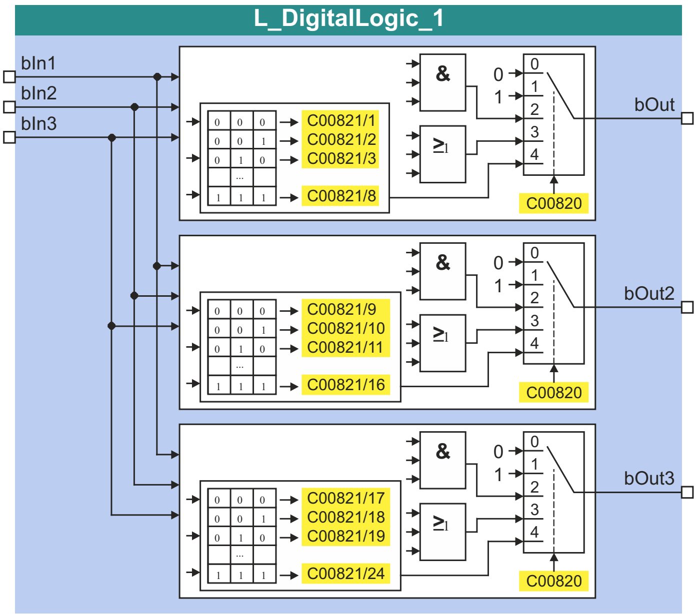
Parameters	Possible settings			Information
C00721/3	0.000	s	3600.000	ON-deceleration • Lenze setting: 0.000 s
C00721/4	0.000	s	3600.000	OFF-deceleration • Lenze setting: 0.000 s

For a detailed description see L_DigitalDelay_1.

19.1.81 L_DigitalLogic_1

This FB provides a binary output signal created by a logic operation of the input signals. Optionally, one of the constant binary values independent from the input signals can be output.

- Output of a constant binary value
- AND operation of the inputs
- OR operation of the inputs
- Output depending on the combination of the input signals



inputs

Designator Data type	Information/possible settings
bIn1 BOOL	Input signal 1
bIn2 BOOL	Input signal 2
bIn3 BOOL	Input signal 3

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal 1
bOut2 BOOL	Output signal 2 • bOut2 is available from version 16.00.00 onwards
bOut3 BOOL	Output signal 3 • bOut3 is available from version 16.00.00 onwards

Parameters

Parameters	Possible settings		Information
C00820			Function selection
	0	"0"	Constant value "FALSE"
	1	"1"	Constant value "TRUE"
	2	$bOut = bIn1 \wedge bIn2 \wedge bIn3$ $bOut2 = bIn1 \wedge bIn2 \wedge bIn3$ $bOut3 = bIn1 \wedge bIn2 \wedge bIn3$	AND operation • bOut2 and bOut3 are available from version 16.00.00 onwards
	3	$bOut = bIn1 \vee bIn2 \vee bIn3$ $bOut2 = bIn1 \vee bIn2 \vee bIn3$ $bOut3 = bIn1 \vee bIn2 \vee bIn3$	OR operation • bOut2 and bOut3 are available from version 16.00.00 onwards
	4	bOut = f (truth table) bOut2 = f (truth table) bOut3 = f (truth table)	The output value depends on the truth table parameterised in C00821/1...24 • bOut2 and bOut3 are available from version 16.00.00 onwards
C00821/1...24			Truth table for function "4: bOut = f (truth table)" • Each of the eight possible input combinations can be assigned to the output value FALSE or TRUE. • For an application example see the following section.
	0	FALSE	
	1	TRUE	

Function 4

If the function 4 is selected in C00820, the output values depend on the truth table parameterised in C00821/1...24.

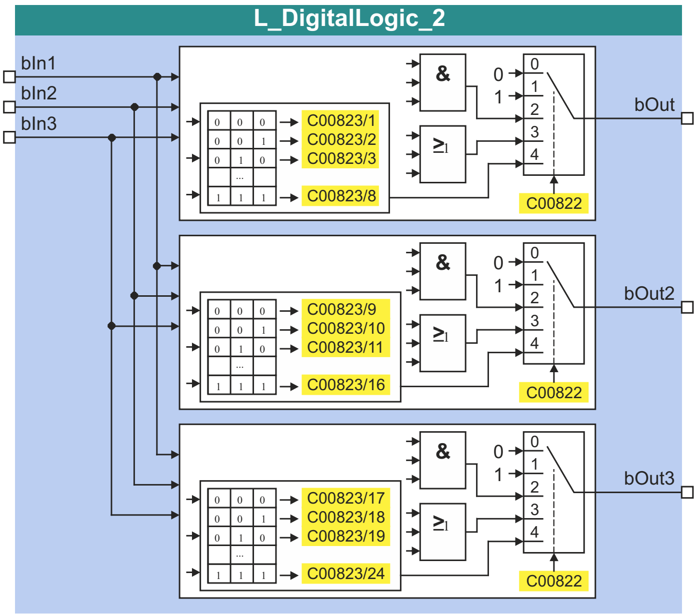
The following table shows which setting may be required in C00821/1...24 to realise the logic operations NAND, NOR, XOR and XNOR:

Input signals			Output				Parameter setting for logic operation			
bIn3	*bIn2*	*bIn1*	*bOut*	*bOut2*	*bOut3*		**NAND**	**NOR**	**XOR**	**XNOR**
0	0	0	C00821/1	C00821/9	C00821/17	=	1	1	0	1
0	0	1	C00821/2	C00821/10	C00821/18	=	1	0	1	0
0	1	0	C00821/3	C00821/11	C00821/19	=	1	0	1	0
0	1	1	C00821/4	C00821/12	C00821/20	=	1	0	0	1
1	0	0	C00821/5	C00821/13	C00821/21	=	1	0	1	0
1	0	1	C00821/6	C00821/14	C00821/22	=	1	0	0	1
1	1	0	C00821/7	C00821/15	C00821/23	=	1	0	0	1
1	1	1	C00821/8	C00821/16	C00821/24	=	0	0	0	1

19.1.82 L_DigitalLogic_2

This FB provides a binary output signal created by a logic operation of the input signals. Optionally, one of the constant binary values independent from the input signals can be output.

- Output of a constant binary value
- AND operation of the inputs
- OR operation of the inputs
- Output depending on the combination of the input signals



inputs

Designator Data type	Information/possible settings
bIn1 BOOL	Input signal 1
bIn2 BOOL	Input signal 2
bIn3 BOOL	Input signal 3

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal 1
bOut2 BOOL	Output signal 2 • bOut2 is available from version 16.00.00 onwards
bOut3 BOOL	Output signal 3 • bOut3 is available from version 16.00.00 onwards

Parameters

Parameters	Possible settings		Information
C00822			Function selection
	0	"0"	Constant value "FALSE"
	1	"1"	Constant value "TRUE"
	2	$bOut = bIn1 \wedge bIn2 \wedge bIn3$ $bOut2 = bIn1 \wedge bIn2 \wedge bIn3$ $bOut3 = bIn1 \wedge bIn2 \wedge bIn3$	AND operation • bOut2 and bOut3 are available from version 16.00.00 onwards
	3	$bOut = bIn1 \vee bIn2 \vee bIn3$ $bOut2 = bIn1 \vee bIn2 \vee bIn3$ $bOut3 = bIn1 \vee bIn2 \vee bIn3$	OR operation • bOut2 and bOut3 are available from version 16.00.00 onwards
	4	bOut = f (truth table) bOut2 = f (truth table) bOut3 = f (truth table)	The output value depends on the truth table parameterised in C00823/1...24 • bOut2 and bOut3 are available from version 16.00.00 onwards
C00823/1...24			Truth table for function "4: bOut = f (truth table)" • Each of the eight possible input combinations can be assigned to the output value FALSE or TRUE. • For an application example see the following section.
	0	FALSE	
	1	TRUE	

Function 4

If the function 4 is selected in C00822, the output values depend on the truth table parameterised in C00823/1...24.

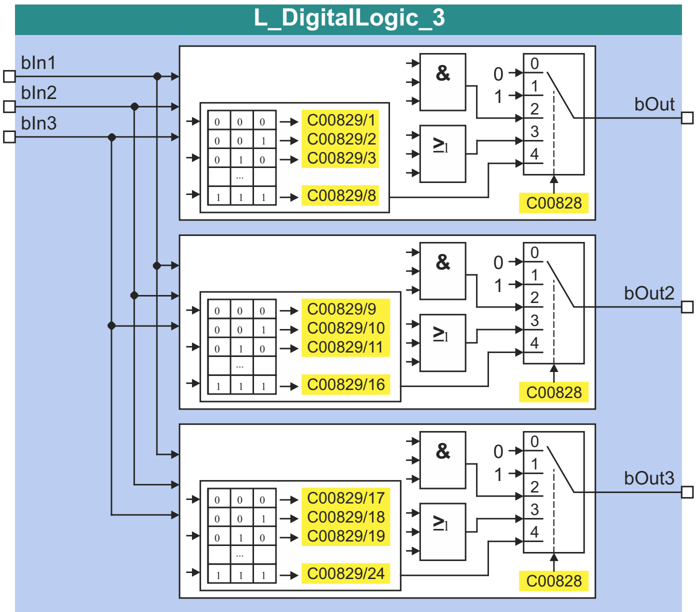
The following table shows which setting may be required in C00823/1...24 to realise the logic operations NAND, NOR, XOR and XNOR:

Input signals			Output				Parameter setting for logic operation			
bIn3	*bIn2*	*bIn1*	*bOut*	*bOut2*	*bOut3*		**NAND**	**NOR**	**XOR**	**XNOR**
0	0	0	C00823/1	C00823/9	C00823/17	=	1	1	0	1
0	0	1	C00823/2	C00823/10	C00823/18	=	1	0	1	0
0	1	0	C00823/3	C00823/11	C00823/19	=	1	0	1	0
0	1	1	C00823/4	C00823/12	C00823/20	=	1	0	0	1
1	0	0	C00823/5	C00823/13	C00823/21	=	1	0	1	0
1	0	1	C00823/6	C00823/14	C00823/22	=	1	0	0	1
1	1	0	C00823/7	C00823/15	C00823/23	=	1	0	0	1
1	1	1	C00823/8	C00823/16	C00823/24	=	0	0	0	1

19.1.83 L_DigitalLogic_3

This FB provides a binary output signal created by a logic operation of the input signals. Optionally, one of the constant binary values independent from the input signals can be output.

- Output of a constant binary value
- AND operation of the inputs
- OR operation of the inputs
- Output depending on the combination of the input signals



inputs

Designator Data type	Information/possible settings
bIn1 BOOL	Input signal 1
bIn2 BOOL	Input signal 2
bIn3 BOOL	Input signal 3

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal 1
bOut2 BOOL	Output signal 2 • bOut2 is available from version 16.00.00 onwards
bOut3 BOOL	Output signal 3 • bOut3 is available from version 16.00.00 onwards

Parameters

Parameters	Possible settings		Information
C00828			Function selection
	0	"0"	Constant value "FALSE"
	1	"1"	Constant value "TRUE"
	2	$bOut = bIn1 \wedge bIn2 \wedge bIn3$ $bOut2 = bIn1 \wedge bIn2 \wedge bIn3$ $bOut3 = bIn1 \wedge bIn2 \wedge bIn3$	AND operation • bOut2 and bOut3 are available from version 16.00.00 onwards
	3	$bOut = bIn1 \vee bIn2 \vee bIn3$ $bOut2 = bIn1 \vee bIn2 \vee bIn3$ $bOut3 = bIn1 \vee bIn2 \vee bIn3$	OR operation • bOut2 and bOut3 are available from version 16.00.00 onwards
	4	bOut = f (truth table) bOut2 = f (truth table) bOut3 = f (truth table)	The output value depends on the truth table parameterised in C00829/1...24 • bOut2 and bOut3 are available from version 16.00.00 onwards
C00829/1...24			Truth table for function "4: bOut = f (truth table)" • Each of the eight possible input combinations can be assigned to the output value FALSE or TRUE. • For an application example see the following section.
	0	FALSE	
	1	TRUE	

Function 4

If the function 4 is selected in C00828, the output values depend on the truth table parameterised in C00829/1...24.

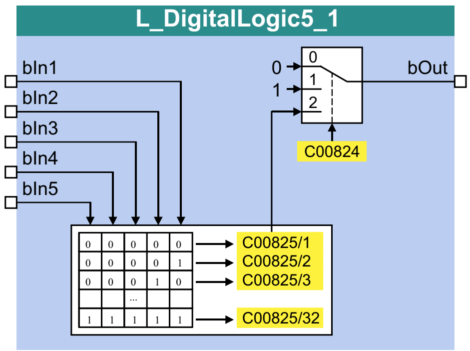
The following table shows which setting may be required in C00829/1...24 to realise the logic operations NAND, NOR, XOR and XNOR:

Input signals			Output				Parameter setting for logic operation			
bIn3	*bIn2*	*bIn1*	*bOut*	*bOut2*	*bOut3*		**NAND**	**NOR**	**XOR**	**XNOR**
0	0	0	C00829/1	C00829/9	C00829/17	=	1	1	0	1
0	0	1	C00829/2	C00829/10	C00829/18	=	1	0	1	0
0	1	0	C00829/3	C00829/11	C00829/19	=	1	0	1	0
0	1	1	C00829/4	C00829/12	C00829/20	=	1	0	0	1
1	0	0	C00829/5	C00829/13	C00829/21	=	1	0	1	0
1	0	1	C00829/6	C00829/14	C00829/22	=	1	0	0	1
1	1	0	C00829/7	C00829/15	C00829/23	=	1	0	0	1
1	1	1	C00829/8	C00829/16	C00829/24	=	0	0	0	1

19.1.84 L_DigitalLogic5_1

This FB provides a binary output signal created by a logic operation of the input signals. Optionally, one of the constant binary values independent from the input signals can be output.

- Output of a constant binary value
- Output depending on the combination of the input signals



inputs

Designator Data type	Information/possible settings
bIn1 ... bIn5 BOOL	Input signal 1 ... 5

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

Parameters

Parameters	Possible settings		Information
C00824			Function selection
	0	"0"	Constant value "FALSE"
	1	"1"	Constant value "TRUE"
	2	bOut = f (truth table)	The output value depends on the parameterised truth table
C00825	see truth table		Truth table Each of the 32 possible input combinations can be assigned to the output value FALSE or TRUE.

Truth table for C00824 = 4

bIn5	bIn4	bIn3	bIn2	bIn1	Output signal bOut
FALSE	FALSE	FALSE	FALSE	FALSE	C00825/1 (FALSE or TRUE)
FALSE	FALSE	FALSE	FALSE	TRUE	C00825/2 (FALSE or TRUE)

bIn5	bIn4	bIn3	bIn2	bIn1	Output signal bOut
FALSE	FALSE	FALSE	TRUE	FALSE	C00825/3 (FALSE or TRUE)
		...			C00825/... (FALSE or TRUE)
TRUE	TRUE	TRUE	FALSE	TRUE	C00825/30 (FALSE or TRUE)
TRUE	TRUE	TRUE	TRUE	FALSE	C00825/31 (FALSE or TRUE)
TRUE	TRUE	TRUE	TRUE	TRUE	C00825/32 (FALSE or TRUE)

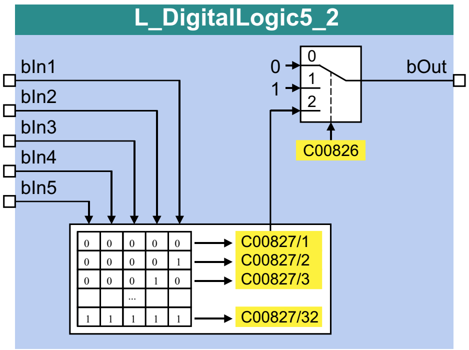
Example: If in case of the signal combination *bIn1* = TRUE, *bIn2* = FALSE, *bIn3* = TRUE, *bIn4* = TRUE and *bIn5* = TRUE, the output signal *bOut* is to be = TRUE, C00825/30 must be set to "TRUE":

bIn5	bIn4	bIn3	bIn2	bIn1	Output signal bOut
TRUE	TRUE	TRUE	FALSE	TRUE	C00825/30 (TRUE)

19.1.85 L_DigitalLogic5_2

This FB provides a binary output signal created by a logic operation of the input signals. Optionally, one of the constant binary values independent from the input signals can be output.

- Output of a constant binary value
- Output depending on the combination of the input signals



inputs

Designator Data type	Information/possible settings
bIn1 ... bIn5 BOOL	Input signal 1 ... 5

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

Parameters

Parameters	Possible settings		Information
C00826			Function selection
	0	"0"	Constant value "FALSE"
	1	"1"	Constant value "TRUE"
	2	bOut = f (truth table)	The output value depends on the parameterised truth table
C00827	see truth table		Truth table Each of the 32 possible input combinations can be assigned to the output value FALSE or TRUE.

Truth table for C00826 = 4

bIn5	bIn4	bIn3	bIn2	bIn1	Output signal bOut
FALSE	FALSE	FALSE	FALSE	FALSE	C00827/1 (FALSE or TRUE)
FALSE	FALSE	FALSE	FALSE	TRUE	C00827/2 (FALSE or TRUE)

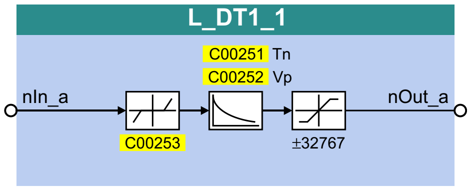
bIn5	bIn4	bIn3	bIn2	bIn1	Output signal bOut
FALSE	FALSE	FALSE	TRUE	FALSE	C00827/3 (FALSE or TRUE)
		...			C00827/... (FALSE or TRUE)
TRUE	TRUE	TRUE	FALSE	TRUE	C00827/30 (FALSE or TRUE)
TRUE	TRUE	TRUE	TRUE	FALSE	C00827/31 (FALSE or TRUE)
TRUE	TRUE	TRUE	TRUE	TRUE	C00827/32 (FALSE or TRUE)

Example: If in case of the signal combination *bIn1* = TRUE, *bIn2* = FALSE, *bIn3* = TRUE, *bIn4* = TRUE and *bIn5* = TRUE, the output signal *bOut* is to be = TRUE, C00827/30 must be set to "TRUE":

bIn5	bIn4	bIn3	bIn2	bIn1	Output signal bOut
TRUE	TRUE	TRUE	FALSE	TRUE	C00827/30 (TRUE)

19.1.86 L_DT1_1

This FB differentiates signals. The function block can be used for acceleration feedforward (dv/dt), for example.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

outputs

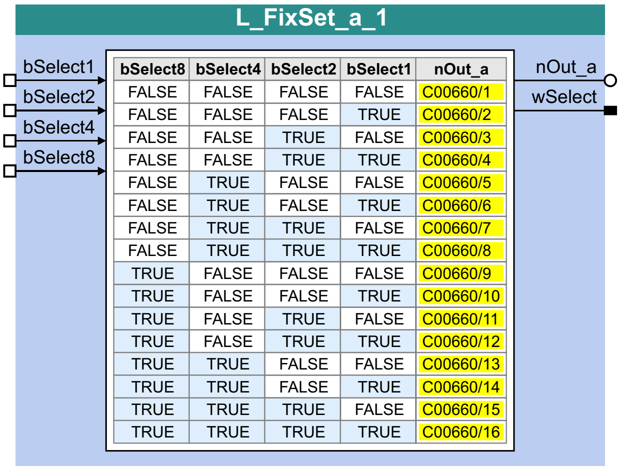
Designator Data type	Value/meaning
nOut_a INT	Output signal

Parameters

Parameters	Possible settings			Information
C00251	10	ms	5000	Time constant Tn • Lenze setting: 1000 ms
C00252	-320.00		320.00	Gain factor Vp • Lenze setting: 1.00
C00253	**Note:** The most significant bit determines the sign of the value, the remaining bits determine the numerical value.			Selection of sensitivity • Depending on the selection, the number of indicated higher-order bits is evaluated. • Lenze setting: 15 bits
	1	15 bits		Bit 0 ... bit 14 are evaluated
	2	14 Bit		Bit 0 ... bit 13 are evaluated
	3	13 bits		Bit 0 ... bit 12 are evaluated
	4	12 bits		Bit 0 ... bit 11 are evaluated
	5	11 Bit		Bit 0 ... bit 10 are evaluated
	6	10 Bit		Bit 0 ... bit 9 are evaluated
	7	9 Bit		Bit 0 ... bit 8 are evaluated

19.1.87 L_FixSet_a_1

This FB outputs one of 16 parameterisable analog signals. Binary coded selection of the "fixed value" to be output via the four selection inputs.



bSelect8	bSelect4	bSelect2	bSelect1	nOut_a
FALSE	FALSE	FALSE	FALSE	C00660/1
FALSE	FALSE	FALSE	TRUE	C00660/2
FALSE	FALSE	TRUE	FALSE	C00660/3
FALSE	FALSE	TRUE	TRUE	C00660/4
FALSE	TRUE	FALSE	FALSE	C00660/5
FALSE	TRUE	FALSE	TRUE	C00660/6
FALSE	TRUE	TRUE	FALSE	C00660/7
FALSE	TRUE	TRUE	TRUE	C00660/8
TRUE	FALSE	FALSE	FALSE	C00660/9
TRUE	FALSE	FALSE	TRUE	C00660/10
TRUE	FALSE	TRUE	FALSE	C00660/11
TRUE	FALSE	TRUE	TRUE	C00660/12
TRUE	TRUE	FALSE	FALSE	C00660/13
TRUE	TRUE	FALSE	TRUE	C00660/14
TRUE	TRUE	TRUE	FALSE	C00660/15
TRUE	TRUE	TRUE	TRUE	C00660/16

inputs

Designator Data type	Information/possible settings
bSelect1 ... bSelect8 BOOL	Binary coded selection of the fixed value to be output • See truth table displayed in the FB above.

outputs

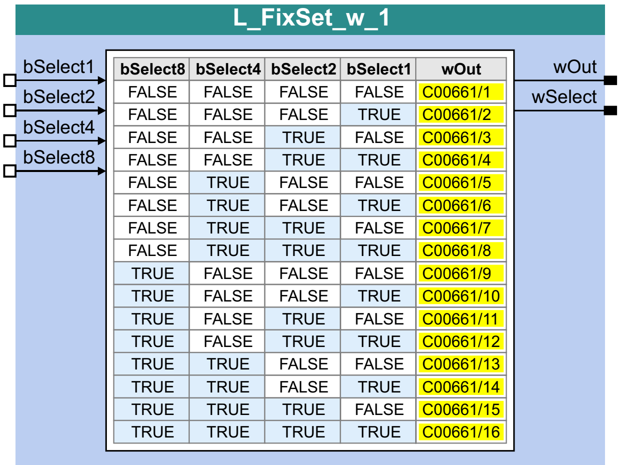
Designator Data type	Value/meaning
nOut_a INT	Output of the selected fixed value
wSelect WORD	Current selection (0 ... 15)

Parameters

Parameters	Possible settings			Information
C00660/1...16	-199.99	%	199.99	Fixed value 0 ... 15 • Lenze setting: 0.00 %

19.1.88 L_FixSet_w_1

This FB outputs one of 16 parameterisable data words. Binary coded selection of the "fixed value" to be output via the four selection inputs.



bSelect8	bSelect4	bSelect2	bSelect1	wOut
FALSE	FALSE	FALSE	FALSE	C00661/1
FALSE	FALSE	FALSE	TRUE	C00661/2
FALSE	FALSE	TRUE	FALSE	C00661/3
FALSE	FALSE	TRUE	TRUE	C00661/4
FALSE	TRUE	FALSE	FALSE	C00661/5
FALSE	TRUE	FALSE	TRUE	C00661/6
FALSE	TRUE	TRUE	FALSE	C00661/7
FALSE	TRUE	TRUE	TRUE	C00661/8
TRUE	FALSE	FALSE	FALSE	C00661/9
TRUE	FALSE	FALSE	TRUE	C00661/10
TRUE	FALSE	TRUE	FALSE	C00661/11
TRUE	FALSE	TRUE	TRUE	C00661/12
TRUE	TRUE	FALSE	FALSE	C00661/13
TRUE	TRUE	FALSE	TRUE	C00661/14
TRUE	TRUE	TRUE	FALSE	C00661/15
TRUE	TRUE	TRUE	TRUE	C00661/16

inputs

Designator Data type	Information/possible settings
bSelect1 ... bSelect8 BOOL	Binary coded selection of the fixed value to be output • See truth table displayed in the FB above.

outputs

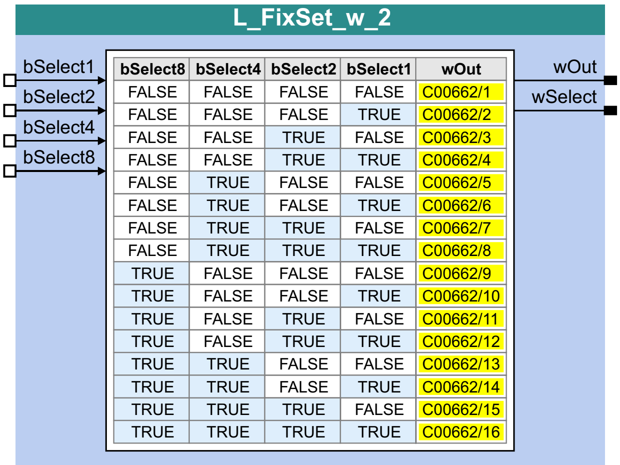
Designator Data type	Value/meaning
wOut WORD	Output of the selected fixed value
wSelect WORD	Current selection (0 ... 15)

Parameters

Parameters	Possible settings			Information
C00661/1...16	0		65535	Fixed values 0 ... 15 • Lenze setting: 0

19.1.89 L_FixSet_w_2

This FB outputs one of 16 parameterisable data words. Binary coded selection of the "fixed value" to be output via the four selection inputs.



bSelect8	bSelect4	bSelect2	bSelect1	wOut
FALSE	FALSE	FALSE	FALSE	C00662/1
FALSE	FALSE	FALSE	TRUE	C00662/2
FALSE	FALSE	TRUE	FALSE	C00662/3
FALSE	FALSE	TRUE	TRUE	C00662/4
FALSE	TRUE	FALSE	FALSE	C00662/5
FALSE	TRUE	FALSE	TRUE	C00662/6
FALSE	TRUE	TRUE	FALSE	C00662/7
FALSE	TRUE	TRUE	TRUE	C00662/8
TRUE	FALSE	FALSE	FALSE	C00662/9
TRUE	FALSE	FALSE	TRUE	C00662/10
TRUE	FALSE	TRUE	FALSE	C00662/11
TRUE	FALSE	TRUE	TRUE	C00662/12
TRUE	TRUE	FALSE	FALSE	C00662/13
TRUE	TRUE	FALSE	TRUE	C00662/14
TRUE	TRUE	TRUE	FALSE	C00662/15
TRUE	TRUE	TRUE	TRUE	C00662/16

inputs

Designator Data type	Information/possible settings
bSelect1 ... bSelect8 BOOL	Binary coded selection of the fixed value to be output • See truth table displayed in the FB above.

outputs

Designator Data type	Value/meaning
wOut WORD	Output of the selected fixed value
wSelect WORD	Current selection (0 ... 15)

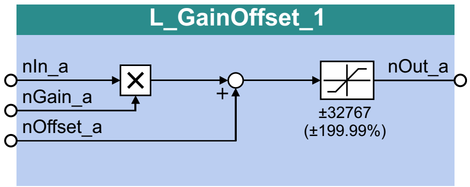
Parameters

Parameters	Possible settings			Information
C00662/1...16	0		65535	Fixed values 0 ... 15 • Lenze setting: 0

19.1.90 L_GainOffset_1

This FB can amplify an analog input signal and add an offset to it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via FB inputs.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %
nGain_a INT	Gain factor • Scaling: 16384 ≡ 100 % • 199.99 % ≈2
nOffset_a INT	Offset • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

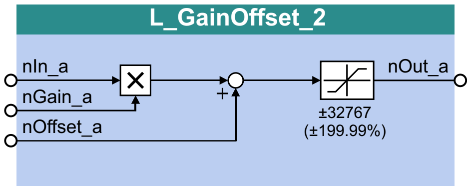
Function

$$\text{nOut\_a} = (\text{nIn\_a} \cdot \text{Gain factor}) + \text{Offset}$$

19.1.91 L_GainOffset_2

This FB can amplify an analog input signal and add an offset to it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via FB inputs.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %
nGain_a INT	Gain factor • Scaling: 16384 ≡ 100 % • 199.99 % ≈2
nOffset_a INT	Offset • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

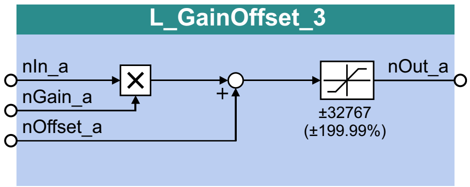
Function

$$nOut\_a = (nIn\_a \cdot \text{Gain factor}) + \text{Offset}$$

19.1.92 L_GainOffset_3

This FB can amplify an analog input signal and add an offset to it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via FB inputs.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %
nGain_a INT	Gain factor • Scaling: 16384 ≡ 100 % • 199.99 % ≈2
nOffset_a INT	Offset • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

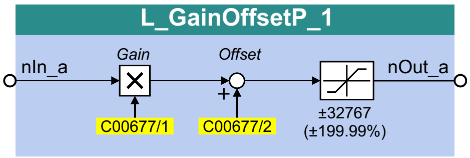
Function

$$\text{nOut\_a} = (\text{nIn\_a} \cdot \text{Gain factor}) + \text{Offset}$$

19.1.93 L_GainOffsetP_1

This FB can amplify an analog input signal and add an offset to it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via parameters.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator / Data type	Information/possible settings
nIn_a / INT	Input signal • Scaling: 16384 ≡ 100 %

outputs

Designator / Data type	Value/meaning
nOut_a / INT	Output signal • Internal limitation to ±199.99 %

Parameters

Parameters	Possible settings			Information
C00677/1	-199.99	%	199.99	Gain factor • Lenze setting: 100.00 % • 199.99 % ≈2
C00677/2	-199.99	%	199.99	Offset • Lenze setting: 0.00 %

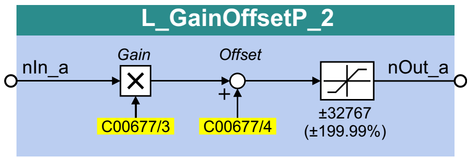
Function

$$nOut\_a = (nIn\_a \cdot \text{Gain factor}) + \text{Offset}$$

19.1.94 L_GainOffsetP_2

This FB can amplify an analog input signal and add an offset to it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via parameters.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

Parameters

Parameters	Possible settings			Information
C00677/3	-199.99	%	199.99	Gain factor • Lenze setting: 100.00 % • 199.99 % ≈2
C00677/4	-199.99	%	199.99	Offset • Lenze setting: 0.00 %

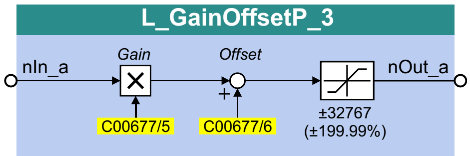
Function

$$\text{nOut\_a} = (\text{nIn\_a} \cdot \text{Gain factor}) + \text{Offset}$$

19.1.95 L_GainOffsetP_3

This FB can amplify an analog input signal and add an offset to it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via parameters.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

Parameters

Parameters	Possible settings			Information
C00677/5	-199.99	%	199.99	Gain factor • Lenze setting: 100.00 % • 199.99 % ≈ 2
C00677/6	-199.99	%	199.99	Offset • Lenze setting: 0.00 %

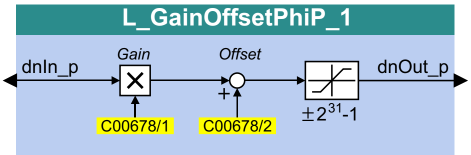
Function

$$\text{nOut\_a} = (\text{nIn\_a} \cdot \text{Gain factor}) + \text{Offset}$$

19.1.96 L_GainOffsetPhiP_1

This FB can amplify an angle signal and add an offset to it afterwards.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via parameters.
- The value provided at the *dnOut_p* output is internally limited to $\pm 2^{31}$-1.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Parameters

Parameters	Possible settings			Information
C00678/1	-2147483647	Incr.	2147483647	Offset • Lenze setting: 0 incr.
C00678/2	-2147483647		2147483647	Gain factor • Lenze setting: 65536

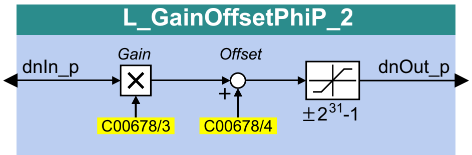
Function

$$\mathrm{dnOut\_p} = (\mathrm{dnIn\_p} \cdot \mathrm{Gain\ factor}) + \mathrm{Offset}$$

19.1.97 L_GainOffsetPhiP_2

This FB can amplify an angle signal and add an offset to it afterwards.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Gain and offset are selected via parameters.
- The value provided at the *dnOut_p* output is internally limited to $\pm 2^{31}$-1.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Parameters

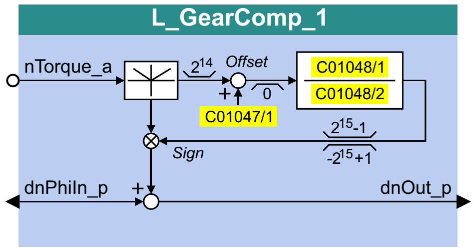
Parameters	Possible settings			Information
C00678/3	-2147483647	Incr.	2147483647	Offset • Lenze setting: 0 incr.
C00678/4	-2147483647		2147483647	Gain factor • Lenze setting: 65536

Function

$$\text{dnOut\_p} = (\text{dnIn\_p} \cdot \text{Gain factor}) + \text{Offset}$$

19.1.98 L_GearComp_1

This FB is used for dynamic compensation of elasticities in the drive train that arise, e.g. by elastic coupling of speed reduction gearboxes or long transmission shafts.



inputs

Designator Data type	Information/possible settings
nTorque_a INT	Torque demand • 16384 ≡ 100 %
dnPhiIn_p DINT	Angle setpoint • 65536 increments ≡ 1 motor revolution

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Corrected angle setpoint • 65536 increments ≡ 1 motor revolution

Parameters

Parameters	Possible settings				Information
C01047/1	-16383			16383	Offset • Static torque offset • Lenze setting: 0
C01048/1	-32767			32767	Meters • Elastic constant • Lenze setting: 1
C01048/2	1			32767	Denominator • Elastic constant • Lenze setting: 1

Function

Elasticity is a measure of how far the load have moved from the ideal setpoint position at motor standstill due to mechanical force effect.

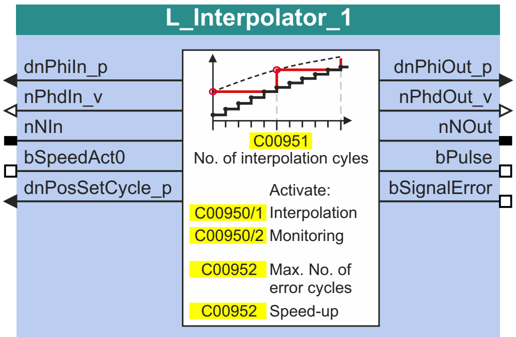
- Example: "Hoist":
  Due to elasticity of the mechanical transmission elements, the real position of the "hook" varies in loaded and unloaded status.
- In order to compensate errors caused by elasticity, the actual torque at the *nTorque_a* input is applied. This torque is a measure for the current load.
- The multiplication by an elasticity factor results in an angle compensation value which is added to the setpoint angle correctly signed depending on the direction of the torque. This serves to correct the false position of the load.
- The elasticity factor is selected in the form of numerator and denominator via C01048/1 and C01048/2.
- C01047/1 serves to select a static correction value (offset).

Note!

Please note that the compensation may be reversed in case of a signed numerator/denominator selection of the elasticity factor!

19.1.99 L_Interpolator_1

This FB interpolates a position setpoint and/or an analog value e.g. to compensate for larger bus transmission cycles or to continue signal characteristics if data telegrams are missing.



inputs

Designator Data type		Information/possible settings	
dnPhIn_p	DINT	Position setpoint • Is interpolated and completed when signal interpolation is activated.	
nPhdIn_v	INT	Angular velocity • Is only passed through to the *nPhdOut_v* output.	
nNIn	INT	Analog value • Is interpolated when signal interpolation is activated.	
bSpeedAct0	BOOL	Input for detecting the "Current speed is zero" status • This status signal needs to be transmitted by the setpoint source to ensure trouble-free operation.	
		TRUE	The current speed is zero. The interpolator in the follow-up controller is kept to the *dnPhiIn_p* input. • This function is for instance required for referencing the axis. Since in this case, the axis can move independently of the master position, the follow-up controller is not allowed to intervene.
dnPosSetCycle_p	DINT	Master cycle length • By specifying the master cycle length, the overflows of the Modulo signal are processed correctly in the "position follower" mode.	

outputs

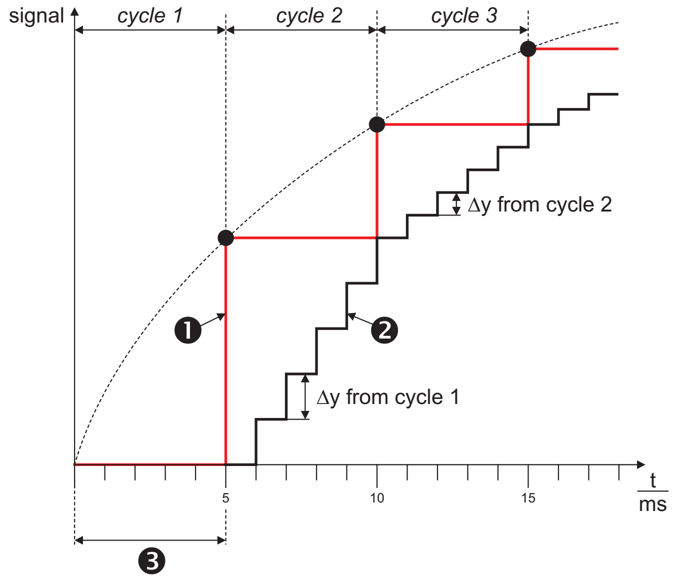
Designator Data type		Value/meaning	
dnPhOut_p	DINT	Output of the *dnPhIn_p* position setpoint which, if applicable, has been interpolated and completed	
nPhdOut_v	INT	Output of the *nPhdIn_v* angular velocity	
nNOut	INT	Output of the *nNIn* analog value which, if applicable, has been interpolated	
bPulse	BOOL	"Input values have been accepted" status signal	
		TRUE	The input values have been accepted during this cycle.
bSignalError	BOOL	"Signal error" status signal • Only if signal monitoring is active (C00950/2 = "1: On").	
		TRUE	The number of missing data telegrams has exceeded the limit value parameterised in C00952.

Parameters

Parameters	Possible settings			Information
C00950/1				Signal interpolation of the *dnPhIn_p* and *nNIn* input signals • Lenze setting: Off ▸ Signal interpolation (📖 1629)
	0	Off		
	1	On		
C00950/2				Signal monitoring of the *dnPhIn_p* input signal • Lenze setting: Off ▸ Signal monitoring (📖 1630)
	0	Off		
	1	On		
C00951	1		65535	Number of interpolation steps • Corresponds to the receive cycle of the data telegrams in [ms]. • Lenze setting: 1
C00952	0		65535	Limit value for missing data telegrams • Lenze setting: 5 ▸ Signal monitoring (📖 1630)
C00953	0		100	Speed-up • Limitation of the amount of correction increments per cycle • Scaling: 1 increment/ms ≡ 0.9155 rpm

19.1.99.1 Signal interpolation

If signal interpolation is active (C00950/1 = 1), the output signal will not reach the level of the corresponding input signal until all interpolation steps parameterised in C00951 have been performed:



❶ Setpoint signal
❷ Interpolated setpoint signal
❸ Receive cycle in [ms] = Number of interpolation steps (in our example: 5)

[19-44] Signal characteristic

Note!

Do not change the number of interpolation steps during operation. Otherwise the interpolation becomes inaccurate.

19.1.99.2 Signal monitoring

If signal monitoring is active (C00950/2 = 1), the signal characteristic of the *dnPhIn_p* input signal is continued even if the data telegram is missing (setpoint selection via CAN).

Monitoring is performed on the basis of the *dnPhIn_p* position setpoint and the *bSpeedAct0* status signal:

- If the *dnPhIn_p* position setpoint remains the same in the next device cycle, it is either because the speed is zero or because no data telegram has been received.
- The evaluation of the *bSpeedAct0* status signal gives information about which reason applies. This status signal needs to be transmitted by the setpoint source to ensure trouble-free operation:
  - *bSpeedAct0* = FALSE means that the speed is not zero, so an error is assumed: The signal characteristic of the *dnPhIn_p* input signal is completed (the current slope is retained).
  - *bSpeedAct0* = TRUE means that the speed is zero, so the unchanged position setpoint is not treated as an error.
- If the number of missing data telegrams exceeds the limit value parameterised in C00952, the *bSignalError* output is set to TRUE.
  - The *bSignalError* output is automatically reset to FALSE if correct signals are detected at *dnPhIn_p* and *bSpeedAct0* again.

Note!

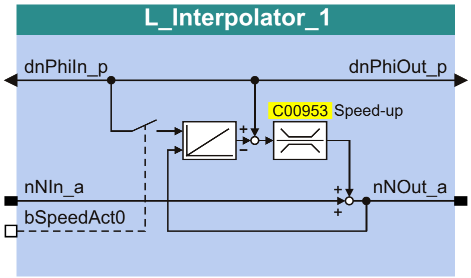
The analog value *nNIn* is not monitored!

19.1.99.3 Angle correction in case of transmission errors

If an angular offset between master and slave is caused due to transmission errors (missing data telegrams), it will be corrected by a catch-up function in the FB. For this purpose, the following connections and parameter settings are required for the slave at the FB **L_Interpolator_1**:

1. The master angle of the master is connected to the *dnPhiIn_p* input.
2. The speed signal of the master is connected to the *nNIn* input.
3. The *nNOut* output is connected to the *nSet_v* input of the FB L_DFSET_1.

The following illustration shows the principle of the catch-up function in the FB **L_Interpolator_1**:



[19-45] Principle of the catch-up function

The speed signal at *nNIn* is provided almost 1:1 at the *nNOut* output. In case of a telegram error, a correction value can be added to the signal. This correction value results from the subtraction of the integrated speed signal from the position value applied at *dnPhiIn_p*.

If, for example, a data telegram should fail, the input values remain constant for one program cycle. During the next cycle, the correct position and the correct speed are restored.

"Holding" the position at *dnPhiIn_p* results in a difference between the position values at *dnPhiIn_p* and at the output of the integrator. This difference is added to the *nNOut_a* output signal.

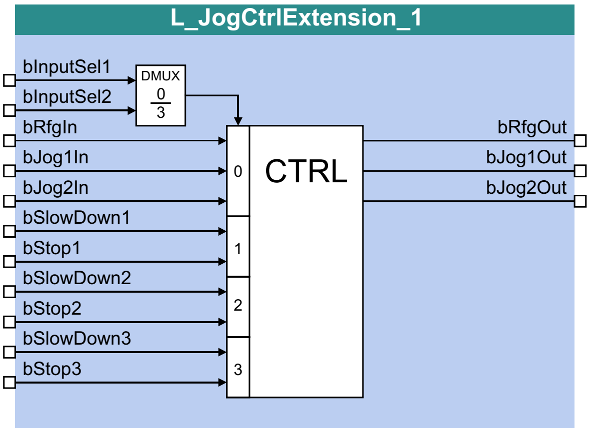
In order that this angle correction does not lead to a strong jerk in the master value, the amount of correction increments is limited per cycle (catch-up cycle) using C00953. A typical correction value is for instance 10 increments/ms.

When the controller is inhibited, the integrator is to be loaded with the position value pending at *dnPhiIn_p* by setting the *bSpeedAct0* input to TRUE. When the controller is enabled, the *nNOut* speed signal is integrated.

19.1.100 L_JogCtrlExtension_1

This FB can be connected upstream to the L_NSet ramp function generator/setpoint generator to implement a switch-off positioning at limit switch.

- Detailed information on this operating mode can be found in the description
  ▸ TA "Switch-off positioning" (📖 544).



inputs

Designator Data type	Information/possible settings
bInputSel1 bInputSel2 BOOL	Activation of the *bSlowDown1/bStop1*, *bSlowDown2/bStop2* and *bSlowDown3/bStop3* signal pairs according to the Truth table
bRfgIn BOOL	Ramping down of the setpoint generator in the downstream L_NSet FB according to the Truth table
bJog1In bJog2In BOOL	Selection inputs for setting fixed speeds in the setpoint generator • If the pre-switch off is inactive (*bInputSel1* and *bInputSel2* are both set to FALSE), the two control signals are output one-to-one at the *bJog1Out* and *bJog2Out* outputs. • To achieve the desired behaviour (starting at high speed, pre-switch off at low speed), both inputs must be set to TRUE. • Fixed setpoint 2 must be less than fixed setpoint 3! Otherwise, the drive will start at a low speed and accelerate after the pre-switch off. • If, in addition to the *bJog1In* and *bJog2In* inputs, other jog signals are set at the L_NSet FB, new fixed setpoints are reached, and the drive traverses at speeds that differ from the selection via *bJog1In* and *bJog2In*.
bSlowDown1 bSlowDown2 bSlowDown3 BOOL	Activation of fixed setpoint 2 in the downstream L_NSet FB • These inputs only fulfil a function if they have been activated via *bInputSel1* and *bInputSel2* previously (see Truth table).
bStop1 bStop2 bStop3 BOOL	Ramping down of the ramp function generator in the downstream L_NSet FB • These inputs only fulfil a function if they have been activated via *bInputSel1* and *bInputSel2* previously (see Truth table).

outputs

Designator Data type	Value/meaning
bRfgOut BOOL	Control signal for ramping down the setpoint generator • Connect this output to the *bRfg0* input of the L_NSet FB.
bJog1Out BOOL	Control signal for setting fixed speeds in the setpoint generator • Connect this output to the *bJog1* input of the L_NSet FB.
bJog2Out BOOL	Control signal for setting fixed speeds in the setpoint generator • Connect this output to the *bJog2* input of the L_NSet FB.

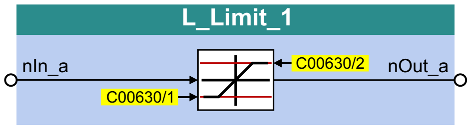
Truth table

<table>
<tr><th colspan="2">Input</th><th rowspan="2">Function</th><th rowspan="2">Response in the L_NSet FB</th></tr>
<tr><th>bInputSel1</th><th>bInputSel2</th></tr>
<tr><td>FALSE</td><td>FALSE</td><td>Pre-switch off inactive</td><td>No response
• The bRfgIn input signal is directly provided at the bRfgOut output.
• The bJogIn1 and bJogIn2 input signals are directly output at the bJog1Out and bJog2Out outputs.</td></tr>
<tr><td>TRUE</td><td>FALSE</td><td>The bSlowDown1 and bStop1 inputs are evaluated.</td><td rowspan="3">Pre-switch off can be activated
• If the SlowDown function is activated via the selected bSlowDown input, fixed setpoint 2 in the setpoint generator is activated via the bJog1Out and bJog2Out outputs.
• If the Stop function is activated via the selected bStop input, the bRfgOut output is set to TRUE and hence the setpoint generator is deactivated.</td></tr>
<tr><td>FALSE</td><td>TRUE</td><td>The bSlowDown2 and bStop2 inputs are evaluated.</td></tr>
<tr><td>TRUE</td><td>TRUE</td><td>The bSlowDown3 and bStop3 inputs are evaluated.</td></tr>
</table>

[19-1] Truth table for activating the pre-switch off

19.1.101 L_Limit_1

This FB limits an analog input signal to a value range whose upper and lower limit can be set via parameters.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Scaling: 16384 ≡ 100 %

Parameters

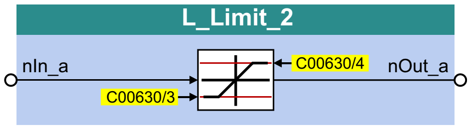
Parameters	Possible settings			Information
C00630/1	-199.99	%	199.99	Lower limit • Lenze setting: -100.00 %
C00630/2	-199.99	%	199.99	Upper limit • Lenze setting: 100.00 %

Tip!

Always set the lower limit lower than the upper limit, otherwise value "0" is provided at the *nOut_a* output.

19.1.102 L_Limit_2

This FB limits an analog input signal to a value range whose upper and lower limit can be set via parameters.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Scaling: 16384 ≡ 100 %

Parameters

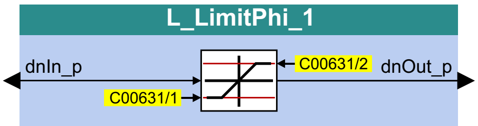
Parameters	Possible settings			Information
C00630/3	-199.99	%	199.99	Lower limit • Lenze setting: -100.00 %
C00630/4	-199.99	%	199.99	Upper limit • Lenze setting: 100.00 %

Tip!

Always set the lower limit lower than the upper limit, otherwise value "0" is provided at the *nOut_a* output.

19.1.103 L_LimitPhi_1

This FB restricts an angle signal to one value range, whose upper and lower limit can be set via parameters.



inputs

Designator Data type	Information/possible settings
dnIn_p INT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p INT	Output signal

Parameters

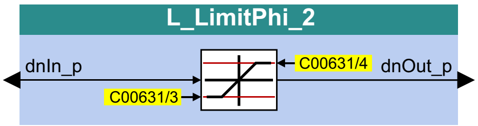
Parameters	Possible settings			Information
C00631/1	-2147483647	Incr.	2147483647	Lower limit • Lenze setting: -2147483647 incr.
C00631/2	-2147483647	Incr.	2147483647	Upper limit • Lenze setting: 2147483647 incr.

Tip!

Always set the lower limit lower than the upper limit, otherwise value "0" is provided at the *dnOut_p* output.

19.1.104 L_LimitPhi_2

This FB restricts an angle signal to one value range, whose upper and lower limit can be set via parameters.



inputs

Designator Data type	Information/possible settings
dnIn_p INT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p INT	Output signal

Parameters

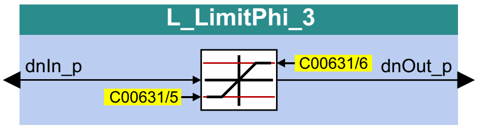
Parameters	Possible settings			Information
C00631/3	-2147483647	Incr.	2147483647	Lower limit • Lenze setting: -2147483647 incr.
C00631/4	-2147483647	Incr.	2147483647	Upper limit • Lenze setting: 2147483647 incr.

Tip!

Always set the lower limit lower than the upper limit, otherwise value "0" is provided at the *dnOut_p* output.

19.1.105 L_LimitPhi_3

This FB restricts an angle signal to one value range, whose upper and lower limit can be set via parameters.



inputs

Designator Data type	Information/possible settings
dnIn_p INT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p INT	Output signal

Parameters

Parameters	Possible settings			Information
C00631/5	-2147483647	Incr.	2147483647	Lower limit • Lenze setting: -2147483647 incr.
C00631/6	-2147483647	Incr.	2147483647	Upper limit • Lenze setting: 2147483647 incr.

Tip!

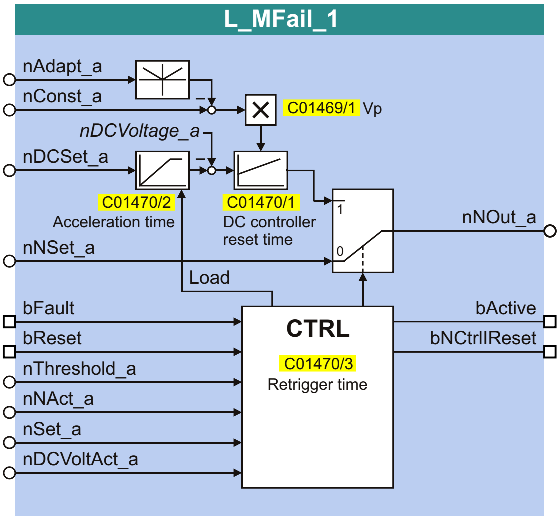
Always set the lower limit lower than the upper limit, otherwise value "0" is provided at the *dnOut_p* output.

19.1.106 L_MFail_1

In case the supply voltage fails, this FB serves to stop (brake) the drive or drive system in a controlled way via L1, L2, L3 or +UG, -UG to prevent coasting.

Stop!

For drive systems that are coupled via digital frequency (a master drive and one or several slave drives) the following applies:
- The mains failure detection and control may only be activated for the master drive.
- All inverters must be operated via the terminals +UG, -UG in the DC bus connection.



inputs

Designator / Data type		Information/possible settings
nAdapt_a	INT	Dynamic adaptation of the proportional gain of the DC-bus voltage controller • The resulting proportional gain results from: $Vp = C01469/1 \cdot \frac{nConst\_a\,[\%] - \lvert nAdapt\_a\,[\%]\rvert}{100\,\%}$ • Scaling: 16384 ≡ 100 %
nConst_a	INT	
nDCSet_a	INT	Voltage setpoint on which the DC-bus voltage is to be kept. • Scaling: 16384 ≡ 1000 V
nNset_a	INT	Speed setpoint in [%] • Scaling: 16384 ≡ 100 % reference speed (C00011)
bFault	BOOL	Activating the mains failure control TRUE: Activate mains failure control.
bReset	BOOL	Reset mains failure control • A reset is always required when the restart protection is active or the restart protection is used and the supply (mains or DC supply) has been switched on. TRUE: Reset mains failure control.
nThreshold_a	INT	Restart threshold in [%] • Scaling: 16384 ≡ 100 % reference speed (C00011)

Designator Data type	Information/possible settings
nNAct_a INT	Comparison value for the restart threshold in [%] • Scaling: 16384 ≡ 100 % reference speed (C00011)
nSet_a INT	Speed starting point for the deceleration in [%] • Scaling: 16384 ≡ 100 % reference speed (C00011)
nDCVoltAct_a INT	Current DC-bus voltage • Scaling: 16384 ≡ 1000 V

outputs

Designator Data type	Value/meaning	
nNOut_a INT	Speed setpoint in [%] • Scaling: 16384 ≡ 100 % reference speed (C00011)	
bActive BOOL	Status signal "Mains failure control active"	
	TRUE	The mains failure control is active.
bNCtrlReset BOOL	Control signal for resetting the I component of the speed controller in the motor control • Connect this output with the *bSpeedCtrlIOn* input of the SB LS_MotionControlKernel (if this is connected upstream to the motor control).	
	TRUE	Reset I component of the speed controller of the motor control.

Parameters

Parameters	Possible settings			Information
C01469/1	0.001		31.000	Vp • Lenze setting: 1.000
C01470/1	0	ms	60000	Vdc-bus controller reset time • Lenze setting: 20 ms
	Note: The set value is internally limited to 20 ... 2000 ms! From version 14.00.00 onwards, the limited value is written back to the code if the setting is outside the limits.			
C01470/2	0	ms	60000	Acceleration time • Lenze setting: 20 ms
	Note: The set value is internally set to 1 ... 16000 ms! From version 14.00.00 onwards, the limited value is written back to the code if the setting is outside the limits.			
C01470/3	0	ms	60000	Retrigger time • Lenze setting: 20 ms

19.1.106.1 Procedure of the mains failure control

A failure of the voltage supply of the power section can be detected by

- an evaluation of the DC-bus voltage
  and/or
- an external monitoring system (e.g. voltage measuring relay).

The type of mains failure control to be used depends on the used drive system.

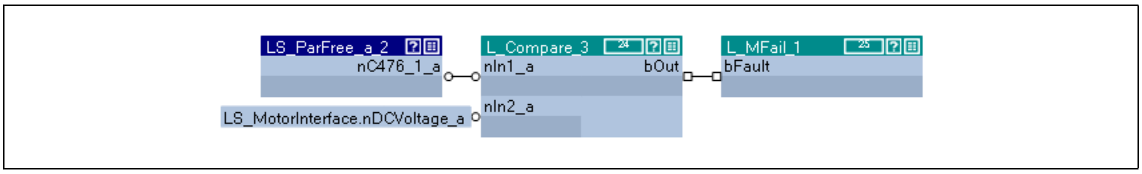
Note!

The following interconnection examples are not functional yet. For an error-free function, connect the FB **L_MFail_1** with further signals!

Evaluation of the DC-bus voltage

This proceeding is used for single drives or multi-axis drives which do not use an external monitoring system.

- For evaluating the DC-bus voltage, you can use a comparator (e.g. FB L_Compare_3) as shown in the following interconnection example 1.
  - In order that the mains failure control will be activated if the DC-bus voltage falls below a certain operating threshold, set the comparison function "2: In1 > In2" has to be set for the FB L_Compare_3 in C00690.
  - For specifying the operating threshold, the SB LS_ParFree_a_2 is used. This system block can output 16 parameterisable analog signals. In the shown example, the operating threshold has to be set in the "free parameter" C00476/1.

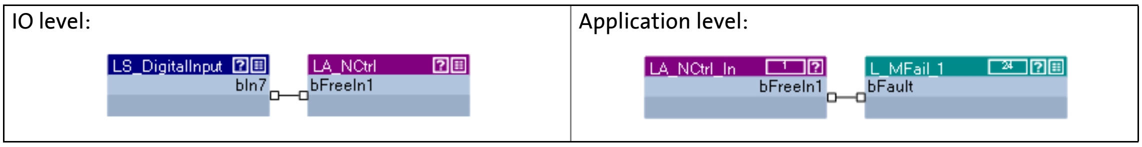


[19-46] Interconnection example 1: Evaluation of the DC-bus voltage with a comparator (cutout)

Use of an external monitoring system

In this proceeding, the digital status signal of an external monitoring system is connected to the FB **L_MFail_1** via a digital input of the inverter.

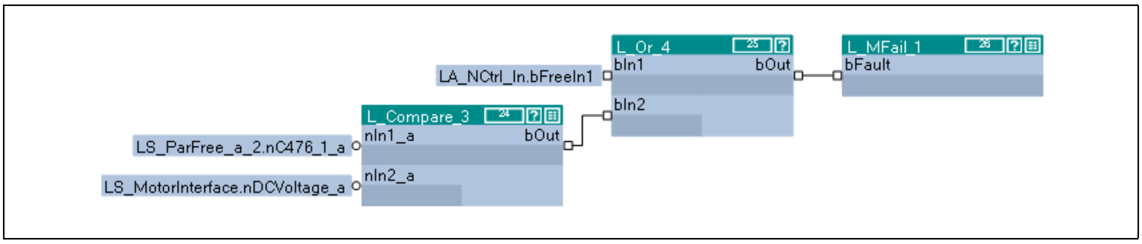
- In the following example, the digital input DI7 is used.
  - The active level (HIGH or LOW active) for DI7 has to be parameterised in C00114 in such a way that *bFault* becomes TRUE when the monitoring system is tripped.
  - A free input of the application block can be used to transfer the digital input signal from I/O level to application level.



[19-47] Interconnection example 2: Use of the digital status signal of an external monitoring system

Combination of the above shown proceedings

The combination of the proceedings can be simply realised via an OR operation:



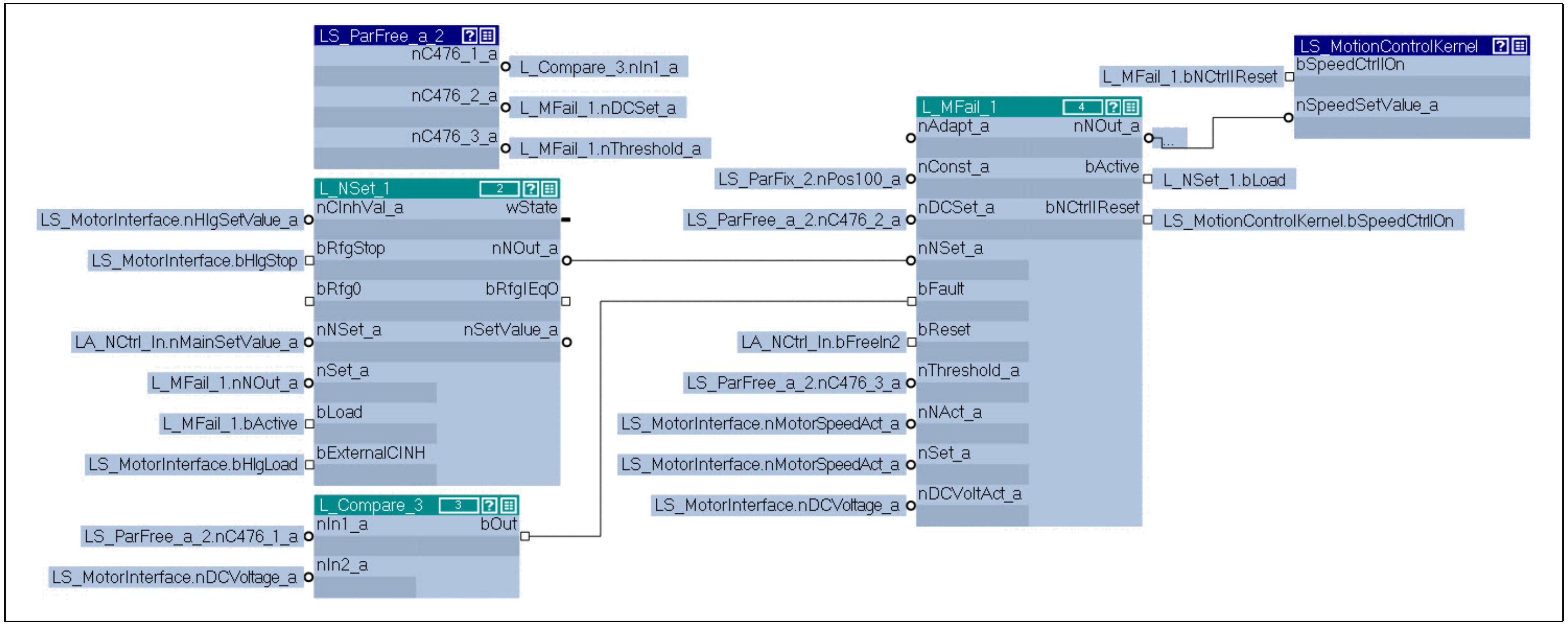
[19-48] Interconnection example 3: Mains failure control of different sources

19.1.106.2 Signal flow for mains failure control

The following illustration shows the basic signal flow (cutout) for a mains failure control.

Note!

The following interconnection example is not functional yet. For an error-free function, connect e.g. the FB L_NSet_1 with further signals!



[19-49] Signal flow (detail)

L_MFail inputs	Function
nAdapt_a	Dynamic adaptation of the proportional gain of the DC-bus voltage controller. • Input not interconnected ➞ Adaptation = 0 %.
nConst_a	Proportional gain of the DC-bus voltage controller • Input with *nPos100_a* constant interconnected ➞ proportional gain = 100 %.
nDCSet_a	Voltage setpoint on which the DC-bus voltage is to be kept. • The voltage setpoint has to be set in the "free parameter" C00476/2.
nNset_a	Speed setpoint path (output: nNOut_a)
bFault	Activating the mains failure control ▸ Procedure of the mains failure control (📖 1641)
bReset	Reset mains failure control • The free *bFreeIn2* input of the application block used in the example can be e.g. connected to a digital input on I/O level.
nThreshold_a	Restart threshold • The restart threshold (at first approx 2 %) has to be set in the "free parameter" C00476/3.
nNAct_a	Comparison value for the restart threshold • Comparison value is the actual speed value
nSet_a	Speed starting point for the deceleration • Speed starting point is the actual speed value
nDCVoltAct_a	Current DC-bus voltage

19.1.106.3 Activating the mains failure control

The mains failure control is activated by setting *bFault* to TRUE. The inverter now generates the required operational energy from the rotational energy of the drive. The drive is braked via the power loss of the inverter and the motor. Thus, the speed deceleration ramp is shorter than for an uncontrolled system (coasting drive).

After activating the mains failure control:

1. The acceleration time set in C01470/2 is used to control the DC-bus voltage to the value at *nDCSet_a*.
2. At *nNOut_a*, an internally generated speed setpoint is output which serves to brake the drive to a speed close to "0" (via the speed setpoint).
   - Starting value for the controlled deceleration is the value at *nSet_a*. This input is appropriately connected to the actual speed value *nMotorSpeedAct_a* of the SB LS_MotorInterface.
   - The speed deceleration ramp (and hence the braking torque) results from the moment of inertia of the load machine(s), the power loss of the drive (system) and the set parameter setting.

> **Stop!**
>
> If a connected braking unit responds, the drive is braked with a max. possible torque (I_{max}). In this case, adapt the parameter setting if required (see the following chapter).
>
> If the power section is not supplied, the drive cannot create a standstill torque (important for active loads as e.g. hoists).

Stopping/cancelling the mains failure control

When *bFault* is reset to FALSE, an internal timing element is triggered. After the time set in C01470/3 has elapsed, the mains failure control is stopped/cancelled.

- When the restart protection is active, the drive is continued to be braked to standstill. ▶ Restart protection (📖 1649)
  - When the restart protection is active, the drive can only be reset by setting the *bReset* input to TRUE.
- When the restart protection not active, the drive is accelerated to the speed setpoint. ▶ Quick mains recovery (KU) (📖 1649)

19.1.106.4 Parameterising the mains failure control

The parameters to be set before initial commissioning strongly depend on the used motor, the moment of inertia of the drive and the drive configuration (single drive, drive system, master/slave operation etc.) and must hence be adapted to the corresponding application case.

Note!

The following data refers to the chapter "Procedure of the mains failure control". (🕮 1641)

1. Measure the DC-bus voltage with an oscilloscope (channel 1).
   - Signal for the message: *nDCVoltage_a* of the SB LS_MotorInterface.
2. Measure the speed with an oscilloscope (channel 2).
   - Signal for the message: *nMotorSpeedAct_a* of the SB LS_MotorInterface.
3. Set the operating threshold for the mains failure control in C00476/1.
   - Scaling: 100 % ≡ 1000 V
   - Recommended setting: approx. 50 V above the reset threshold for undervoltage in the DC bus (LU_{off}).
   - The operating thresholds for undervoltage (LU) and overvoltage (OU) depend on the setting in C00173:

Mains voltage		Undervoltage (LU)		Overvoltage (OU)	
1-phase	**C00173**	Operating threshold LU_{on}	Reset threshold LU_{off}	Operating threshold OU_{on}	Reset threshold OU_{off}
$230V_{AC}$	0, 1, 2, 3	180 V_{DC}	240 V_{DC}	400 V_{DC}	390 V_{DC}

Mains voltage		Undervoltage (LU)		Overvoltage (OU)	
3-phase	**C00173**	Operating threshold LU_{on}	Reset threshold LU_{off}	Operating threshold OU_{on}	Reset threshold OU_{off}
$400V_{AC}$	0	285 V_{DC}	430 V_{DC}	800 V_{DC}	790 V_{DC}
$440V_{AC}$	1	400 V_{DC}	430 V_{DC}		
$480V_{AC}$	2	490 V_{DC}	535 V_{DC}		
$500V_{AC}$	3	540 V_{DC}	585 V_{DC}		

4. Set the voltage setpoint in C00476/2 the DC-bus voltage is to be controlled to.
   - Scaling: 100 % ≡ 1000 V
   - Recommended setting: approx. 700 V → C00476/2 = 70 %

Stop!

The voltage setpoint must be below the operating threshold of a braking unit if connected. If a connected braking unit responds, the drive with max. possible torque (I_{max}) will be braked. The desired operational performance gets lost.

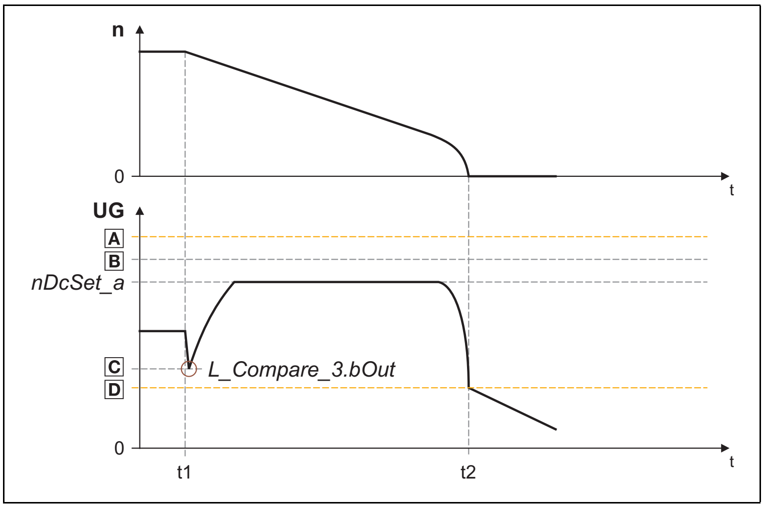
19.1.106.5 Commissioning the mains failure control

Commissioning should be executed with motors without load.

1. Start the drive.

2. Set the acceleration time:
   - Set the speed setpoint to 100 %, operate the inverter with max. speed.
   - Set controller inhibit and measure deceleration time up to standstill.
   - Set approx. 1/10 of the measured deceleration time in C01470/2.
3. Set retrigger time:
   - In case of mains failure control via detection of the DC-bus voltage level:
     Set the deceleration time measured under point 2 in C01470/3.
   - In case of mains failure control via an external system:
     Set the time in which the drive is to be continued to be brought to a standstill under control at short-term mains recovery in C01470/3.
4. Switch off the supply voltage (mains or DC bus).

The following profile is to be shown on the oscilloscope:



n = speed of the drive

A Operating threshold OU
B Switch-on threshold - braking unit
C Operating threshold - mains failure
D Operating threshold LU

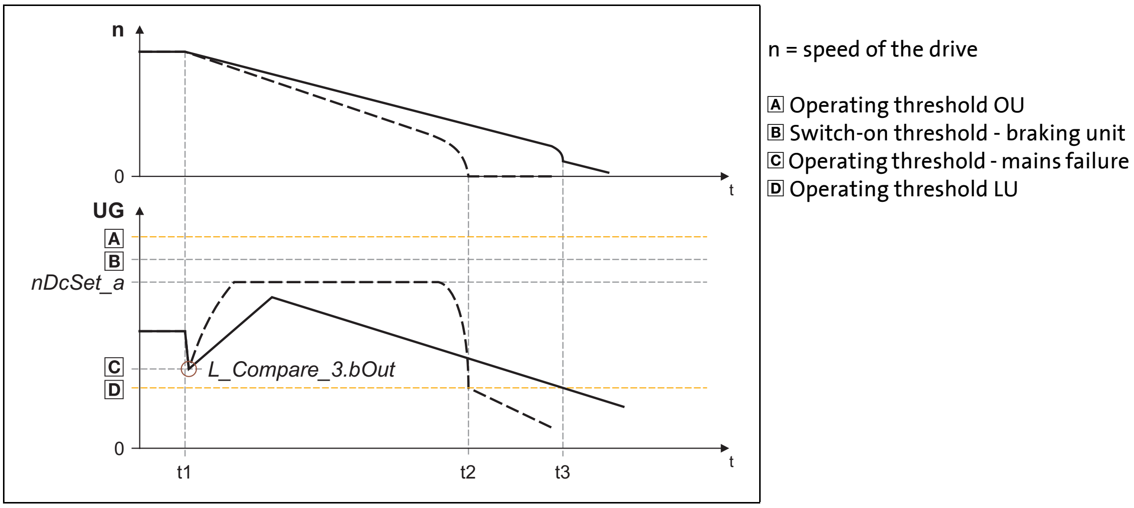
[19-50] Schematic diagram with activated mains failure control (ideal profile)

Point in time	Info
t = t1	Mains failure
t = t2	Speed "0" reached.

19.1.106.6 Fine adjustment of the mains failure control

For the fine adjustment, repeat the following points several times:

1. A final speed as low as possible has to be reached before the inverter reaches the operating threshold for undervoltage (LU):
   - Increase proportional gain Vp in C01469/1 .
   - Reduce Vdc-bus controller reset time in C01470/1.
2. The responding of the braking unit or the reaching of the operating threshold for overvoltage (OU) has to be prevented:
   - Increase Vdc controller reset time in C01470/1 until the ideal profile shown in [19-50] will be almost reached.
   - If required, reduce the *nDCSet_a* voltage setpoint to which the DC-bus voltage is to be controlled (in the interconnection example via the free parameter C00476/2).
3. Increasing the deceleration time or reducing the braking torque is only possible to a limited extent:
   - Increasing the acceleration time in C01470/2 reduces the initial braking torque and simultaneously increases the deceleration time.
   - Increasing the Vdc controller reset time in C01470/1 reduces the braking torque and simultaneously increases the deceleration time. If the reset times are too high, the inverter reaches the operating threshold for undervoltage (LU) before standstill is reached. Hence, the drive is not controlled anymore.



[19-51] Schematic diagram with different braking torques

Point in time	Info
t = t1	Mains failure
t = t2	Speed "0" with higher braking torque reached (short reset time).
t = t3	With a lower braking torque (higher reset time), the drive reaches the operating threshold for undervoltage (LU) without reaching speed "0".
t > t3	Drive is not controlled anymore (is braked by friction).

19.1.106.7 Reset mains failure control

The mains failure control is rest by setting *bReset* to TRUE.

- A reset via *bReset* is always required when the restart protection is active or the restart protection is used and the supply (mains or DC supply) has been switched on.
- The *bReset* input can, for instance, be connected to a digital input.

19.1.106.8 Restart protection

The integrated restart protection is to prevent a restart in the lower speed range when the supply voltage was only interrupted briefly (mains recovery before drive stands still).

- The *nThreshold_a* input serves to specify the restart threshold in [%] with regard to the reference speed (C00011) below which no start has to take place after mains recovery.
  - If in case of mains recovery the speed is below the restart threshold, the drive is continued to be braked in a controlled way. This function is only stopped by setting *bReset* to TRUE.
  - If the speed is above the restart threshold after mains recovery, the drive changes to its setpoint *nNSet_a*.
  - The function is switched inactive when *nThreshold_a* = 0 %.
- A reset takes place by setting *bReset* to TRUE and is required after every mains switching.
  - In this case, the *bActive* output is set to TRUE when the *bFault* input is set to FALSE.

19.1.106.9 Quick mains recovery (KU)

The quick mains recovery causes a restart of the inverter if the restart protection is not active anymore.

- In case, the drive reaches its setpoint.
- If this behaviour is not desired, you can delay the restart by setting a retrigger time in C01470/3 or prevent it by using the restart protection.
- Dependent on the system, the mains recovery is reported through the mains failure control via the level of the DC-bus voltage. ▶ Procedure of the mains failure control (📖 1641)

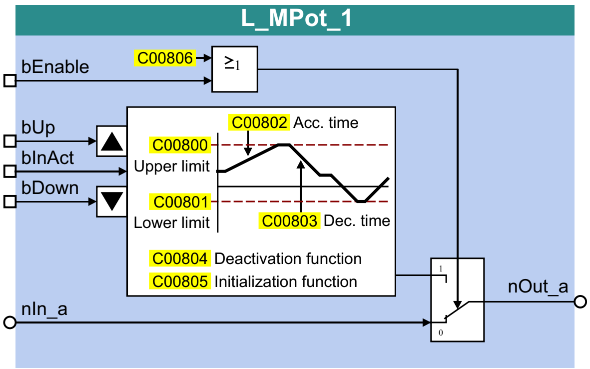
A quick mains recovery is caused by a "quick interruption" of the energy supply company (e.g. thunderstorm) and by faulty components in the supply cables (e.g. collector rings).

- Then, set the retrigger time in C01470/3 higher than the measured deceleration time reached in braking operation.

19.1.107 L_MPot_1

This FB replaces a hardware motor potentiometer and can be used as an alternative setpoint source controlled via two inputs.

- The signal is output via a ramp function generator with linear ramps.
- The acceleration and deceleration times are set via parameters.
- Constant ramping even with speed limit values changed online.
- The motor potentiometer function can be switched on/off online.



inputs

Designator Data type	Information/possible settings	
bEnable BOOL	Switch over motor potentiometer function *bEnable* input and C00806 code are ORed.	
	TRUE	Motor potentiometer function is active, setpoint can be changed via *bUp* and *bDown*. • With switching to TRUE, the value applied to *nIn_a* is automatically transferred to the motor potentiometer.
	FALSE	The value applied to *nIn_a* is output at *nOut_a*.
nIn_a INT	When bEnable = FALSE, the analog nIn_ is input signal switched to the nOut_a output.	
bUp BOOL	Approaching of the upper speed limit value set in C00800.	
	TRUE	The *nOut_a* output signal runs to its upper limit value (*nHighLimit*). • If the *bDown* input is simultaneously set to TRUE, the *nOut_a* output signal is not changed.
bDown BOOL	Approaching of the lower speed limit value set in C00801.	
	TRUE	The *nOut_a* output signal runs to its lower limit value (*nLowLimit*). • If the *bUp* input is simultaneously set to TRUE, the *nOut_a* output signal is not changed.
bInAct BOOL	Deactivate motor potentiometer function • This input has the highest priority. • When the motor potentiometer is deactivated, the *nOut_a* output signal follows the function set with code C00804.	
	TRUE	Motor potentiometer function is deactivated.

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

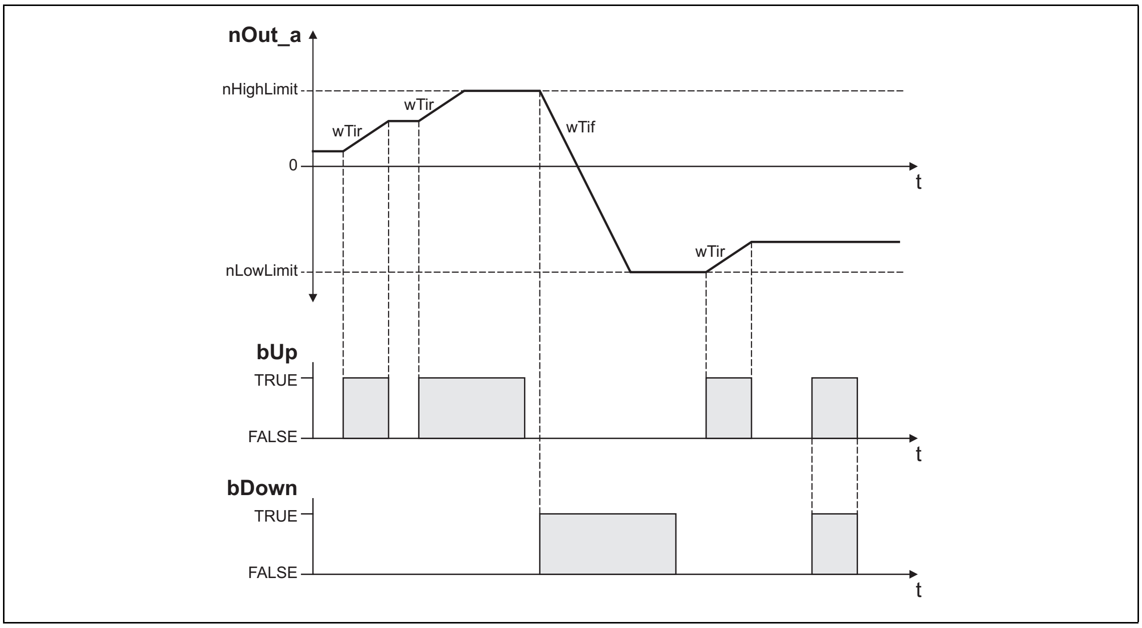
Parameters

Parameters	Possible settings			Information
C00800	-199.99	%	199.99	Upper limit • Lenze setting: 100.00 %
C00801	-199.99	%	199.99	Lower limit • Lenze setting: -100.00 %
C00802	0.1	s	6000.0	Acceleration time • Lenze setting: 10.0 s
C00803	0.1	s	6000.0	Deceleration time • Lenze setting: 10.0 s
C00804				Inactive function • Selection of response when deactivating the motor potentiometer via the input *bInAct*. • Lenze setting: 0
	0	No further action; *nOut_a* retains its value.		
	1	The motor potentiometer returns to 0 % within the deceleration time T_{if}		
	2	The motor potentiometer runs to the lower limit value (C00801) within the deceleration time T_{if}		
	3	The motor potentiometer output immediately changes to 0 %		**Important for the emergency stop function**
	4	The motor potentiometer output immediately changes to the lower limit value (C00801)		
	5	The motor potentiometer runs to the upper limit value (C00800) within the acceleration time T_{ir}		
C00805				Init function • Selection of response when switching on the device. • Lenze setting: 0
	0	The output value being output during mains power-off is saved non-volatilely in the internal memory of the inverter. It will be reloaded during mains power-on.		
	1	The lower limit value (C00801) is loaded during mains power-on.		
	2	An output value = 0 % is loaded during mains power-on.		
C00806				Use of the motor potentiometer • When switching to 1: YES, the value applied to *nIn_a* is automatically transferred to the motor potentiometer. • Lenze setting: 0
	0	No		
	1	Yes		

19.1.107.1 Activate & control motor potentiometer

When *bInAct* is set to FALSE, the motor potentiometer is activated.

- The currently active function depends on the current output signal *nOut_a*, the limit values set and the control signals at *bUp* and *bDown*.
- When the *nOut_a* output signal is outside the limits set, the output signal runs to the next limit with the Ti times set. This process is independent of the control signals at *bUp* and *bDown*.
- When the *nOut_a* output signal is inside the limits set, the output signal changes according to the control signals at *bUp* and *bDown*.

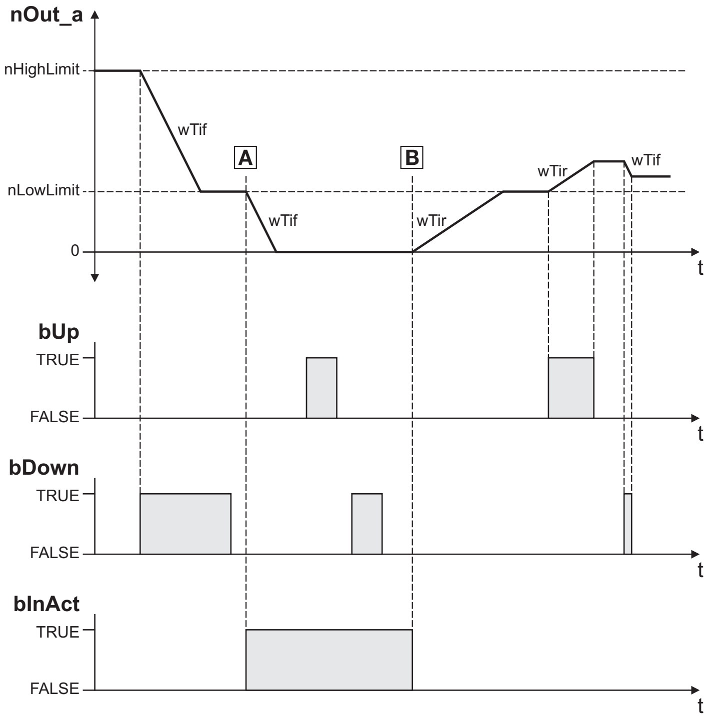


[19-52] Example: Control of the motor potentiometer

bUp	bDown	bInact	Function
FALSE	FALSE	FALSE	The *nOut_a* output signal remains unchanged.
TRUE	FALSE		The *nOut_a* output signal runs to its upper limit value (*nHighLimit*).
FALSE	TRUE		The *nOut_a* output signal runs to its lower limit value (*nLowLimit*).
TRUE	TRUE		The *nOut_a* output signal remains unchanged.
-	-	TRUE	The motor potentiometer function is deactivated. The *nOut_a* output signal responds according to the function selected via *Function*.

19.1.107.2 Deactivate motor potentiometer

When the motor potentiometer is deactivated by setting *bInAct* to TRUE, the *nOut_a* output signal responds according to the function selected via *Function*.



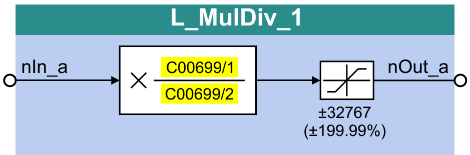
A When the motor potentiometer is deactivated, *nOut_a* is reduced to 0 % with the deceleration time *wTif*.
B If it is activated again, *nOut_a* is increased again to *nLowLimit* with the acceleration time *wTir*.

[19-53] Example: Deactivation of the motor potentiometer when the *Function* = 1 has been selected

19.1.108 L_MulDiv_1

This FB multiplies the analog input signal with a parameterisable factor.

- The value of the factor is determined by a quotient consisting of numerator and denominator .
- The value output at *nOut_a* is limited to ±199.99 %.
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
nIn1 INT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Product value (result of the multiplication) • Internal limitation to ± 32767

Parameters

Parameters	Possible settings			Information
C00699/1	-32767		32767	Meters
C00699/2	-32767		32767	Denominator

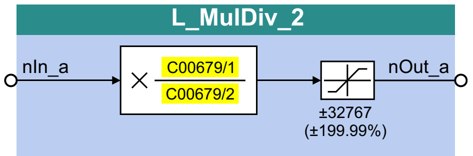
Function

$$nOut\_a = nIn\_a \times \frac{C00699/1}{C00699/2}$$

19.1.109 L_MulDiv_2

This FB multiplies the analog input signal with a parameterisable factor.

- The value of the factor is determined by a quotient consisting of numerator and denominator .
- The value output at *nOut_a* is limited to ±199.99 %.
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
nIn1 INT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Product value (result of the multiplication) • Internal limitation to ± 32767

Parameters

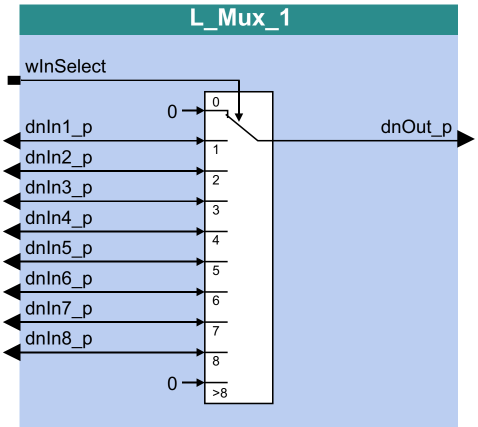
Parameters	Possible settings			Information
C00679/1	-32767		32767	Meters
C00679/2	-32767		32767	Denominator

Function

$$nOut\_a = nIn\_a \times \frac{C00679/1}{C00679/2}$$

19.1.110 L_Mux_1

This FB provides one of the eight input signals *dnIn1_p* ... *dnIn8_p* at the output *dnOut_p*. The selection is made by means of the signal at the input *wInSelect*.



inputs

Designator Data type	Information/possible settings	
wInSelect WORD	Input signal	
	1 ... 8	• The values "1" ... "8" select the input signal to be applied to the output. • Values from"1" ... "8" set the *dnOut_p* output to "0".
dnIn1_p DINT ... dnIn8_p DINT	Input signal	

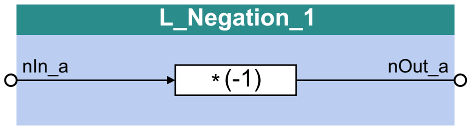
outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal

19.1.111 L_Negation_1

This FB converts the sign of the input signal, i.e. the input signal is multiplied by the value -1 and is then output.

- With the value - 32768 at the *nIn_a* input, the value + 32767 is provided at the *nOut_a* output.



inputs

Designator Data type	Value/meaning
nIn_a INT	Input signal

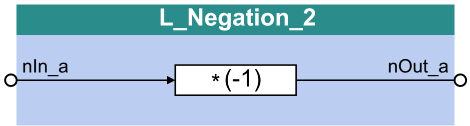
outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.112 L_Negation_2

This FB converts the sign of the input signal, i.e. the input signal is multiplied by the value -1 and is then output.

- With the value - 32768 at the *nIn_a* input, the value + 32767 is provided at the *nOut_a* output.



inputs

Designator Data type	Value/meaning
nIn_a INT	Input signal

outputs

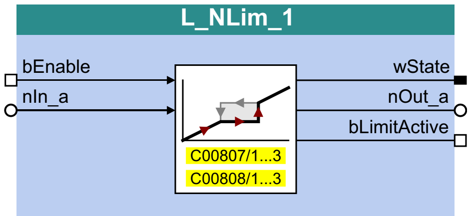
Designator Data type	Value/meaning
nOut_a INT	Output signal

19.1.113 L_NLim_1

This FB can hide up to three parameterisable blocking zones within one continuous signal characteristic.

Note!

A value of "0" cannot be masked out if the *nIn_a* input signal changes signs.



inputs

Designator Data type	Information/possible settings	
bEnable BOOL	Activate zone masking	
	FALSE	The FB has been deactivated. • The input signal is output one-to-one at the *nOut_a* output.
	TRUE	The FB has been activated. • Zone masking of the input signal is carried out according to the parameterised blocking zones.
nIn_a INT	Input signal	

outputs

Identifier/data type	Value/meaning	
wState WORD	Bit-coded status word • Bits that are not listed are reserved for future extensions.	
	Bit 0	No blocking zone active
	Bit 1	Blocking zone 1 active
	Bit 2	Blocking zone 2 active
	Bit 3	Blocking zone 3 active
nOut_a INT	Output signal • If the FB has been activated, the output signal is outside the blocking zones.	
bLimitActive BOOL	"Limitation active" status signal	
	TRUE	The input signal is inside a blocking zones and is limited to the respective boundary value of the blocking zone.

Parameters

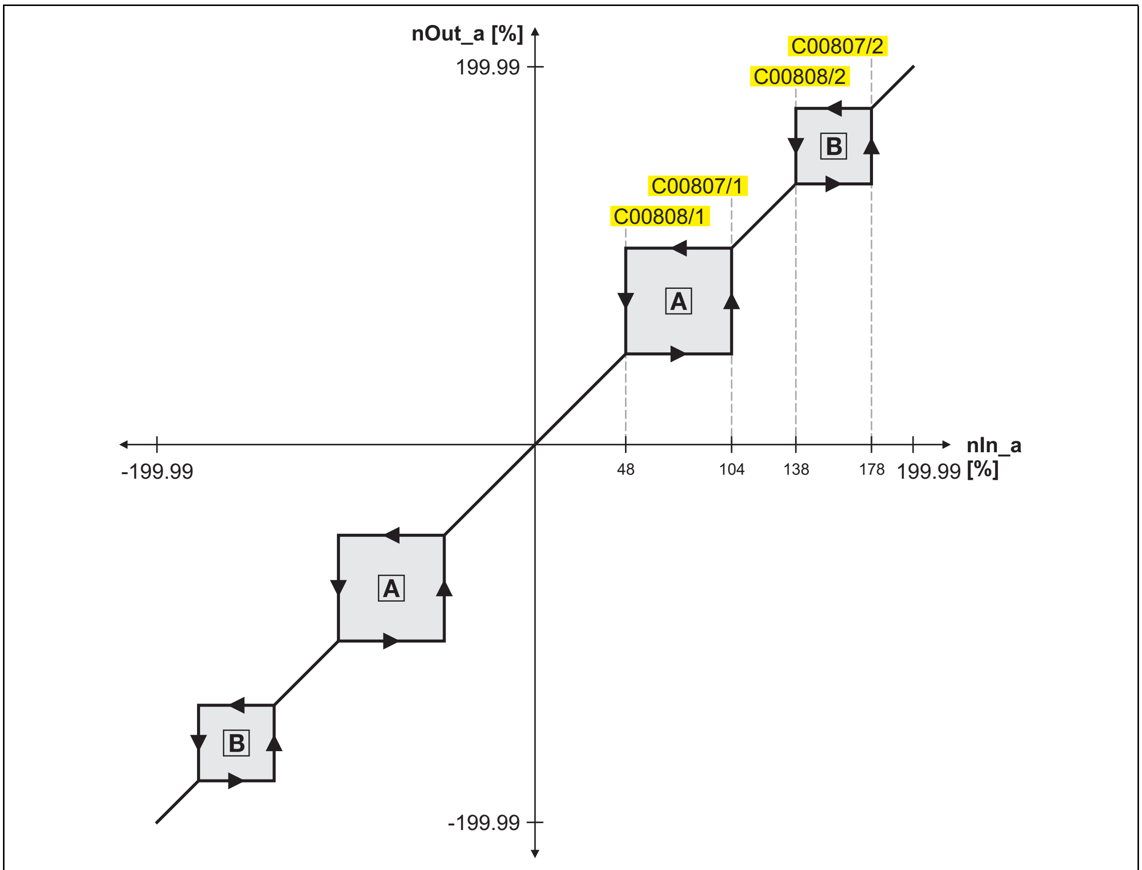
Parameters	Possible settings			Information
C00807/1...3	0.00	%	199.99	Max. skip frequency 1 ... 3 • Lenze setting: 0.00 %
C00808/1...3	0.00	%	199.99	Min. skip frequency 1 ... 3 • Lenze setting: 0.00 %

Definition of the blocking zones

Up to three zones can be parameterised which are to be skipped by the *nOut_a* output signal.

The example below shows the parameter setting of two blocking zones for the **L_NLim_1** FB:

Parameters	Blocking zone 1		Blocking zone 2		Blocking zone 3	
Minimum limit value	C00808/1:	48 %	C00808/2:	138 %	C00808/3:	0 %
Maximum limit value	C00807/1:	104 %	C00807/2:	178 %	C00807/3:	0 %



Ⓐ Blocking zone 1
Ⓑ Blocking zone 2

[19-54] Zone masking by means of parameterisable blocking zones (in our example: L_NLim_1)

- The parameterised blocking zones have the same effect on negative input signals.
- A blocking zone is deactivated by entering identical limit values (in our example: blocking zone 3).

Overlapping of blocking zones

If blocking zones overlap, the lowest and highest value of the overlapping zones form a new zone.

In this case, the status display (*wState* output) will also just display one zone (the lower one of the two original zones).

Abutting blocking zones

If two blocking zones abut (e.g. 20 ... 30 % and 30 ... 40 %), the limit value between the two zones (in this example 30 %) is also passed through.

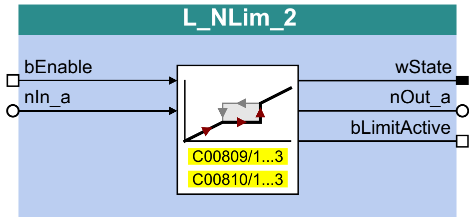
The same applies to a limit range of 0 ... xx %. At the zero crossing of the *nIn_a* input signal, a value of "0" is also output at the *nOut_a* output. Value "0" cannot be excluded. If the *nIn_a* input signal drops back to "0", the *nOut_a* output will retain the upper limit value.

19.1.114 L_NLim_2

This FB can hide up to three parameterisable blocking zones within one continuous signal characteristic.

Note!

A value of "0" cannot be masked out if the *nIn_a* input signal changes signs.



inputs

Designator Data type		Information/possible settings
bEnable BOOL	Activate zone masking	
	FALSE	The FB has been deactivated. • The input signal is output one-to-one at the *nOut_a* output.
	TRUE	The FB has been activated. • Zone masking of the input signal is carried out according to the parameterised blocking zones.
nIn_a INT	Input signal	

outputs

Identifier/data type		Value/meaning
wState WORD	Bit-coded status word • Bits that are not listed are reserved for future extensions.	
	Bit 0	No blocking zone active
	Bit 1	Blocking zone 1 active
	Bit 2	Blocking zone 2 active
	Bit 3	Blocking zone 3 active
nOut_a INT	Output signal • If the FB has been activated, the output signal is outside the blocking zones.	
bLimitActive BOOL	"Limitation active" status signal	
	TRUE	The input signal is inside a blocking zones and is limited to the respective boundary value of the blocking zone.

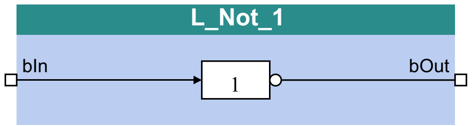
Parameters

Parameters	Possible settings			Information
C00809/1...3	0.00	%	199.99	Max. skip frequency 1 ... 3 • Lenze setting: 0.00 %
C00810/1...3	0.00	%	199.99	Min. skip frequency 1 ... 3 • Lenze setting: 0.00 %

For a detailed functional description see L_NLim_1.

19.1.115 L_Not_1

This FB negates a BOOL data type signal.



inputs

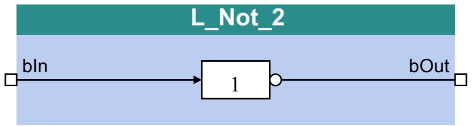
Designator Data type	Value/meaning
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Result of the NOT operation (negated input signal)

19.1.116 L_Not_2

This FB negates a BOOL data type signal.



inputs

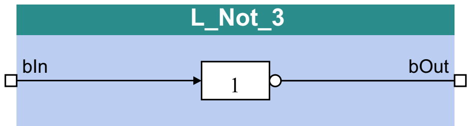
Designator Data type	Value/meaning
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Result of the NOT operation (negated input signal)

19.1.117 L_Not_3

This FB negates a BOOL data type signal.



inputs

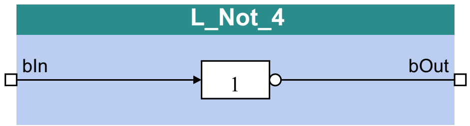
Designator Data type	Value/meaning
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Result of the NOT operation (negated input signal)

19.1.118 L_Not_4

This FB negates a BOOL data type signal.



inputs

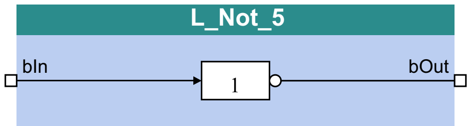
Designator Data type	Value/meaning
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Result of the NOT operation (negated input signal)

19.1.119 L_Not_5

This FB negates a BOOL data type signal.



inputs

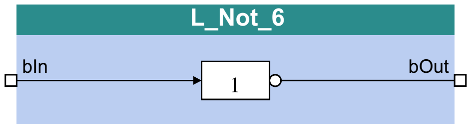
Designator Data type	Value/meaning
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Result of the NOT operation (negated input signal)

19.1.120 L_Not_6

This FB negates a BOOL data type signal.



inputs

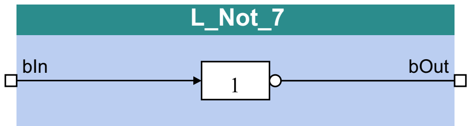
Designator Data type	Value/meaning
bIn BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Result of the NOT operation (negated input signal)

19.1.121 L_Not_7

This FB negates a BOOL data type signal.



inputs

Designator Data type	Value/meaning
bIn BOOL	Input signal

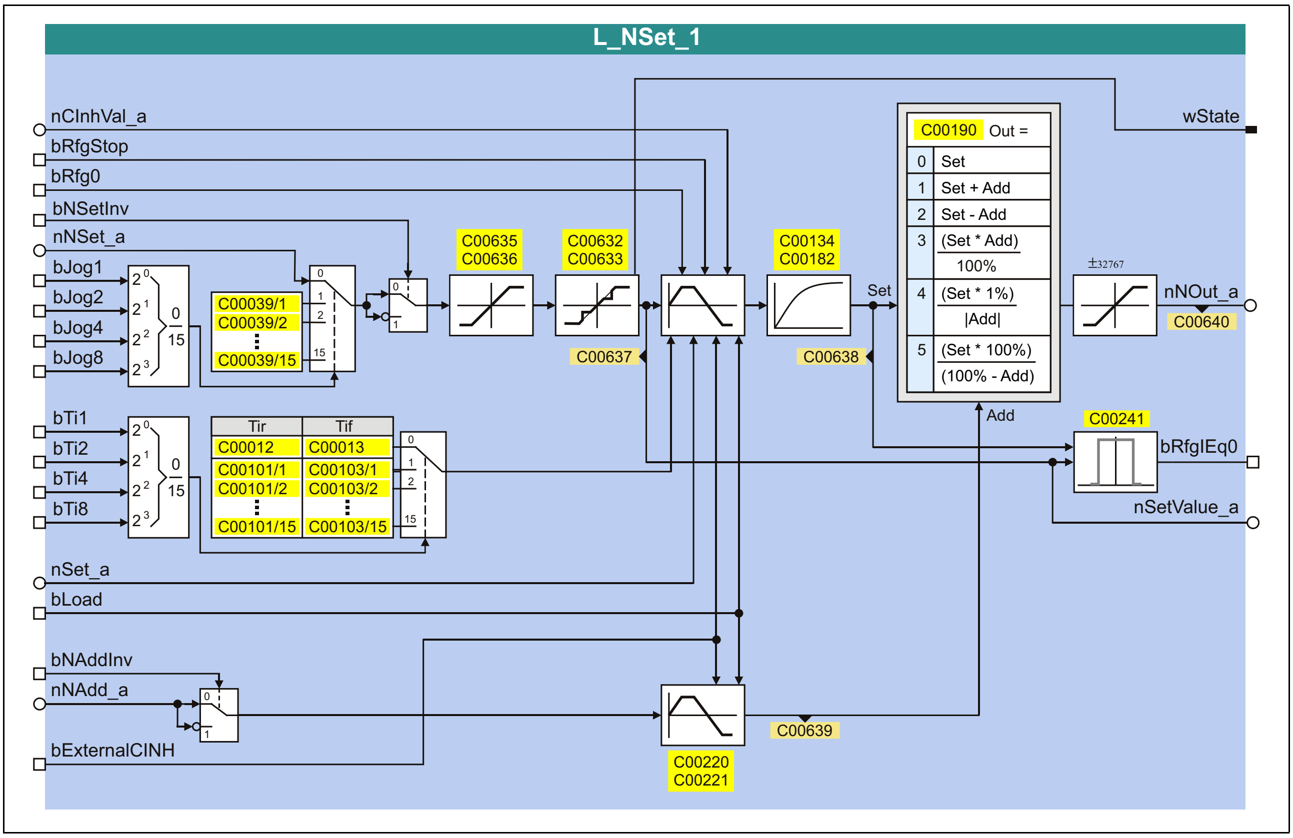
outputs

Designator Data type	Value/meaning
bOut BOOL	Result of the NOT operation (negated input signal)

19.1.122 L_NSet_1

This FB is used for general signal processing of process values and is provided with the following functions:

- Ramp function generator
  - With linear ramps for main and additional setpoint path
  - With S-shaped ramp (PT1 rounding)
  - Setting and holding
- Internal limitation of the input signal
- 3 adjustable blocking zones
- Arithmetic function
- 15 fixed setpoints (JOG setpoints)
- 15 acceleration and deceleration times



inputs

Designator Data type	Information/possible settings	
nCInhVal_a INT	Main setpoint signal which is to be accepted by the main setpoint integrator when the controller is inhibited.	
bRfgStop BOOL	Holding (freezing) of the current value of the main setpoint integrator	
	TRUE	The current value of the main setpoint integrator is held.
bRfg0 BOOL	Leading the main setpoint integrator to 0 within the current Ti times	
	TRUE	The current value of the main setpoint integrator is led to "0" within the Ti time set.
bNSetInv BOOL	Signal inversion for the main setpoint	
	TRUE	Main setpoint signal is inverted.
nNset_a INT	Main setpoint signal • Other signals are also permitted	
bJog1 ... bJog8 BOOL	Inputs for overriding fixed setpoints (JOG setpoints) for the main setpoint • Selection inputs are binary coded.	
bTI1 ... bTI8 BOOL	Selection inputs for alternative acceleration/deceleration times for the main setpoint • Selection inputs are binary coded.	
nSet_a INT	Starting value which is loaded into the main setpoint integrator by setting *bLoad* to TRUE.	
bLoad BOOL	Control of both ramp function generators in special situations, e.g. QSP	
	TRUE	The *nSet_a* input signal is loaded into the main setpoint integrator and the additional setpoint integrator is set to "0".
bAddInv BOOL	Signal inversion for the additional setpoint	
	TRUE	Additional setpoint signal is inverted.
nNAdd_a INT	Additional setpoint signal • Other signals are also permitted	
bExternalCINH BOOL	Additional load input for the main setpoint integrator and the additional setpoint integrator	
	TRUE	The main setpoint integrator is set to the value applied at *nCInhVal_a*. The additional setpoint integrator is set to "0". ▸ Application example for the additional load function (📖 1678)

outputs

Designator Data type	Value/meaning
nNOut_a INT	Speed setpoint output signal • Scaling: 16384 ≡ 100 %
bRfgIEqO BOOL	Status signal "setpoint before ramp = setpoint after ramp (ramp inactive)"

Designator Data type	Value/meaning	
wState WORD	Bit-coded status word • Bits that are not listed are reserved for future extensions.	
	Bit 0	No blocking zone active
	Bit 1	Blocking zone 1 active
	Bit 2	Blocking zone 2 active
	Bit 3	Blocking zone 3 active
	Bit 4	Jog in blocking zone
	Bit 5	MaxLimit active
	Bit 6	MinLimit active
nSetValue_a INT	Speed-setpoint input signal of the ramp function generator • Scaling: 16384 ≡ 100 %	

Parameters

Parameters	Possible settings			Information
C00012	0.000	s	999.900	Acceleration time T_{ir} for the main setpoint • Lenze setting: 0.000 s
C00013	0.000	s	999.900	Deceleration time T_{if} for the main setpoint • Lenze setting: 0.000 s
C00039/1..15	-199.99	%	199.99	Fixed setpoints (JOG setpoints) • Lenze setting: 0.00 %
C00101/1..15	0.000	s	999.900	Alternative acceleration times (T_{ir}) for the main setpoint • Lenze setting: 0.000 s
C00103/1..15	0.000	s	999.900	Alternative deceleration times (T_{if}) for the main setpoint • Lenze setting: 0.000 s
C00134				Activates ramp rounding with PT1 behaviour for the main setpoint • The corresponding S-ramp time must be set in C00182. • Lenze setting: 0 (deactivated)
	0	Off		
	1	PT1 behaviour		
C00182	0.01	s	50.00	S-ramp time PT1 • Lenze setting: 20.00 s
C00190				Selection of the arithmetic function for combining main and additional setpoint
	0	NOut = NSet		Lenze setting • The additional setpoint is not processed.
	1	NOut = NSet + NAdd		
	2	NOut = NSet - NAdd		
	3	NOut = (NSet * NAdd) / 100%		
	4	NOut = (NSet * 1%) / |NAdd|		
	5	NOut = (NSet * 100%) / (100% - NAdd)		
C00220	0.000	s	999.900	Acceleration time T_{ir} for the additional setpoint • Lenze setting: 0.000 s

Parameters	Possible settings			Information
C00221	0.000	s	999.900	Deceleration time T_{if} for the additional setpoint • Lenze setting: 0.000 s
C00241	0.00	%	100.00	Hysteresis window for zero detection of speed output setpoint (output *bRfgIEqO*) • Lenze setting: 0.50 %
C00632/1...3	0.00	%	199.99	Maximum limit values for the speed blocking zones • Selection of the maximum limit values for the blocking zones in which the speed must not be constant. • Lenze setting: 0.00 %
C00633/1...3	0.00	%	199.99	Minimum limit values for the speed blocking zones • Selection of the minimum limit values for the blocking zones in which the speed must not be constant. • Lenze setting: 0.00 %
C00634				Status (bit-coded) • Bits that are not listed are reserved for future extensions.
	Bit 0	No blocking zone active		
	Bit 1	Blocking zone 1 active		
	Bit 2	Blocking zone 2 active		
	Bit 3	Blocking zone 3 active		
	Bit 4	Jog in blocking zone		
	Bit 5	MaxLimit active		
	Bit 6	MinLimit active		
C00635	-199.99	%	199.99	nMaxLimit • Maximum speed setpoint for speed setpoint limitation • Lenze setting: 199.99 %
C00636	-199.99	%	199.99	nMinLimit • Minimum speed setpoint for speed setpoint limitation • Lenze setting: -199.99 %
C00637	-199.99	%	199.99	Blocking zone output • Display of the speed setpoint after being processed by blocking zone function.
C00638	-199.99	%	199.99	Ramp smoothing output • Display of the speed setpoint after being processed by PT1 filter function.
C00639	-199.99	%	199.99	Additional value output • Display of the additional speed setpoint after being processed by ramp generator.
C00640	-199.99	%	199.99	Output nNOut_a • Display of the generated main speed setpoint at the *nNOut_a* output.

19.1.122.1 Main setpoint path

- The signals in the main setpoint path are limited to a value range of ±32767.
- The signal at *nNSet_a* is first led via the JOG selection function.
- A selected JOG value switches the *nNSet_a* input inactive. Then, the subsequent signal conditioning operates with the JOG value.

19.1.122.2 JOG setpoints

In addition to the direct main setpoint selection via the *nNSet_a* input, so-called JOG setpoints can be preset in C00039/1...15.

- The JOG setpoints are binary-coded and can be called using the *bJog1* ... *bJog8* selection inputs so that 15 options are available:

Selection inputs				Main setpoint Main setpoint
bJog8	bJog4	bJog2	bJog1	
FALSE	FALSE	FALSE	FALSE	*nNset_a*
FALSE	FALSE	FALSE	TRUE	C00039/1
FALSE	FALSE	TRUE	FALSE	C00039/2
FALSE	FALSE	TRUE	TRUE	C00039/3
FALSE	TRUE	FALSE	FALSE	C00039/4
FALSE	TRUE	FALSE	TRUE	C00039/5
FALSE	TRUE	TRUE	FALSE	C00039/6
FALSE	TRUE	TRUE	TRUE	C00039/7
TRUE	FALSE	FALSE	FALSE	C00039/8
TRUE	FALSE	FALSE	TRUE	C00039/9
TRUE	FALSE	TRUE	FALSE	C00039/10
TRUE	FALSE	TRUE	TRUE	C00039/11
TRUE	TRUE	FALSE	FALSE	C00039/12
TRUE	TRUE	FALSE	TRUE	C00039/13
TRUE	TRUE	TRUE	FALSE	C00039/14
TRUE	TRUE	TRUE	TRUE	C00039/15

- The number of selection inputs to be assigned depends on the number of JOG setpoints required:

Number of JOG setpoints required	Number of selection inputs to be assigned (bJog1 ... bJog8)
1	At least 1
2 ... 3	at least 2
4 ... 7	at least 3
8 ... 15	4

19.1.122.3 Setpoint inversion

The output signal of the JOG function is led via an inverter.

The sign of the setpoint changes if *bNSetInv* is set to TRUE.

19.1.122.4 Value range of the input signal

The value range of the input signal can be limited by using the following parameters:

- C00635: MaxLimit (default setting: +199.99 %)
- C00636: MinLimit (default setting: -199.99 %)

19.1.122.5 Skip frequency function

If the speed setpoints in speed-variable drives are linearly increasing, for instance, the frequency/speed range is divided into a number of equal time segments. Therefore, there may be speeds during acceleration time which must be bridged very fast (e.g. natural resonant frequencies).

The skip frequency function offers the opportunity to select a range in which the initial speed is maintained. If the speed setpoint leaves that range, the drive will be accelerated to reach the desired speed.

Note!

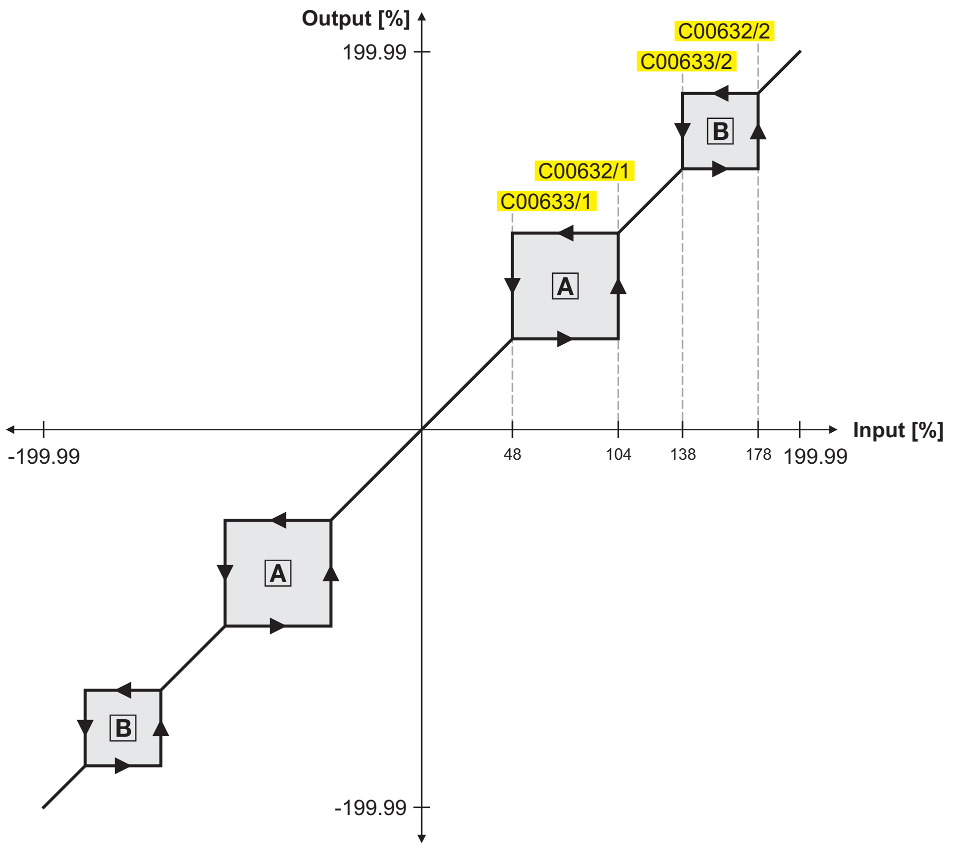
- Blocking frequencies act on the main setpoint only.
- It is not possible to exclude "0" speed if there is a sign reversal of the speed setpoint.

Definition of the blocking zones

The subcodes of codes C00632 and C00633 can be used to define three zones which are to be skipped by the output setpoint and which are to be passed as fast as possible by the ramp function generator.

The example below shows the parameter setting of two blocking zones:

Parameters	Blocking zone 1		Blocking zone 2		Blocking zone 3	
Minimum limit value	C00633/1:	48 %	C00633/2:	138 %	C00633/3:	0 %
Maximum limit value	C00632/1:	104 %	C00632/2:	178 %	C00632/3:	0 %



A Blocking zone 1
B Blocking zone 2

[19-55] Zone masking by means of parameterisable blocking zones

- The parameterised blocking zones have the same effect on negative input signals.
- A blocking zone is deactivated by entering identical limit values (in our example: blocking zone 3).

Overlapping of blocking zones

If blocking zones overlap, the lowest and highest value of the overlapping zones form a new zone.

In this case, the status display (output *wState* or display parameter C00634) only indicates one zone (the lower of the two original zones).

Abutting blocking zones

If two blocking zones abut (e.g. 20 ... 30 % and 30 ... 40 %), the limit value between the two zones (in this example 30 %) is also passed through.

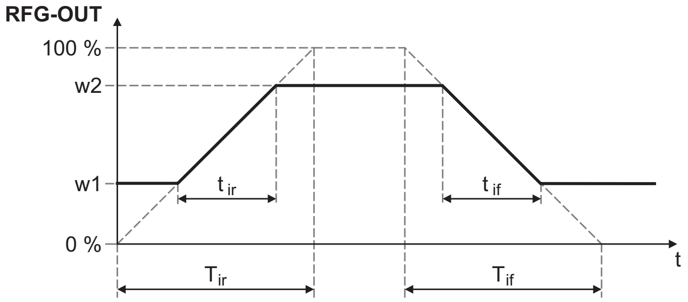
The same applies to a limit range of 0 ... xx %. During zero crossing of the speed setpoint, "0" speed is output as setpoint. It is possible to exclude "0" speed. However, in this case, the output speed will remain on the upper limit value when the input setpoint becomes "0".

Tip!

As described above, the acceleration phase starts after the blocking zones have been passed through. The ramp function generator integrated in the **L_Nset** function block limits the progression of the speed. For this reason, the time values set for the integrated ramp function generator should be as low as possible whereas the setpoint for the **L_NSet** function block should be generated by a ramp function generator with higher time values (e.g. L_MPot function block).

19.1.122.6 Ramp function generator for the main setpoint

The setpoint is now led via a ramp function generator with linear characteristic. The ramp function generator converts setpoint step-changes at the input into a ramp.



w1, w2 = change of the main setpoint as a function of t_{ir} and t_{if}
RFG-OUT = output of the ramp function generator

[19-56] Acceleration and deceleration times

- t_{ir} and t_{if} are the desired times for changing between w1 and w2.
- S-ramps are possible by selecting S-ramp times.
- The t_{ir}/t_{if} values are converted into the required Ti times according to the following formula:

$T_{ir} = t_{ir} \cdot \frac{100\,\%}{w2 - w1}$	$T_{if} = t_{if} \cdot \frac{100\,\%}{w2 - w1}$

Setting and selection of Ti times

Via parameters, you can select 16 different Tir and Tif times each for the ramp function generator.

- The selection is made via the binary coded selection inputs *bTI1* ... *bTI8*:

Selection inputs				Used Acceleration time	Used Deceleration time
bTI8	bTI4	bTI2	bTI1		
FALSE	FALSE	FALSE	FALSE	C00012	C00013
FALSE	FALSE	FALSE	TRUE	C00101/1	C00103/1
FALSE	FALSE	TRUE	FALSE	C00101/2	C00103/2
FALSE	FALSE	TRUE	TRUE	C00101/3	C00103/3
FALSE	TRUE	FALSE	FALSE	C00101/4	C00103/4
FALSE	TRUE	FALSE	TRUE	C00101/5	C00103/5
FALSE	TRUE	TRUE	FALSE	C00101/6	C00103/6
FALSE	TRUE	TRUE	TRUE	C00101/7	C00103/7
TRUE	FALSE	FALSE	FALSE	C00101/8	C00103/8
TRUE	FALSE	FALSE	TRUE	C00101/9	C00103/9
TRUE	FALSE	TRUE	FALSE	C00101/10	C00103/10
TRUE	FALSE	TRUE	TRUE	C00101/11	C00103/11
TRUE	TRUE	FALSE	FALSE	C00101/12	C00103/12
TRUE	TRUE	FALSE	TRUE	C00101/13	C00103/13
TRUE	TRUE	TRUE	FALSE	C00101/14	C00103/14
TRUE	TRUE	TRUE	TRUE	C00101/15	C00103/15

Function

- When the controller is inhibited (CINH), the ramp function generator accepts the value applied at *nCInhVal_a* and transfers it to the downstream function. This function has priority over all other functions.
- *bRfgStop* = TRUE
  - The ramp function generator is stopped. Changes at the input of the ramp function generator have no effect on the output signal.
- *bRfg0* = TRUE
  - The ramp function generator runs to 0 along its deceleration ramp.
- Furthermore it is possible to load the ramp function generator online with a defined value. For this purpose, *bLoad* must be set to TRUE. As long as this input is set, the value at *nSet_a* is transferred to the ramp function generator and provided at the output.

Priorities:

CINH	bLoad	bRfg0	bRfgStop	Function
FALSE	FALSE	FALSE	FALSE	The ramp function generator follows the input value via the set ramps.
FALSE	FALSE	FALSE	**TRUE**	**Stop the ramp function generator:** The value at the output of the ramp function generator is held.
FALSE	FALSE	**TRUE**	FALSE	**Ramp down the ramp function generator:** The ramp function generator runs to 0 within the set deceleration time.
FALSE	FALSE	**TRUE**	TRUE	
FALSE	**TRUE**	FALSE	FALSE	**Load ramp function generator online:** The ramp function generator accepts the value at *nSet_a* and provides it at its output.
FALSE	**TRUE**	FALSE	TRUE	
FALSE	**TRUE**	TRUE	FALSE	
FALSE	**TRUE**	TRUE	TRUE	
TRUE	FALSE	FALSE	FALSE	**Controller inhibit:** The ramp function generator accepts the value at *nCInhVal_a* and provides it at its output.
TRUE	FALSE	FALSE	TRUE	
TRUE	FALSE	TRUE	FALSE	
TRUE	FALSE	TRUE	TRUE	
TRUE	TRUE	FALSE	FALSE	
TRUE	TRUE	FALSE	TRUE	
TRUE	TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE	

19.1.122.7 S-ramp

A PT1 element is connected downstream of the linear ramp function generator. This arrangement implements an S-shaped ramp for a nearly jerk-free acceleration and deceleration.

- The PT1 element can be switched on/off via the *bSShapeActive* input.
- The corresponding S-ramp time can be set under C00182.

19.1.122.8 Additional setpoint

Use the *nNAdd_a* input to define an additional value (e.g. a correcting signal) and combine it arithmetically with the main setpoint *nNSet_a*.

- First, the additional setpoint is led via a ramp function generator with linear characteristic. Its Ti times can be set in C00220 (acceleration time) and C00221 (deceleration time).
- When the input *bNAddInv* is set to TRUE, the additional setpoint can be inverted before having an effect on the ramp function generator.
- When the input *bLoad* is set to TRUE, the ramp function generator is set to zero for the additional setpoint and held there without considering the Ti times. The same applies when the controller is inhibited.
- The following arithmetic combination of main setpoint and additional setpoint can be selected in C00190:

Value in C00190	Function	Info
0	*nNOut_a* = *nNSet_a*	The additional setpoint *nNAdd_a* is not processed.
1	*nNOut_a* = *nNSet_a* + *nNAdd_a*	
2	*nNOut_a* = *nNSet_a* - *nNAdd_a*	
3	*nNOut_a* = (*nNSet_a* * *nNAdd_a*) / 100 %	Internal scaling: • 100 % ≡ 16384 • 1 % ≡ 164
4	*nNOut_a* = (*nNSet_a* * 1 %) / |*nNAdd_a*|	
5	*nNOut_a* = (*nNSet_a* * 100 %) / (100 % - *nNAdd_a*)	

19.1.122.9 Application example for the additional load function

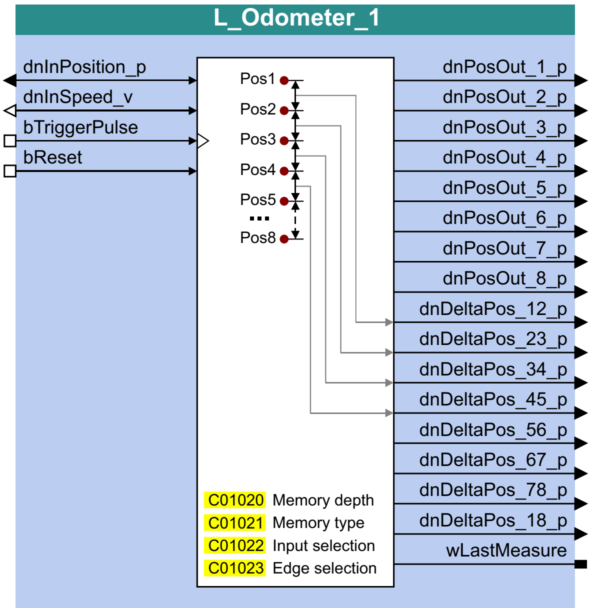
The motor control of the drive is provided with a function for automatically carrying along ramp function generators for "jerk-free" setpoint connection. For speed-controlled drive tasks, the LS_MotorInterface SB outputs the current actual speed value via the *nHlgSetValue_a* output (e.g. in case of a pulse inhibit, flying restart, controller inhibit).

- In case of a pulse inhibit, the main setpoint generator must be carried along with the current actual speed value to ensure jerk-free setpoint transfer.
- The actual speed value is carried along automatically if the following wiring is provided:
  - LS_MotorInterface.*nHlgSetValue_a* ➞ L_NSet_1.*nCInhValue_a*
  - LS_MotorInterface.*bHlgLoad* ➞ L_NSet_1.*bExternalCINH*

19.1.123 L_Odometer_1

This FB can be used to record positions and calculate distances.

- Depending on the input selection set in C01022, a position signal can be detected at the *dnInPosition_p* input or a speed can be detected at the *nInSpeed_v* input.
- The position is detected/accepted via the edge selected in C01023 at the *bTriggerPulse* input.
- If "Ring buffer" is selected in C01021, the measurement will start all over again after the number of measurements selected in C01020 has been performed and the old values will be overwritten. Otherwise, the measurement will stop.



inputs

<table>
<tr><th>Designator
Data type</th><th colspan="2">Information/possible settings</th></tr>
<tr><td>dnInPosition_p
DINT</td><td colspan="2">Position measurement input
• Input is only evaluated if C01022 = "0: Pos input" has been set.</td></tr>
<tr><td>nInSpeed_v
INT</td><td colspan="2">Speed measurement input
• The speed is internally integrated into a position.
• Input is only evaluated if C01022 = "1: V input" has been set.</td></tr>
<tr><td>bTriggerPulse
BOOL</td><td colspan="2">Detect position/speed
• The tripping edge can be parameterised in C01023.</td></tr>
<tr><td rowspan="2">bReset
BOOL</td><td colspan="2">Reset measurement results and internal integrator</td></tr>
<tr><td>TRUE</td><td>Reset measurement results and internal integrator.</td></tr>
</table>

outputs

Designator (Data type)	Value/meaning
dnPosOut_1_p ... dnPosOut_8_p DINT	Detected positions
dnDeltaPos_12_p dnDeltaPos_23_p dnDeltaPos_34_p ... dnDeltaPos_18_p DINT	Calculated distances between the detected positions • *dnDeltaPos_12_p* = Distance between *dnPosOut_1_p* and *dnPosOut_1_p* • *dnDeltaPos_23_p* = Distance between *dnPosOut_2_p* and *dnPosOut_3_p* • *dnDeltaPos_34_p* = Distance between *dnPosOut_3_p* and *dnPosOut_4_p* ... • *dnDeltaPos_18_p* = Distance between *dnPosOut_1_p* and *dnPosOut_8_p*
wLastMeasure WORD	Memory location number (1 ... 8) of the last measurement

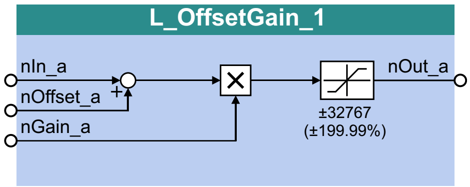
Parameters

Parameters	Possible settings		Information
C01020			Memory length • Lenze setting: 7 measurements
	1	1 measurement	
	2	2 measurements	
	...	...	
	7	7 measurements	
C01021			Memory type • Lenze setting: No ring buffer
	0	No ring buffer	
	1	Ring buffer	
C01022			Input selection • Lenze setting: Pos input
	0	Pos input	
	1	V input	
C01023			Edge selection • Lenze setting: HIGH edge
	0	High edge	
	1	Low edge	
	2	High and low edge	

19.1.124 L_OffsetGain_1

This FB can add an offset to an analog input signal and amplify it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via FB inputs.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %
nOffset_a INT	Offset • Scaling: 16384 ≡ 100 %
nGain_a INT	Gain factor • Scaling: 16384 ≡ 100 % • 199.99 % ≈2

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

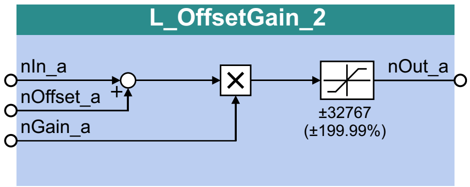
Function

$$nOut\_a = (nIn\_a + Offset) \cdot Gain\ factor$$

19.1.125 L_OffsetGain_2

This FB can add an offset to an analog input signal and amplify it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via FB inputs.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %
nOffset_a INT	Offset • Scaling: 16384 ≡ 100 %
nGain_a INT	Gain factor • Scaling: 16384 ≡ 100 % • 199.99 % ≈2

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

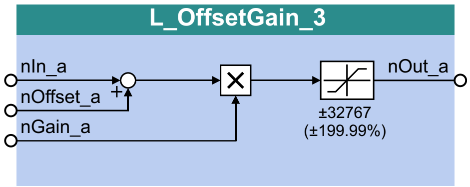
Function

$$nOut\_a = (nIn\_a + Offset) \cdot Gain\ factor$$

19.1.126 L_OffsetGain_3

This FB can add an offset to an analog input signal and amplify it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via FB inputs.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal • Scaling: 16384 ≡ 100 %
nOffset_a INT	Offset • Scaling: 16384 ≡ 100 %
nGain_a INT	Gain factor • Scaling: 16384 ≡ 100 % • 199.99 % ≈2

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

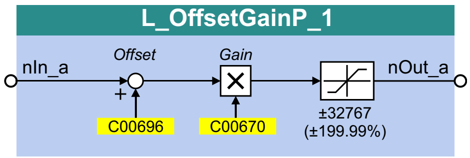
Function

$$\text{nOut\_a} = (\text{nIn\_a} + \text{Offset}) \cdot \text{Gain factor}$$

19.1.127 L_OffsetGainP_1

This FB can add an offset to an analog input signal and amplify it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via parameters.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

Parameters

Parameters	Possible settings			Information
C00670	-100.0000		100.0000	Gain factor • High gain factor for further processing of smallest input signals. • Please observe the difference with regard to the gain factors of other blocks in percent (±199.99 % ≈ 2). • Lenze setting: 1.0000
C00696	-199.99	%	199.99	Offset • Lenze setting: 0.00 %

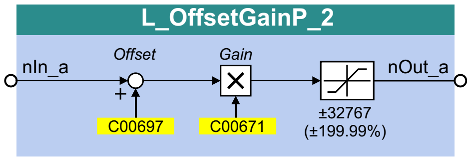
Function

$$nOut\_a = (nIn\_a + Offset) \cdot Gain\ factor$$

19.1.128 L_OffsetGainP_2

This FB can add an offset to an analog input signal and amplify it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via parameters.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

Parameters

Parameters	Possible settings			Information
C00671	-100.0000		100.0000	Gain factor • High gain factor for further processing of smallest input signals. • Please observe the difference with regard to the gain factors of other blocks in percent (±199.99 % ≈ 2). • Lenze setting: 1.0000
C00697	-199.99	%	199.99	Offset • Lenze setting: 0.00 %

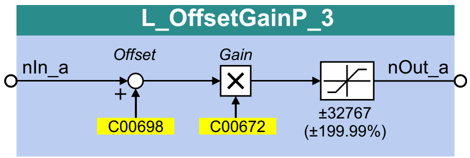
Function

$$nOut\_a = (nIn\_a + Offset) \cdot Gain\ factor$$

19.1.129 L_OffsetGainP_3

This FB can add an offset to an analog input signal and amplify it afterwards. Preferably to be interconnected directly after the analog input terminals.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via parameters.
- The value provided at the *nOut_a* output is internally limited to ±199.99 %.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

outputs

Designator Data type	Value/meaning
nOut_a INT	Output signal • Internal limitation to ±199.99 %

Parameters

Parameters	Possible settings			Information
C00672	-100.0000		100.0000	Gain factor • High gain factor for further processing of smallest input signals. • Please observe the difference with regard to the gain factors of other blocks in percent (±199.99 % ≈ 2). • Lenze setting: 1.0000
C00698	-199.99	%	199.99	Offset • Lenze setting: 0.00 %

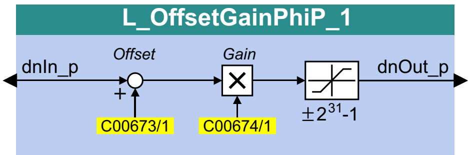
Function

$$\text{nOut\_a} = (\text{nIn\_a} + \text{Offset}) \cdot \text{Gain factor}$$

19.1.130 L_OffsetGainPhiP_1

This FB can add an offset to an angle signal and amplify it afterwards.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via parameters.
- The value provided at the *dnOut_p* output is internally limited to $\pm 2^{31}$-1.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Parameters

Parameters	Possible settings			Information
C00673/1	-2147483647	Incr.	2147483647	Offset • Lenze setting: 0 incr.
C00674/1	-2147483647		2147483647	Gain factor • Scaling: 65535 ≡ gain factor 1 • Lenze setting: 65536

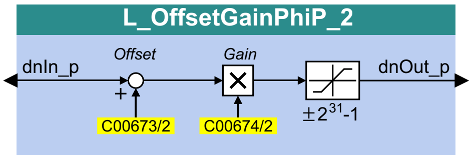
Function

$$dnOut\_p = (dnIn\_p + Offset) \cdot Gain\ factor$$

19.1.131 L_OffsetGainPhiP_2

This FB can add an offset to an angle signal and amplify it afterwards.

- The internal calculations (addition and subtraction) are carried out with 32 bits without overflow/underflow. Division is not remainder considered.
- Offset and gain are selected via parameters.
- The value provided at the *dnOut_p* output is internally limited to $\pm 2^{31}$-1.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to $\pm 2^{31}$-1 (±2147483647)

Parameters

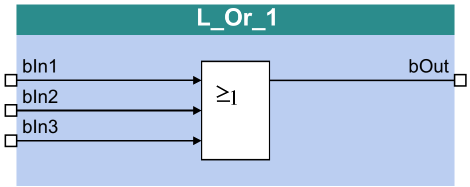
Parameters	Possible settings			Information
C00673/2	-2147483647	Incr.	2147483647	Offset • Lenze setting: 0 incr.
C00674/2	-2147483647		2147483647	Gain factor • Scaling: 65535 ≡ gain factor 1 • Lenze setting: 65536

Function

$$\text{dnOut\_p} = (\text{dnIn\_p} + \text{Offset}) \cdot \text{Gain factor}$$

19.1.132 L_Or_1

This FB implements the OR operation of input signals.



inputs

Designator Data type	Information/possible settings
bIn1 bIn2 bIn3 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

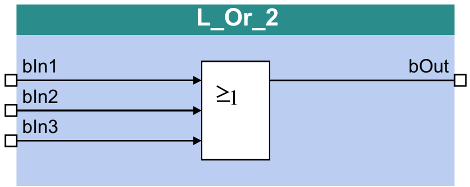
Function

inputs			Output
bIn3	bIn2	bIn1	bOut
FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	TRUE
FALSE	TRUE	FALSE	
FALSE	TRUE	TRUE	
TRUE	FALSE	FALSE	
TRUE	FALSE	TRUE	
TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	

[19-57] Truth table of the FB L_Or_1

19.1.133 L_Or_2

This FB implements the OR operation of input signals.



inputs

Designator Data type	Information/possible settings
bIn1 bIn2 bIn3 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

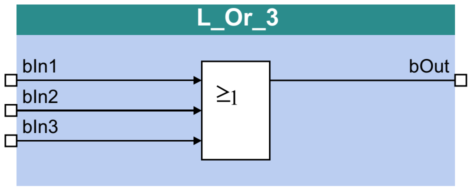
Function

inputs			Output
bIn3	bIn2	bIn1	bOut
FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	TRUE
FALSE	TRUE	FALSE	
FALSE	TRUE	TRUE	
TRUE	FALSE	FALSE	
TRUE	FALSE	TRUE	
TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	

[19-58] Truth table of the FB L_Or_2

19.1.134 L_Or_3

This FB implements the OR operation of input signals.



inputs

Designator Data type	Information/possible settings
bIn1 bIn2 bIn3 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

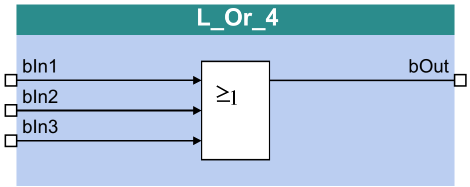
Function

inputs			Output
bIn3	**bIn2**	**bIn1**	**bOut**
FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	TRUE
FALSE	TRUE	FALSE	
FALSE	TRUE	TRUE	
TRUE	FALSE	FALSE	
TRUE	FALSE	TRUE	
TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	

[19-59] Truth table of the L_Or_3 FB

19.1.135 L_Or_4

This FB implements the OR operation of input signals.



inputs

Designator Data type	Information/possible settings
bIn1 bIn2 bIn3 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

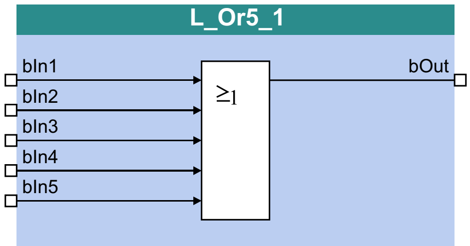
Function

inputs			Output
bIn3	bIn2	bIn1	bOut
FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	TRUE
FALSE	TRUE	FALSE	
FALSE	TRUE	TRUE	
TRUE	FALSE	FALSE	
TRUE	FALSE	TRUE	
TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	

[19-60] Truth table of the L_Or_4 FB

19.1.136 L_Or5_1

This FB implements the OR operation of input signals.



inputs

Designator Data type	Information/possible settings
bIn1 ... bIn5 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

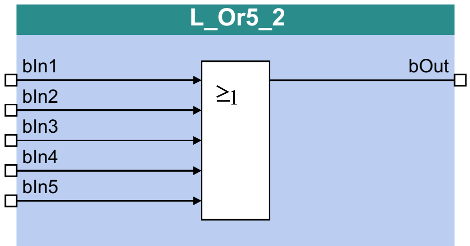
Function

inputs					Output
bIn5	**bIn4**	**bIn3**	**bIn2**	**bIn1**	**bOut**
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	FALSE	FALSE	TRUE	TRUE
FALSE	FALSE	FALSE	TRUE	FALSE	
FALSE	FALSE	FALSE	TRUE	TRUE	
FALSE	FALSE	TRUE	FALSE	FALSE	
...					
TRUE	TRUE	TRUE	FALSE	TRUE	
TRUE	TRUE	TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE	TRUE	

[19-61] Truth table of the FB L_Or5_1

19.1.137 L_Or5_2

This FB implements the OR operation of input signals.



inputs

Designator Data type	Information/possible settings
bIn1 ... bIn5 BOOL	Input signal

outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal

Function

inputs					Output
bIn5	**bIn4**	**bIn3**	**bIn2**	**bIn1**	**bOut**
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	FALSE	FALSE	TRUE	TRUE
FALSE	FALSE	FALSE	TRUE	FALSE	
FALSE	FALSE	FALSE	TRUE	TRUE	
FALSE	FALSE	TRUE	FALSE	FALSE	
...					
TRUE	TRUE	TRUE	FALSE	TRUE	
TRUE	TRUE	TRUE	TRUE	FALSE	
TRUE	TRUE	TRUE	TRUE	TRUE	

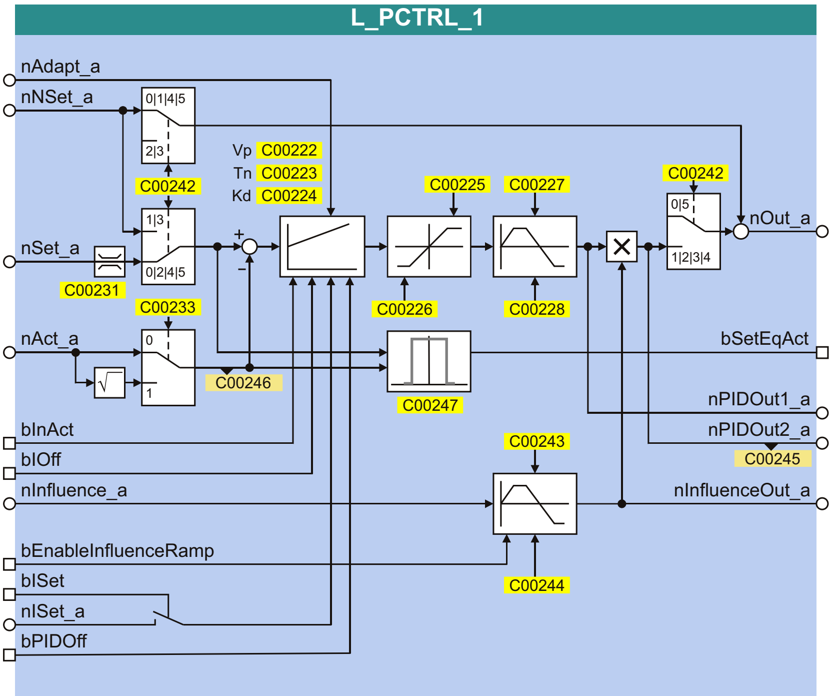
[19-62] Truth table of the FB L_Or5_2

19.1.138 L_PCTRL_1

This FB is a PID controller and can be used for various control tasks (e.g. as dancer position controller, tension controller, or pressure controller).

The FB is provided with the following functions:

- Adjustable control algorithm (P, PI, PID)
- Ramp function generator for preventing setpoint step-changes at the input
- Limitation of the controller output
- Factorisation of the output signal
- Vp adaptation
- Integral action component can be switched off
- Comparison function "Actual value = setpoint"



inputs

Designator Data type	Information/possible settings
nAdapt_a INT	Percentage adaptation of the gain Vp set in C00222 and the reset time Tn set in C00223. Then, the adapted reset time is as follows: $$Tn = \frac{Vp_{C222} \cdot nAdapt\_a\,[\%] \cdot Tn_{C223}\,[ms]}{100\,[\%]}$$ • Internal limitation to ± 199.99 % • Changes can be done online. • Display parameter: C00830/62
nNset_a INT	Speed setpoint • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 % • Display parameter: C00830/89

Designator Data type	Information/possible settings	
nSet_a INT	Sensor and process setpoint for operating modes 2, 4 and 5 • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 % • Display parameter: C00830/63	
nAct_a INT	Speed or actual sensor value (actual process value) • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 % • Display parameter: C00830/61	
bInAct BOOL	Deactivate process controller temporarily (stop) • Changes can be done online. • Display parameter: C00833/76 Note: This input is not interconnected in the LA_NCtrl application block.	
	TRUE	• The current output value is frozen. • The internal control algorithm is stopped. • However, a setpoint selected via input *nNSet_a* is still provided in operating modes 0/1/4/5.
bIOff BOOL	Switch off the I-component of the process controller • Changes can be done online. • Display parameter: C00833/77	
	TRUE	The I component of the process controller is set to zero.
nInfluence_a INT	Limitation of the influencing factor in percent • *nInfluence_a* serves to limit the influencing factor of the PID controller contained in the FB to a required value (- 199.99 % ... + 199.99 %). • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 % • Display parameter: C00830/64	
bEnableInfluenceRamp BOOL	Activate ramp for influencing factor • Display parameter: C00833/106	
	TRUE	Influencing factor of the PID controller is ramped up to the *nInfluence_a* value.
	FALSE	Influencing factor of the PID controller is ramped down to "0".
bISet BOOL	Accept I component *nISet_a* in PID controller	
	TRUE	The value at the input *nISet_a* is accepted in the PID controller.
nISet_a INT	Selection of I component of PID controller • With a TRUE signal at *bISet*, the assigned value is accepted in the PID controller. • Scaling: 16384 ≡ 100 % • Internal limitation to ± 199.99 %	
bPIDOff BOOL	Reset the entire PID controller	
	TRUE	• The I component of the controller is set to zero. • The controller output is set to zero. • The internal control algorithm is stopped.

outputs

Designator Data type	Value/meaning	
nOut_a INT	Output signal • Internal limitation to ±32767 (±199.99 %) • Scaling: 16384 ≡ 100 %	
bSetEqAct INT	Comparison function "Actual value = setpoint" • The window for the comparison operation can be set in C00247.	
	TRUE	Setpoint and actual value are identical, i.e. no system deviation available.
nPIDOut1_a INT	PID controller output without influencing factor *nInfluence_a* • Inputs *bEnableInfluenceRamp* and *nInfluence_a* do not have any effect here, the limited PID output value influenced by the internal ramp times is output. • There is no connection with the additive input *nNSet_a*. • Scaling: 16384 ≡ 100 %	
nPIDOut2_a INT	PID controller output with influencing factor *nInfluence_a*. • There is no connection with the additive input *nNSet_a*. • Scaling: 16384 ≡ 100 % • Display parameter: C00245	
nInfluenceOut_a INT	Current influencing factor ("ramp status") on the PID output value • Scaling: 16384 ≡ 100 %	

Parameters

Parameters	Possible settings			Information
C00222	0.1	0.1	500.0	Gain Vp • Lenze setting: 1.0
C00223	20	ms	6000	Reset time Tn • Lenze setting: 400 ms
C00224	0.0	0.1	5.0	Differential component Kd • Lenze setting: 0.0
C00225	-199.99	%	+199.99	MaxLimit • Maximum value of the PID operating range • Lenze setting: 199.99 %
C00226	-199.99	%	+199.99	MinLimit • Minimum value of the PID operating range • Lenze setting: -199.99 %
C00227	0.000	s	999.999	Acceleration time for the ramp at the PID output (should be set as steep as possible) • Lenze setting: 0.010 s
C00228	0.000	s	999.999	Deceleration time for the ramp at the PID output • Lenze setting: 0.010 s
C00231/1 (Pos. Maximum) C00231/2 (Pos. Minimum) C00231/3 (Neg. Minimum) C00231/4 (Neg. Maximum)	0.00	%	199.99	Operating range • Determination of the operating range for the PID process controller by limiting the input signal *nSet_a*. • Lenze setting: No limitation (-199.99 % ... +199.99 %)

Parameters	Possible settings				Information
C00233					Root function • Lenze setting: "0: Off"
	0	Off			The actual value at *nAct_a* is not changed for further processing.
	1	On			The square root of the actual value at *nAct_a* is taken for further processing.
C00242					Operating mode • Lenze setting: "0: Off"
	0	Off			The input setpoint *nNSet_a* is output without any changes at the output *nOut_a*.
	1	nNSet + nNSet_PID			*nNSet_a* and *nAct_a* are used as PID input values. The arriving *nNSet_a* is additively linked to the value output by the PID element.
	2	nSet_PID			*nSet_a* and *nAct_a* are used as PID input values. The input *nNSet_a* is not considered.
	3	nNSet_PID			*nNSet_a* and *nAct_a* are used as PID input values. The input *nSet_a* is not considered.
	4	nNSet + nSet_PID			*nSet_a* and *nAct_a* are used as PID input values. The arriving *nNSet_a* setpoint is additively linked to the value output by the PID element.
	5	nNSet || nSet_PID			*nSet_a* and *nAct_a* are used as PID input values. The setpoint *nNSet_a* is output at the output *nOut_a* . The PID output value is output at the output *nPIDOut_a*.
C00243	0.000		s	999.999	Influence acceleration time • Acceleration time T_{ir} for the influencing factor. • Lenze setting: 5.000 s
C00244	0.000		s	999.999	Influence deceleration time • Deceleration time T_{if} for the influencing factor. • Lenze setting: 5.000 s
C00245	-199.99		%	+199.99	Display of PID output value *nPIDOut_a*
C00246	-199.99		%	+199.99	Display of the internal PID input value *nAct_a*
C00247	0		%	100	Window for comparison function "Actual value = setpoint" • Lenze setting: 2 % • Hysteresis: 1 % (fixed)

19.1.138.1 Control characteristic

The PI algorithm is active in the Lenze setting.

Gain (P component)

The input value is controlled by a linear characteristic. The slope of the characteristic is determined by the controller gain Vp.

The controller gain Vp is set under C00222.

- The controller gain can be adapted via the input *nAdapt_a* (also possible in online mode).
- The input value *nAdapt_a* has a direct effect on the controller gain:

$$P = \text{nAdapt\_a} \cdot \text{C00222}$$

Example: With the parameterised controller gain Vp = 2.0 and *nAdapt_a* = 75 %, the resulting gain factor is as follows:

$$P = \frac{75\,[\%]}{100\,[\%]} \cdot 2.0 = 1.5$$

Integral action component (I component)

The I component can be selected via the input *nISet_a*. With a TRUE signal at *bISet*, the assigned value is accepted in the PID controller.

- Setting the adjustment time Tn to the maximum value of "6000 ms" deactivates the I component.
- The I component of the controller can also be deactivated by setting the input *bIOff* to TRUE.
- The I component can be switched on and off online.

Reset time

The adjustment time Tn is set under C00223.

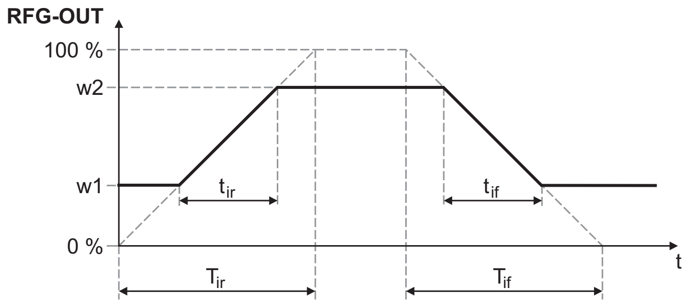
Differential component Kd (D component)

The differential component Kd is set under C00224.

- The setting "0.0 s" deactivates the D component (Lenze setting). In this way, the PID controller becomes a PI controller or P controller, if the I component has been deactivated as well.

19.1.138.2 Ramp function generator

The PID output is led via a ramp function generator with linear characteristic. This serves to transfer setpoint step-changes at the PID output into a ramp which should be as steep as possible.



w1, w2 = change of the main setpoint as a function of t_{ir} and t_{if}
RFG-OUT = output of the ramp function generator

[19-63] Acceleration and deceleration times

- t_{ir} and t_{if} are the desired times for changing between w1 and w2.
- The ramps for acceleration and deceleration can be set individually.
  - C00227: Acceleration time t_{ir}
  - C00228: Deceleration time t_{if}
- The t_{ir}/t_{if} values are converted into the required Ti times according to the following formula:

$T_{ir} = t_{ir} \cdot \frac{100\ \%}{w2 - w1}$	$T_{if} = t_{if} \cdot \frac{100\ \%}{w2 - w1}$

- The ramp function generator is immediately set to "0" by setting *bInAct* to TRUE.

19.1.138.3 Operating range of the PID process controller

The value range of the input signal *nSet_a* and thus the operating range of the PID process controller can be limited with the following parameters:

- C00231/1: Pos. maximum (default setting: 199.99 %)
- C00231/2: Pos. minimum (default setting: 0.00 %)
- C00231/3: Neg. minimum (default setting: 0.00 %)
- C00231/4: Neg. maximum (default setting: 199.99 %)

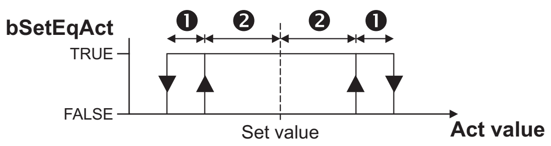
19.1.138.4 Evaluation of the output signal

After the limitation, the output signal is evaluated with the influencing factor *nInfluence_a*. The evaluation is activated/suppressed along a ramp when the *bEnableInfluenceRamp* input is set to TRUE. The ramp times are set with the parameters "Influence acceleration time" (C00243) and "Influence deceleration time" (C00244).

19.1.138.5 Comparison function "Actual value = setpoint"

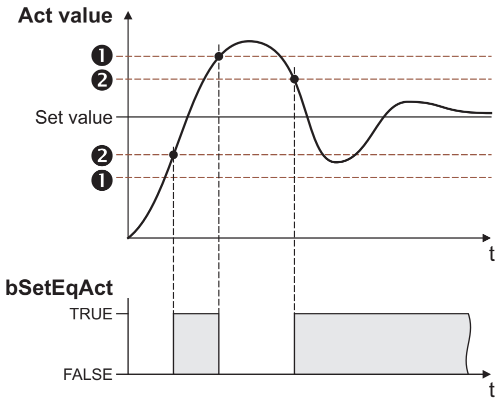
If setpoint and actual value are identical and there is no system deviation, the *bSetEqAct* status output is set to TRUE.

- The hysteresis of the comparison function has a fixed value of 1 %.
- The symmetrical window around the setpoint for the comparison function can be set in C00247 (Lenze setting: 2 %).



❶ Hysteresis (1 %, fixed value)
❷ C00247: Window

[19-64] Comparison function: Switching performance



❶ Hysteresis (1 %, fixed value)
❷ C00247: Window

[19-65] Comparison function: Example

19.1.138.6 Control functions

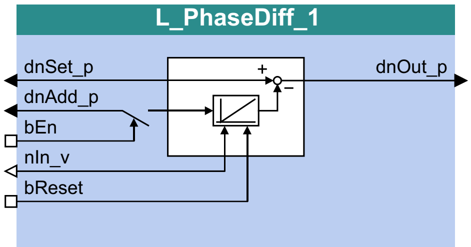
The process controller has various digital inputs for controlling the FB:

Designator Data type	Information/possible settings	
bInAct BOOL	Deactivate process controller temporarily (stop) • Changes can be done online. • Display parameter: C00833/76 Note: This input is not interconnected in the LA_NCtrl application block.	
	TRUE	• The current output value is frozen. • The internal control algorithm is stopped. • However, a setpoint selected via input *nNSet_a* is still provided in operating modes 0/1/4/5.
bIOff BOOL	Switch off the I-component of the process controller • Changes can be done online. • Display parameter: C00833/77	
	TRUE	The I component of the process controller is set to zero.
bPIDOff BOOL	Reset the entire PID controller	
	TRUE	• The I component of the controller is set to zero. • The controller output is set to zero. • The internal control algorithm is stopped.

19.1.139 L_PhaseDiff_1

This FB generates a position difference for the specified position setpoint from a position value and a speed signal.

- In an integrator, the *nIn_v* speed signal is integrated into a position value and subtracted from the *dnSet_p* position setpoint.
- In addition, the adaptive *dnAdd_p* position value can be added to the integrator content by setting *bEn* to TRUE.



inputs

Designator Data type	Information/possible settings	
dnSet_p DINT	Selection of a position setpoint	
dnAdd_p DINT	Adaptive position value for the actual position	
bEn BOOL	Activate addition of the adaptive position value	
	FALSE	1. The speed signal at *nIn_v* is integrated by the angle integrator. 2. The result of the angle integrator is subtracted from the angle signal at *dnSet_p* and output at *dnOut_p* afterwards.
	TRUE	The adaptive position value selected via *dnAdd_p* is added: 1. The speed signal at *nIn_v* is integrated by the angle integrator. 2. The angle signal at *dnAdd_p* is added to the integrated speed signal in every task cycle. 3. The result of the angle integrator is subtracted from the angle signal at *dnSet_p* and output at *dnOut_p* afterwards.
nIn_v INT	Selection of the actual speed to be converted into the position value	
bReset BOOL	Reset actual angle integrator	
	TRUE	Actual angle integrator is set to "0".

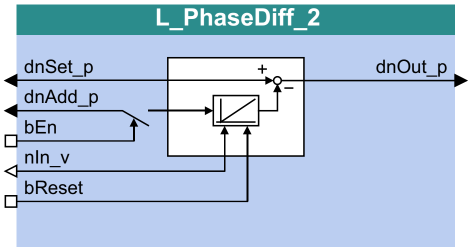
outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output angle signal (position difference) • Without limitation

19.1.140 L_PhaseDiff_2

This FB generates a position difference for the specified position setpoint from a position value and a speed signal.

- In an integrator, the *nIn_v* speed signal is integrated into a position value and subtracted from the *dnSet_p* position setpoint.
- In addition, the adaptive *dnAdd_p* position value can be added to the integrator content by setting *bEn* to TRUE.



inputs

Designator Data type	Information/possible settings	
dnSet_p DINT	Selection of a position setpoint	
dnAdd_p DINT	Adaptive position value for the actual position	
bEn BOOL	Activate addition of the adaptive position value	
	FALSE	1. The speed signal at *nIn_v* is integrated by the angle integrator. 2. The result of the angle integrator is subtracted from the angle signal at *dnSet_p* and output at *dnOut_p* afterwards.
	TRUE	The adaptive position value selected via *dnAdd_p* is added: 1. The speed signal at *nIn_v* is integrated by the angle integrator. 2. The angle signal at *dnAdd_p* is added to the integrated speed signal in every task cycle. 3. The result of the angle integrator is subtracted from the angle signal at *dnSet_p* and output at *dnOut_p* afterwards.
nIn_v INT	Selection of the actual speed to be converted into the position value	
bReset BOOL	Reset actual angle integrator	
	TRUE	Actual angle integrator is set to "0".

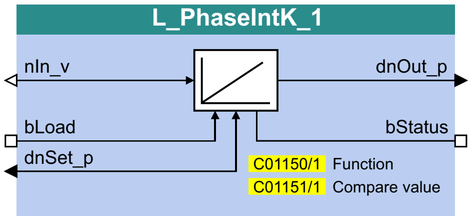
outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output angle signal (position difference) • Without limitation

19.1.141 L_PhaseIntK_1

This FB can integrate a speed or a velocity to an angle (path). The FB can also detect a relatively completed distance.

- The integrator can take max. ±32000 encoder revolutions.



inputs

Designator Data type	Information/possible settings	
nIn_v INT	Selection of the actual speed • 16384 ≡ 15000 rpm	
bLoad BOOL	Load angle integrator with starting value and reset status signal	
	TRUE	Angle integrator is loaded with the value at *dnSet_p* and *bStatus* is reset to FALSE.
dnSet_p DINT	Starting value for angle integrator	

outputs

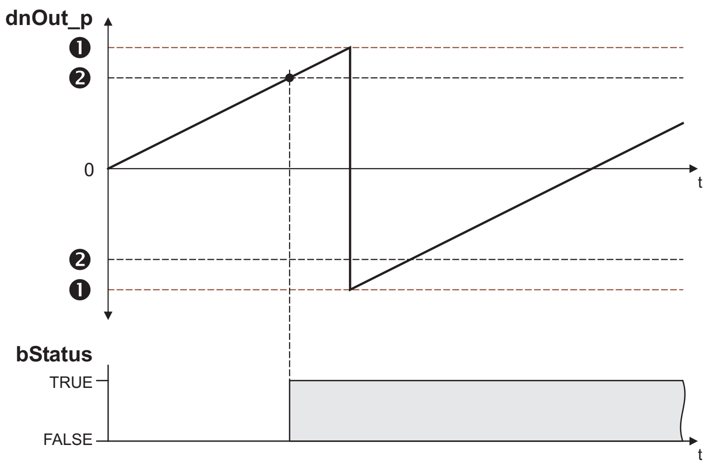
Designator Data type	Value/meaning	
dnOut_p DINT	Angle output signal • 65536 [incr.] ≡ 1 encoder revolution • Overflow is possible (display via *bStatus*)	
bStatus BOOL	Status signal "Overflow occurred/distance processed" • Status signal can be reset via *bLoad*.	
	TRUE	Overflow has occurred or distance is processed.

Parameters

Parameters	Possible settings			Information
C01150/1				Function
	0	Loading with level		Load integrator with TRUE level at the input *bLoad*(Lenze setting)
	1	Loading with edge		Load integrator with FALSE/TRUE edge at the input *bLoad*.
	2	Loading with level + reset		Load integrator when reaching the comparison value or with TRUE level at the input *bLoad*.
C01151/1	0		2000000000	Comparison value • Is valid for both the positive and the negative value range. • Lenze setting: 0

19.1.141.1 Function at constant input value

Selection: C01150 = "0: Loading with level" or "1: Loading with edge"



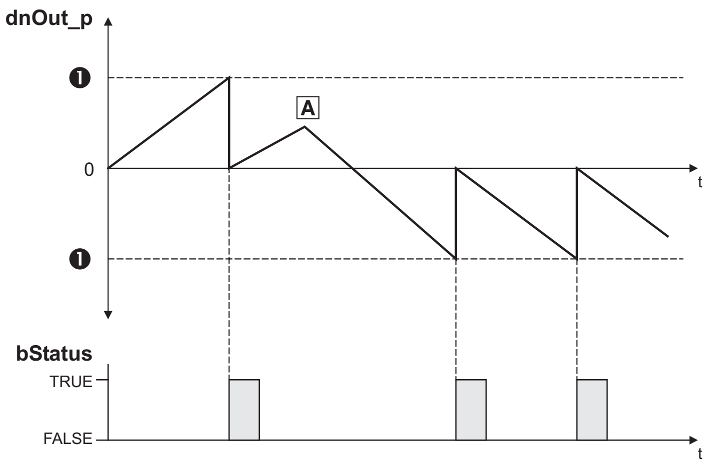
❶ Overflow at ±2147483648 incr. (±32767 encoder revolutions)
❷ C01151: Comparison value (applies to both positive and negative value range)

[19-66] Switching performance if the overflow is in the positive direction

- If "0: Loading with level" is selected in C01150, the *bLoad* input is status-controlled: In case of a TRUE signal, the integrator is loaded with the value at *dnSet_p* and the *bStatus* output is set to FALSE.
- If "1: Loading with edge" is selected in C01150, the *bLoad* input is edge-controlled: In case of a FALSE/TRUE edge, the integrator is loaded with the value at *dnSet_p* and then immediately continues to integrate, the *bStatus* output is set to FALSE.
- A positive signal at *nIn_v* is incremented (the counter content is increased with every function call).
- A negative signal at *nIn_v* is decremented (the counter content is reduced with every function call).
- *dnOut_p* outputs the counter content of the bipolar integrator.
  - If the counter content exceeds a value of +32767 encoder revolutions (corresponds to +2147483647 incr.), an overflow occurs and the counting process continues at a value of -32768 encoder revolutions.
  - If the counter content falls below a value of -32768 encoder revolutions (corresponds to -2147483648 incr.), an overflow occurs and the counting process starts at a value of +32767 encoder revolutions.
- *bStatus* will be set to TRUE if the comparison value set in C01151/x is reached.

19.1.141.2 Function at input value with sign reversal

Selection: C01150 = "2: Loading with level + reset"



A Sign reversal of the value at *nIn_v*
❶ C01151/x: Comparison value (is valid for both the positive and the negative value range)

[19-67] Switching performance if the input signal changes signs

- If "2: Loading with level + reset" is selected in C01150, the *bLoad* input is status-controlled: In case of a TRUE signal, the integrator is loaded with the value at *dnSet_p* and the *bStatus* output is set to FALSE.
- A positive signal at *nIn_v* is incremented (the counter content is increased with every function call).
- A negative signal at *nIn_v* is decremented (the counter content is reduced with every function call).
- *dnOut_p* outputs the counter content of the bipolar integrator.
  - If the positive counter content is higher than the comparison value set in C01151/x, the comparison value will be subtracted from the counter content, and *bStatus* will be set to TRUE for one task cycle.
  - If the negative counter content is lower than the comparison value set in C01151/x, the comparison value will be added to the counter content, and *bStatus* will be set to TRUE for one task cycle.

19.1.141.3 Calculation of the output signal

The output value at *dnOut_p* can be detected according to the following formula:

$$\text{dnOut\_p [incr.]} = \text{nIn\_v [rpm]} \cdot \text{t [s]} \cdot \text{65535 [incr./rev.]}$$

t = integration time
16384 ≡ 15000 rpm
1 ≡ 1 incr.

Example

You want to determine the counter content of the integrator at a certain speed at the input and a certain integration time **t**.

Given values:

- *nIn_v* = 1000 rpm ≈ integer value 1092
- Integration time t = 10 s
- Starting value of the integrator = 0

Solution:

- Conversion of the input signal at *nIn_v*:

$$1000\text{ rpm} = \frac{1000\text{ rev.}}{60\text{ s}}$$

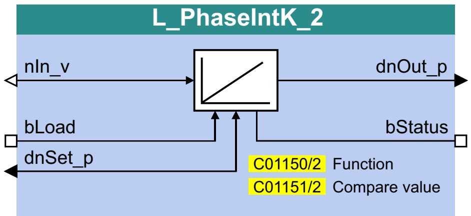
- Calculation of the output value:

$$\text{dnOut\_p} = \frac{1000\text{ rev.}}{60\text{ s}} \cdot 10\text{ s} \cdot \frac{65535\text{ incr.}}{\text{Rev.}} = 10922666\text{ incr.}$$

19.1.142 L_PhaseIntK_2

This FB can integrate a speed or a velocity to an angle (path). The FB can also detect a relatively completed distance.

- The integrator can take max. ±32000 encoder revolutions.



inputs

Designator	Data type	Information/possible settings	
nIn_v	INT	Selection of the actual speed • 16384 ≡ 15000 rpm	
bLoad	BOOL	Load angle integrator with starting value and reset status signal	
		TRUE	Angle integrator is loaded with the value at *dnSet_p* and *bStatus* is reset to FALSE.
dnSet_p	DINT	Starting value for angle integrator	

outputs

Designator	Data type	Value/meaning	
dnOut_p	DINT	Angle output signal • 65536 [incr.] ≡ 1 encoder revolution • Overflow is possible (display via *bStatus*)	
bStatus	BOOL	Status signal "Overflow occurred/distance processed" • Status signal can be reset via *bLoad*.	
		TRUE	Overflow has occurred or distance is processed.

Parameters

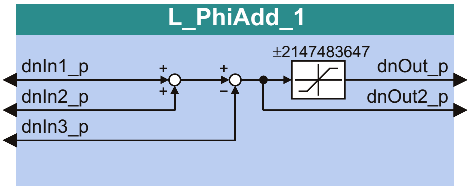
Parameters	Possible settings			Information
C01150/2				Function
	0	Loading with level		Load integrator with TRUE level at the input *bLoad*(Lenze setting)
	1	Loading with edge		Load integrator with FALSE/TRUE edge at the input *bLoad*.
	2	Loading with level + reset		Load integrator when reaching the comparison value or with TRUE level at the input *bLoad*.
C01151/2	0		2000000000	Comparison value • Is valid for both the positive and the negative value range. • Lenze setting: 0

For a detailed functional description see L_PhaseIntK_1.

19.1.143 L_PhiAdd_1

This FB adds or subtracts angle signals depending on the input used.

- The value provided at the *dnOut_p* output is internally limited to ±2147483647 ($\pm 2^{31}$ - 1).
- The value provided at the *dnOut2_p* output is not limited internally, this enables an internal overflow.



inputs

Designator Data type	Information/possible settings
dnIn1_p DINT	Input signal 1
dnIn2_p DINT	Input signal 2 (addition)
dnIn3_p DINT	Input signal 3 (subtraction)

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647
dnOut2_p DINT	Output signal 2 • Without limitation/with overflow (which makes a wrong output possible)

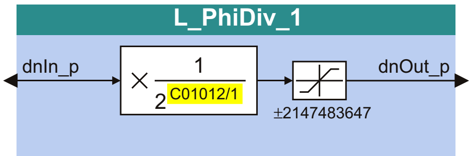
Function

$$\text{dnOut\_p} = \text{dnIn1\_p} + \text{dnIn2\_p} - \text{dnIn3\_p}$$

19.1.144 L_PhiDiv_1

This FB divides or multiplies angle signals in power of two format.

- A positive division factor (C01012/1) causes a division.
- A negative division factor (C01012/1) causes a multiplication.
- The value provided at the *dnOut_p* output is internally limited to ±2147483647 (±2^{31} - 1).
- Division is not remainder considered.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal • Internal limitation to ±2147483647

Parameters

Parameters	Possible settings			Information
C01012/1	-31		31	Division factor • Lenze setting: 0

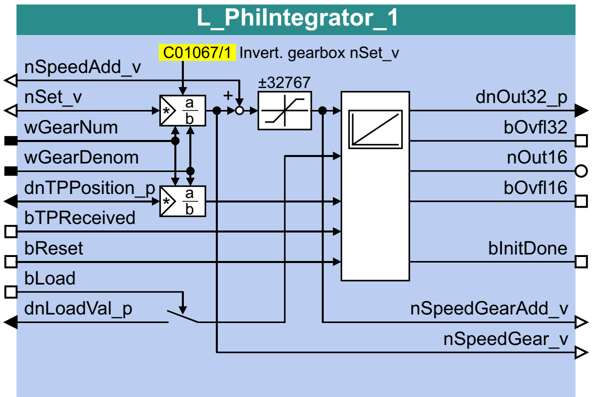
Function

$$dnOut\_p = dnIn\_p \cdot \frac{1}{2^{C01012/1}}$$

19.1.145 L_PhiIntegrator_1

This FB evaluates a speed with a gearbox factor and adds them in an integrator.

- The integrator value is output as 16-bit value and 32-bit value.
- An overflow of the prevailing output is signalled.
- The integrator can be loaded and initialised (e.g. with a touch probe signal).



inputs

Designator Data type	Information/possible settings	
nSpeedAdd_v INT	Additive speed	
nSet_v INT	Speed input (angular difference)	
wGearNum WORD	Gearbox factor (numerator)	
wGearDenom WORD	Gearbox factor (denominator) • When *wGearDenom* = 0, it is internally calculated with 1.	
dnTPPosition_p DINT	Initialisation value	
bTPReceived BOOL	Initialise integrator	
	FALSE↗TRUE	Initialise integrator
bReset BOOL	Reset integrator	
	TRUE	Reset integrator • The integrator is not integrated.
bLoad BOOL	Load integrator	
	TRUE	Load integrator with *dnLoadVal_p* • The integrator is not integrated. • The *bInitDone* output is reset to FALSE with a delay of one call cycle after *bLoad* is reset to FALSE and a subsequent FALSE/TRUE edge to *bTPReceived*.
dnLoadVal_p DINT	Value the integrator is to be loaded with.	

outputs

Designator Data type	Value/meaning	
dnOut32_p DINT	Integrator value (32 bits)	
bOvfl32 BOOL	Status signal "overflow (32 bits)"	
	TRUE	Overflow
nOut16 INT	Integrator value (16 bits)	
bOvfl16 BOOL	Status signal "overflow (16 bits)"	
	TRUE	Overflow
bInitDone BOOL	Status signal "Initialisation completed"	
	TRUE	Integrator is initialised
nSpeedGearAdd_v INT	Resulting speed with additive speed (*nSet_v* * gearbox factor + *nSpeedAdd_v*)	
nSpeedGear_v INT	Resulting speed without additive speed (*nSet_v* * gearbox factor)	

Parameters

Parameters	Possible settings		Information
C01067/1			Invert. gearbox nSet_v
	0	Not inverted	
	1	Inverted	
	2	Automatically from MCK	

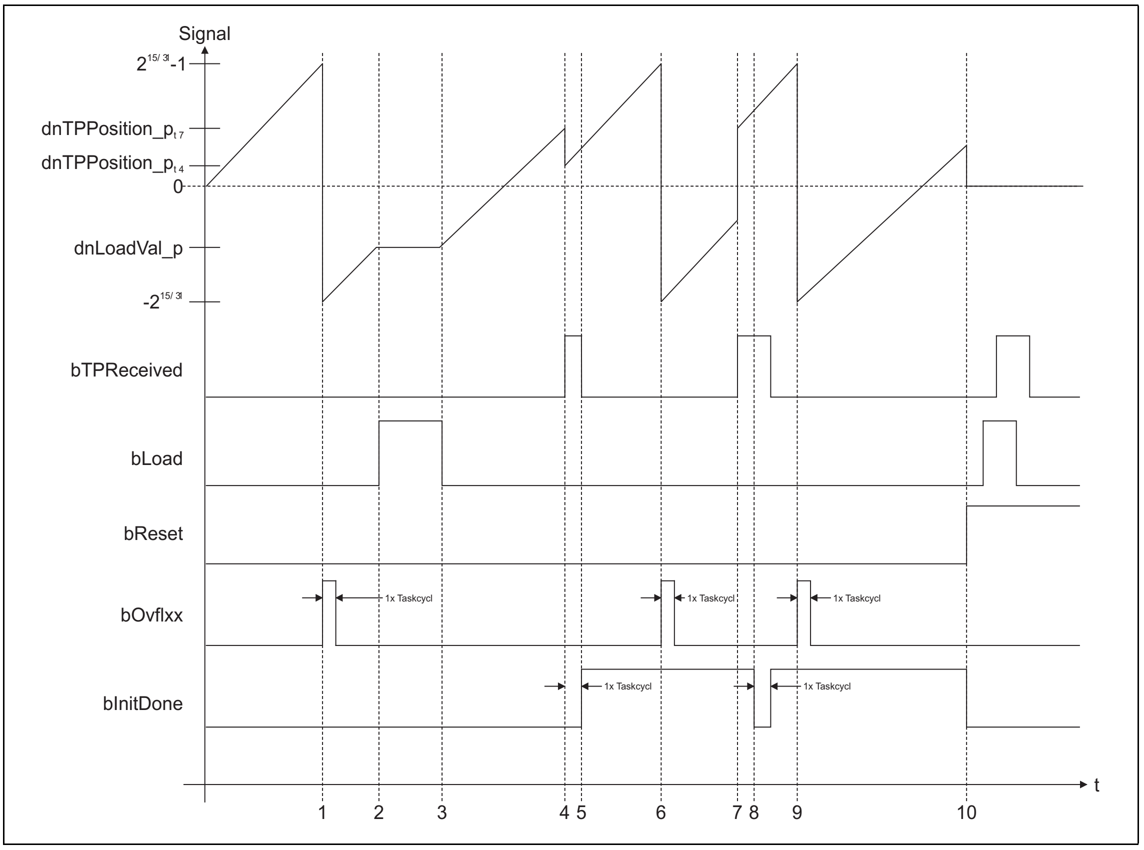
19.1.145.1 Function

The *nSet_v* input value is evaluated, integrated and output as 16-bit value (*nOut16*) and 32-bit value *dnOut32_p*) with the gearbox factors *wGearNum* and *wGearDenom*.

$$nOut16 = nOut16 + nSet\_v \cdot \frac{wGearNum}{wGearDenom}$$

$$dnOut32\_p = dnOut32\_p + nSet\_v \cdot \frac{wGearNum}{wGearDenom}$$

- Evaluation of the gearbox factor is remainder considered.
- A positive or negative overflow of the integrator is signalised by a TRUE signal (for a task cycle) at the *bOvfl16* output for the *nOut16* output or at the *bOvfl32* output for the *dnOut32_p* output.



[19-68] Signal characteristic

19.1.145.2 Example

The current speed at *nSet_v* and the gearbox factors serve to create a specific machine measuring system from the motor measuring system.

- 216 increments in the motor measuring system are to correspond to 1 motor revolution.
- 216 increments in the machine measuring system (*nOut16*) are to correspond to 1 machine shaft revolution.

Initialise integrator

A FALSE/TRUE edge at *bTPReceived* initialises the integrator, i.e. a calculation with *dnTPPosition_p* and *dnLoadVal_p* is made:

$$\text{nOut16} = \text{(INT)}(\text{dnLoadVal\_p} + \text{dnTPPosition\_p} \cdot \text{Gearbox factor})$$
$$\text{dnOut32\_p} = \text{dnLoadVal\_p} + \text{dnTPPosition\_p} \cdot \text{Gearbox factor}$$

- After initialisation is completed, the *bInitDone* output is set to TRUE in the next call cycle.
- In case of a renewed initialisation without a previous reset (*bReset* = TRUE), the *bInitDone* output is set to FALSE for a task cycle.

Reset integrator

When *bReset* is set to TRUE, the integrator is set to 0. No calculations are made.

- The Boolean outputs are set to FALSE.
- The *bReset* input has the highest priority.

Load integrator

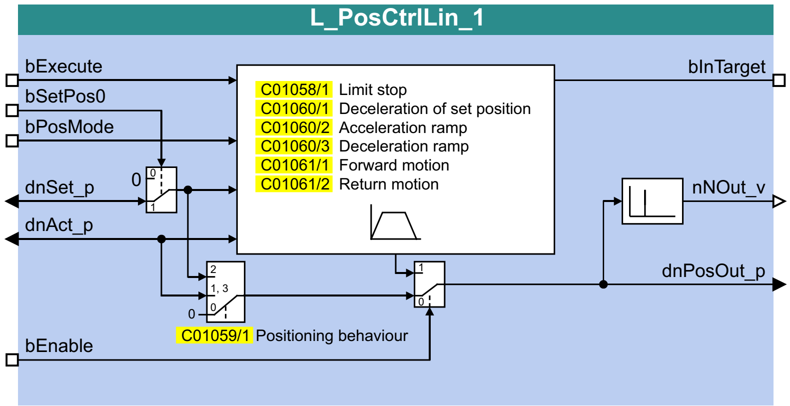
When *bLoad* is set to TRUE, the integrator is loaded with the value at *dnLoadVal_p* and the outputs are set.

- When *bLoad* is set to TRUE, the input is not integrated and no check for overflow is executed.
- When *bLoad* is reset to FALSE, the integrator continues from the loaded value. The *bInitDone* output is reset to FALSE after a FALSE/TRUE edge at *bTPReceived* with a delay of one call cycle.

19.1.146 L_PosCtrlLin_1

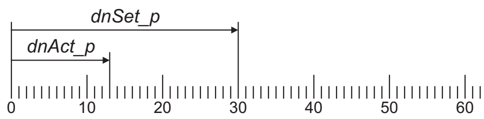
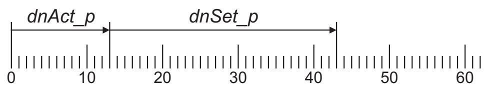
This FB serves to implement the following positioning functions:

- Bring drive to curve position (e.g. after mains connection, manual jog, homing)
- Release drive from curve and bring it to safe position (position override function)
- Positioning the X axis via X offset (higher-level positioning)
- Positioning the Y axis via Y offset (higher-level positioning)



inputs

Designator (Data type)		Information/possible settings
bExecute (BOOL)		Execute positioning profile • Only possible if the *bEnable* input is set to TRUE.
	FALSE	No positioning/positioning is aborted. • The speed at the *nNOut_v* output is braked to standstill via the deceleration ramp set in C01060/1 (*nNOut_v* = 0). • At the same time, the *dnPosOut_p* output is stopped.
	FALSE↗TRUE	Positioning is executed/continued. • In case of an activated limit stop (C01058/1 = TRUE), a renewed positive edge for a following positioning procedure is required.
bSetPos0 (BOOL)		Travel to zero position • Only possible if the *bEnable* input is set to TRUE.
	FALSE	The value at the *dnSet_p* input is used as setpoint position.
	TRUE	The value at the *dnSet_p* input is ignored. The setpoint position is internally set to "0" and the *dnPosOut_p* output can be travelled to zero position.

Designator Data type	Information/possible settings	
bPosMode BOOL	Positioning mode	
	FALSE	Absolute positioning • The setpoint at *dnSet_p* is the absolute setpoint position (with regard to zero position): 
	TRUE	Relative positioning • With a FALSE/TRUE edge at the *bExecute* input, the outputs of the current position (*dnAct_p*) are traversed by the *dnSet_p* value:  **Note:** Only to be used with activated limit stop (C01058/1 = TRUE)!
dnSet_p DINT	Setpoint for positioning in [increments] • Scaling: A revolution is displayed with 65536 increments or steps.	
dnAct_p DINT	Actual position in [increments] • Connect this input with e.g. *dnPosOut_p* of this FB or with another FB which outputs the actual position of the drive.	
bEnable BOOL	Activate/deactivate positioning function • This input has the highest priority.	
	FALSE	Positioning function deactivated. • For behaviour see C01059/1.
	TRUE	Positioning function activated.

outputs

Designator Data type	Value/meaning	
bInTarget BOOL	Status signal "Target position reached"	
	TRUE	Target position reached.
nNOut_v INT	Speed output of the profile generator • Scaling: 16384 ≡ 15000 rpm	
dnPosOut_p DINT	Output position of the profile generator in [increments] • Scaling: A revolution is displayed with 65536 increments or steps.	

Parameters

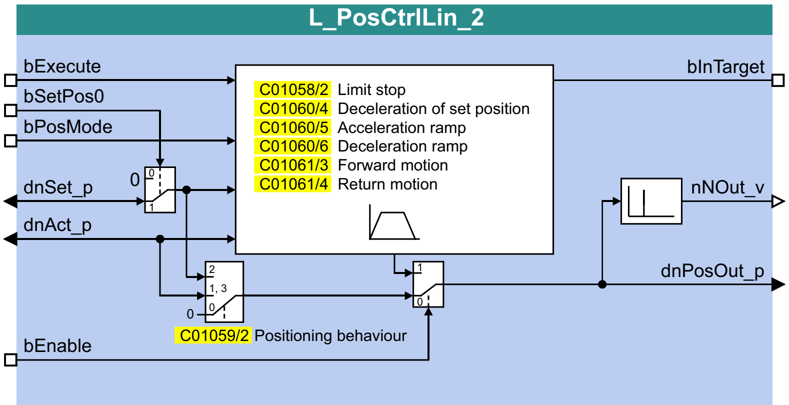
Parameters	Possible settings		Information
C01058/1			Limit stop
	0	deactivated (Lenze setting)	The outputs continuously follow the setpoint at *dnSet_p*.
	1	activated	The outputs follow the setpoint at *dnSet_p* once. • If the setpoint position changes, a new FALSE/TRUE edge is required at the *bExecute* input for another positioning process.

Parameters	Possible settings			Information
C01059/1				Positioning behaviour • Behaviour with deactivated positioning function (*bEnable* = FALSE).
	0	*dnOut_p* = 0 (Lenze setting)		The *dnPosOut_p* output jumps to "0". • The speed signal at the *nNOut_v* output maps this position step change as angle signal. **Note:** If the *nNOut_v* output has an impact on the setpoint generation for the drive, undefined motions (shocks, following errors) may occur.
	1	*dnOut_p/nNOut_v* follow *dnAct_p*		The outputs *dnPosOut_p* and *nNOut_v* follow the value at the *dnAct_p* input (actual position).
	2	*dnOut_p/nNOut_v* follow *dnSet_p*		The outputs *dnPosOut_p* and *nNOut_v* follow the value at the *dnSet_p* input (setpoint).
	3	*dnOut_p/nNOut_v* follow *dnAct_p* (without limitation)		The outputs *dnPosOut_p* and *nNOut_v* follow the value at the *dnAct_p* input (actual position), but without limitation.
C01060/1	0.010	s	130.000	Deceleration of set position • Deceleration ramp for stop before reaching the setpoint position (*bExecute* = FALSE). • Lenze setting: 1.000 s
C01060/2	0.010	s	130.000	Acceleration ramp • Lenze setting: 1.000 s
C01060/3	0.010	s	130.000	Deceleration ramp • Lenze setting: 1.000 s
C01061/1	-15000	rpm	15000	Forward motion • Positive speed • Lenze setting: 200 rpm
C01061/2	-15000	rpm	15000	Return motion • Negative speed • Lenze setting: 200 rpm

19.1.147 L_PosCtrlLin_2

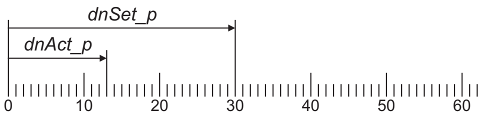
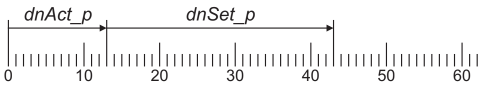
This FB serves to implement the following positioning functions:

- Bring drive to curve position (e.g. after mains connection, manual jog, homing)
- Release drive from curve and bring it to safe position (position override function)
- Positioning the X axis via X offset (higher-level positioning)
- Positioning the Y axis via Y offset (higher-level positioning)



inputs

Designator Data type	Information/possible settings	
bExecute BOOL	Execute positioning profile • Only possible if the *bEnable* input is set to TRUE.	
	FALSE	No positioning/positioning is aborted. • The speed at the *nNOut_v* output is braked to standstill via the deceleration ramp set in C01060/4 (*nNOut_v* = 0). • At the same time, the *dnPosOut_p* output is stopped.
	FALSE↗TRUE	Positioning is executed/continued. • In case of an activated limit stop (C01058/2 = TRUE), a renewed positive edge for a following positioning procedure is required.
bSetPos0 BOOL	Travel to zero position • Only possible if the *bEnable* input is set to TRUE.	
	FALSE	The value at the *dnSet_p* input is used as setpoint position.
	TRUE	The value at the *dnSet_p* input is ignored. The setpoint position is internally set to "0" and the *dnPosOut_p* output can be travelled to zero position.

Designator Data type	Information/possible settings	
bPosMode BOOL	Positioning mode	
	FALSE	Absolute positioning • The setpoint at *dnSet_p* is the absolute setpoint position (with regard to zero position): 
	TRUE	Relative positioning • With a FALSE/TRUE edge at the *bExecute* input, the outputs of the current position (*dnAct_p*) are traversed by the *dnSet_p* value:  **Note:** Only to be used with activated limit stop (C01058/2 = TRUE)!
dnSet_p DINT	Setpoint for positioning in [increments] • Scaling: A revolution is displayed with 65536 increments or steps.	
dnAct_p DINT	Actual position in [increments] • Connect this input with e.g. *dnPosOut_p* of this FB or with another FB which outputs the actual position of the drive.	
bEnable BOOL	Activate/deactivate positioning function • This input has the highest priority.	
	FALSE	Positioning function deactivated. • For behaviour see C01059/2.
	TRUE	Positioning function activated.

outputs

Designator Data type	Value/meaning	
bInTarget BOOL	Status signal "Target position reached"	
	TRUE	Target position reached.
nNOut_v INT	Speed output of the profile generator • Scaling: 16384 ≡ 15000 rpm	
dnPosOut_p DINT	Output position of the profile generator in [increments] • Scaling: A revolution is displayed with 65536 increments or steps.	

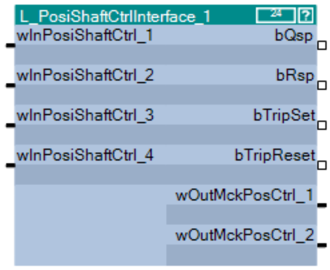
Parameters

Parameters	Possible settings		Information
C01058/2			Limit stop
	0	deactivated (Lenze setting)	The outputs continuously follow the setpoint at *dnSet_p*.
	1	activated	The outputs follow the setpoint at *dnSet_p* once. • If the setpoint position changes, a new FALSE/TRUE edge is required at the *bExecute* input for another positioning process.

Parameters	Possible settings			Information
C01059/2				Positioning behaviour • Behaviour with deactivated positioning function (*bEnable* = FALSE).
	0	*dnOut_p* = 0 (Lenze setting)		The *dnPosOut_p* output jumps to "0". • The speed signal at the *nNOut_v* output maps this position step change as angle signal. **Note:** If the *nNOut_v* output has an impact on the setpoint generation for the drive, undefined motions (shocks, following errors) may occur.
	1	*dnOut_p/nNOut_v* follow *dnAct_p*		The outputs *dnPosOut_p* and *nNOut_v* follow the value at the *dnAct_p* input (actual position).
	2	*dnOut_p/nNOut_v* follow *dnSet_p*		The outputs *dnPosOut_p* and *nNOut_v* follow the value at the *dnSet_p* input (setpoint).
	3	*dnOut_p/nNOut_v* follow *dnAct_p* (without limitation)		The outputs *dnPosOut_p* and *nNOut_v* follow the value at the *dnAct_p* input (actual position), but without limitation.
C01060/4	0.010	s	130.000	Deceleration of set position • Deceleration ramp for stop before reaching the setpoint position (*bExecute* = FALSE). • Lenze setting: 1.000 s
C01060/5	0.010	s	130.000	Acceleration ramp • Lenze setting: 1.000 s
C01060/6	0.010	s	130.000	Deceleration ramp • Lenze setting: 1.000 s
C01061/3	-15000	rpm	15000	Forward motion • Positive speed • Lenze setting: 200 rpm
C01061/4	-15000	rpm	15000	Return motion • Negative speed • Lenze setting: 200 rpm

19.1.148 L_PosiShaftCtrlInterface_1

FB in preparation!

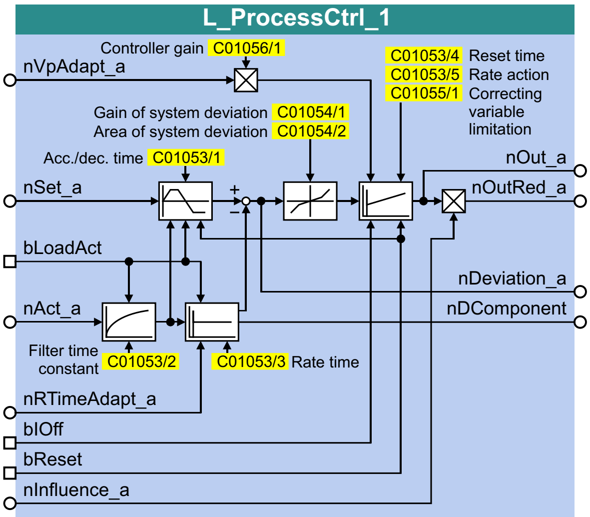


19.1.149 L_ProcessCtrl_1

This FB serves to implement a dancer position or tension control.

The FB is provided with the following functions:

- Adjustable control algorithm (P, PI, PID) with adaptable gain
- Reduced controller dynamics at low system deviation
- Setpoint ramp generator for preventing setpoint step-changes at the input
- Setpoint ramp generator can be loaded with actual value
- Low-pass filter and rate action in the actual value feedback
- Integral action component can be switched off
- Interruptible control



inputs

Designator Data type	Information/possible settings	
nVpAdapt_a INT	Proportional evaluation of the controller gain (Vp) • Scaling: 16384 ≡ 100 % • Internal limitation to 0 ... 16384 (0 ... 100 %)	
nSet_a INT	Controller setpoint	
bLoadAct BOOL	Accept actual controller value	
	TRUE	The actual controller value *nAct_a* is taken over into the ramp generator, the low pass and the rate action.
nAct_a INT	Actual controller value	
nRTimeAdapt_a INT	Proportional evaluation of the rate time in the actual value path • Scaling: 16384 ≡ 100 % rate time (C01053) • Internal limitation to 0 ... 16384 (0 ... 100 %)	
bIOff BOOL	Reset controller I component	
	TRUE	The controller I component is reset.

Designator Data type	Information/possible settings	
bReset BOOL	Reset entire control	
	TRUE	All outputs are reset to 0.
nInfluence_a INT	Proportional evaluation of the controller correcting variable *nOutRed_a* • Scaling: 16384 ≡ 100 % controller correcting variable (*nOut_a*)	

outputs

Designator Data type	Value/meaning
nOut_a INT	Controller correcting variable • Internal limitation to ±16384 (±100 %)
nOutRed_a INT	Controller correcting variable (evaluated by *nInfluence_a*)
nDeviation_a INT	System deviation • Internal limitation to ±32767 (±199.99 %)
nDComponent INT	D component of the rate action • Internal limitation to ±16384 (±100 %)

Parameters

Parameters	Possible settings			Information
C01053/1	0.000	s	30.000	Acceleration/deceleration time of the setpoint ramp generator • Lenze setting: 0.000 s
C01053/2	0.000	s	30.000	Filter time constant for actual controller value • Lenze setting: 0.000 s
C01053/3	0.000	s	30.000	Rate time for actual controller value • Lenze setting: 0.000 s
C01053/4	0.000	s	30.000	Controller reset time • Lenze setting: 1.000 s
C01053/5	0.000	s	30.000	Controller rate action • Lenze setting: 0.000 s
C01054/1	0.00	%	199.99	Gain of the system deviation in the range of reduced sensitivity • Lenze setting: 100.00 %
C01054/2	0.00	%	199.99	Range of system deviation with reduced gain/sensitivity • Lenze setting: 0.00 %
C01055/1				Limitation of the controller correcting variable to the positive area
	False	Limitation not active. (Lenze setting)		
	True	Limitation active.		
C01056/1	0.00		100.00	Controller gain • Lenze setting: 0.10

19.1.149.1 Control characteristic

The dancer position or tension controller can be optionally operated as P, PI or PID controller. In the Lenze setting, the PI algorithm is active.

Gain (P component)

The input value is controlled by a linear characteristic. The slope of the characteristic is determined by the controller gain Vp.

The controller gain Vp is set in C01056/1.

- The controller gain can be adapted via the *nVpAdapt_a* input (also possible in online mode).
- The *nVpAdapt_a* input value has a direct effect on the controller gain:

$$P = \text{nVpAdapt\_a} \cdot \text{C01056/1}$$

Example: With the parameterised controller gain Vp = 2.0 and *nVpAdapt_a* = 75 %, the resulting gain factor is as follows:

$$P = \frac{75\,[\%]}{100\,[\%]} \cdot 2.0 = 1.5$$

Reset time Tn (I component)

The adjustment time Tn is set under C01053/4.

- The I component of the controller can be deactivated by setting the input *bIOff* to TRUE.
- The I component can be switched on and off online.

Differential component Kd (D component)

The differential component Kd is set under C01053/5.

- The setting "0.0 s" deactivates the D component (Lenze setting). In this way, the PID controller becomes a PI controller or P controller, if the I component has been deactivated as well.

Evaluation of the output signal (controller influence)

If the motor speed or motor torque is precontrolled, a low influence is sufficient for the controller to comply with the setpoint.

Use the *nInfluence_a* input to select the influencing factor the controller correcting variable (*nOut_a*) is to be evaluated with. The evaluated controller correcting variable is output at *nOutRed_a*.

Note!

The controller influence evaluates the output signal multiplicatively. A change of the *nInfluence_a* influencing factor also changes the dynamics of the controller!

Switching on/off the controller

By setting the *bReset* input to TRUE, the process controller can be switched off.

Loading the setpoint ramp generator with the actual value

If the actual value is loaded into the setpoint ramp generator in switched-off state, it has the advantage that in the moment of the renewed controller enable, the system deviation is zero first. Thus, compensation processes can be mostly prevented.

When the *bLoadAct* input is set to TRUE, the setpoint ramp generator can be loaded with the actual value. This keeps the system deviation equal to zero and I component of the controller.

Low pass and rate action in the actual value path

In order to filter signal interferences more effectively, you can activate a low pass in the actual value path.

- The filter time constant for the low pass is set in C01053/2.
- The rate time constant in the feedback path can be set in C01053/3. This serves to compensate interfering decelerations.

Reduced controller dynamics at low system deviation

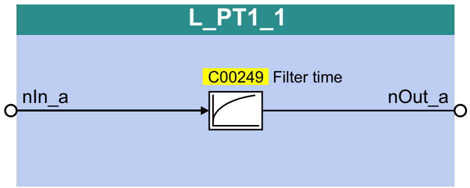
A reduced controller dynamics at low system deviations serves to have a positive effect on the damping behaviour of the control loop.

- C01054/2 serves to determine the tolerance zone in which the system deviation is transmitted to the controller with a slight gain.
- C01054/1 serves to determine the amount of percent the gain in the defined tolerance zone is to be reduced to.

19.1.150 L_PT1_1

This FB filters and delays analog signals.

- The filter time constant T can be set under C00249.
- The gain is defined with $V_p = 1$.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

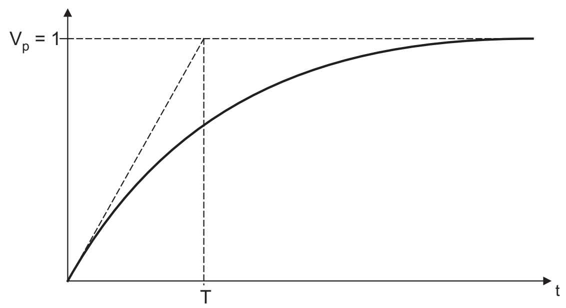
outputs

Designator Data type	Value/meaning
nOut INT	Output signal

Parameters

Parameters	Possible settings			Information
C00249	0	ms	5000	Filter time constant • The filter is not active with a setting of "0 ms". The input signal is passed through one-to-one to the output. • Lenze setting: 2000 ms

Function

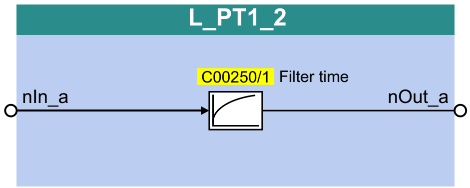


[19-69] Filter time constant T of the first-order delay element

19.1.151 L_PT1_2

This FB filters and delays analog signals.

- Filter time constant T can be set in C00250/1.
- The gain is defined with $V_p = 1$.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

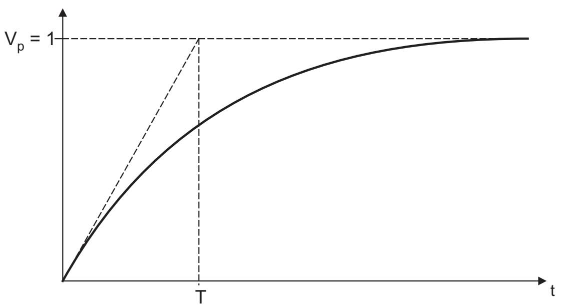
outputs

Designator Data type	Value/meaning
nOut INT	Output signal

Parameters

Parameters	Possible settings			Information
C00250/1	0	ms	5000	Filter time constant • The filter is not active with a setting of "0 ms". The input signal is passed through one-to-one to the output. • Lenze setting: 2000 ms

Function

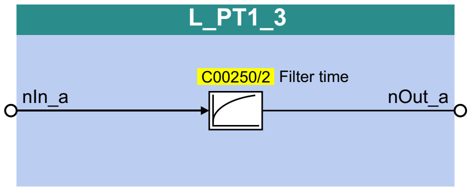


[19-70] Filter time constant T of the first-order delay element

19.1.152 L_PT1_3

This FB filters and delays analog signals.

- Filter time constant T can be set in C00250/2.
- The gain is defined with $V_p = 1$.



inputs

Designator Data type	Information/possible settings
nIn_a INT	Input signal

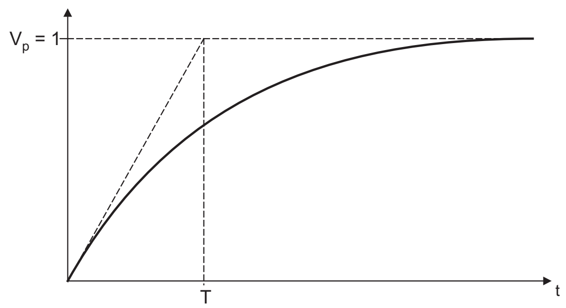
outputs

Designator Data type	Value/meaning
nOut INT	Output signal

Parameters

Parameters	Possible settings			Information
C00250/2	0	ms	5000	Filter time constant • The filter is not active with a setting of "0 ms". The input signal is passed through one-to-one to the output. • Lenze setting: 2000 ms

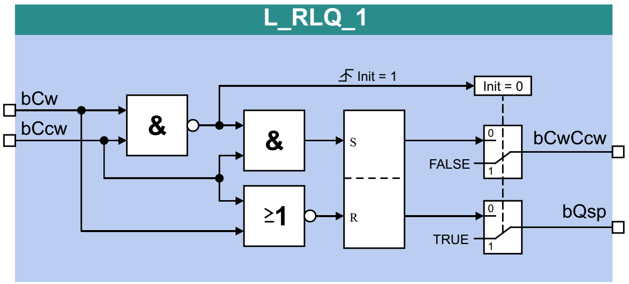
Function



[19-71] Filter time constant T of the first-order delay element

19.1.153 L_RLQ_1

This FB links a selected direction of rotation to the quick stop function with wire-break protection.



inputs

Designator Data type	Information/possible settings
bCw BOOL	Input • TRUE = CW rotation
bCCw BOOL	Input • TRUE = CCW rotation

outputs

Designator Data type	Value/meaning
bQSP BOOL	Output signal for quick stop (QSP)
bCwCcw BOOL	Output signal for CW/CCW rotation • TRUE = CCW rotation

Function

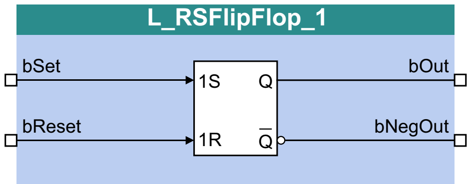
inputs		outputs		Notes
bCw	bCCw	bCwCcw	bQSP	
TRUE	TRUE	FALSE	TRUE	The inputs have this status only if a TRUE signal is being applied to both inputs at the moment of switch-on! See also FB illustration above, "Init" = 1.
If *one* of the inputs has the TRUE status, the following truth table applies:				
FALSE	FALSE	FALSE	TRUE	See also FB illustration above, "Init" = 0.
TRUE	FALSE	FALSE	FALSE	
FALSE	TRUE	TRUE	FALSE	
TRUE	TRUE	X (save)		

[19-72] Truth table of the FB L_RLQ, 0 = FALSE, 1 = TRUE

19.1.154 L_RSFlipFlop_1

The functionality of this FB corresponds to that of an RS flipflop:

- An input signal at *bSet* serves to set the *bOut* output permanently.
- An input signal at *bReset* serves to reset the output.
- Additional, inverted output *bNegOut*



inputs

Designator Data type	Information/possible settings	
bSet BOOL	Setting input	
	TRUE	• The *bOut* output is set to TRUE. • The *bNegOut* output is set to FALSE.
bReset BOOL	Reset input	
	TRUE	• The *bOut* output is set to FALSE. • The *bNegOut* output is set to TRUE.

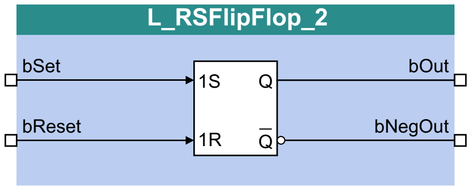
outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal
bNegOut BOOL	Output signal, inverted

19.1.155 L_RSFlipFlop_2

The functionality of this FB corresponds to that of an RS flipflop:

- An input signal at *bSet* serves to set the *bOut* output permanently.
- An input signal at *bReset* serves to reset the output.
- Additional, inverted output *bNegOut*



inputs

Designator Data type	Information/possible settings	
bSet BOOL	Setting input	
	TRUE	• The *bOut* output is set to TRUE. • The *bNegOut* output is set to FALSE.
bReset BOOL	Reset input	
	TRUE	• The *bOut* output is set to FALSE. • The *bNegOut* output is set to TRUE.

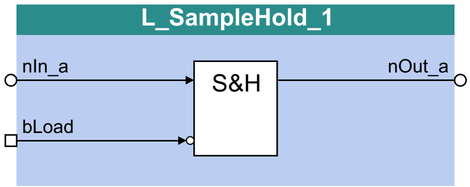
outputs

Designator Data type	Value/meaning
bOut BOOL	Output signal
bNegOut BOOL	Output signal, inverted

19.1.156 L_SampleHold_1

This FB can save a signal.

- The saved value is also available after mains switching.



inputs

Designator Data type	Information/possible settings	
nIn_a INT	Input signal	
bLoad BOOL	Save input signal	
	FALSE	The last-valid value at *nIn* is saved and output to *nOut*. A signal change at *nIn* does not cause a change at *nOut*.
	TRUE	The *nOut* output provides *dnIn*.

outputs

Designator Data type	Value/meaning
nOut INT	Output signal

Function

- When *bLoad* = TRUE, the *nIn_a* signal is switched to *nOut_a*.
- When *bLoad* = FALSE, the last-valid value is saved and output at *nOut_a*. A signal change at *nIn_a* does not cause a change at *nOut_a*.

Behaviour after mains switching

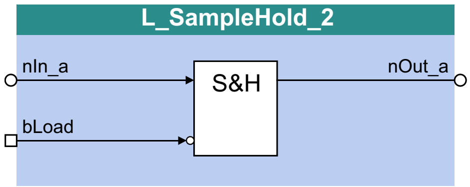
The last-loaded value is permanently stored after switching of the supply voltage and reloaded after restart.

- In order that the saved value is not immediately overwritten with the current input signal at *nIn* after restart, *bLoad* must be set to FALSE at restart.

19.1.157 L_SampleHold_2

This FB can save a signal.

- The saved value is also available after mains switching.



inputs

Designator Data type	Information/possible settings	
nIn_a INT	Input signal	
bLoad BOOL	Save input signal	
	FALSE	The last-valid value at *nIn* is saved and output to *nOut*. A signal change at *nIn* does not cause a change at *nOut*.
	TRUE	The *nOut* output provides *dnIn*.

outputs

Designator Data type	Value/meaning
nOut INT	Output signal

Function

- When *bLoad* = TRUE, the *nIn_a* signal is switched to *nOut_a*.
- When *bLoad* = FALSE, the last-valid value is saved and output at *nOut_a*. A signal change at *nIn_a* does not cause a change at *nOut_a*.

Behaviour after mains switching

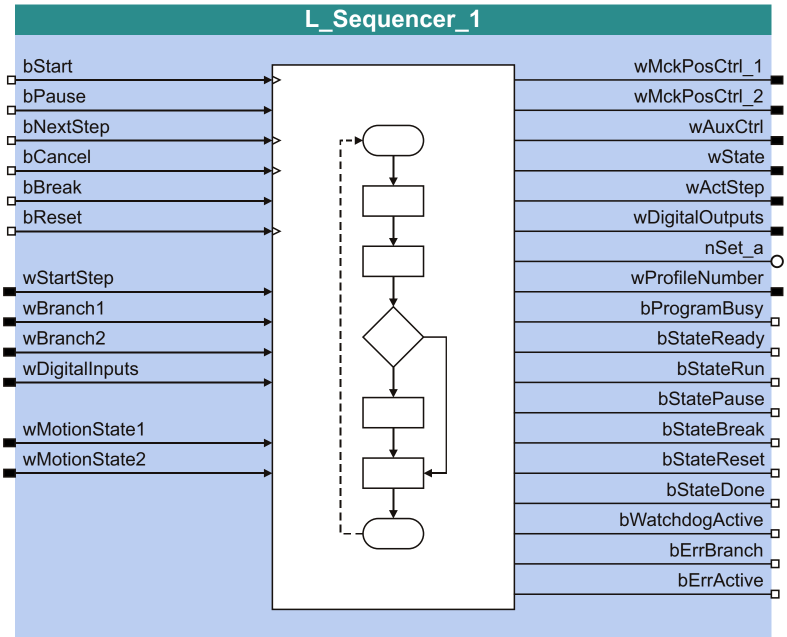
The last-loaded value is permanently stored after switching of the supply voltage and reloaded after restart.

- In order that the saved value is not immediately overwritten with the current input signal at *nIn* after restart, *bLoad* must be set to FALSE at restart.

19.1.158 L_Sequencer_1

This FB processes a positioning program by means of a sequence table.

- The sequence table is a central component of this FB and is represented by a code with 100 subcodes where each subcode references to a program segment which is called "action".
- Different action types are available which serve to realise, for instance, program branching, switching operation, waiting times and counters. The actions themselves are represented by codes as well.
  - The following action types are passive: "Branching", "variable branching", "switching", "setting the counter", "counting", "time waiting" and "program end". When these action types are being processed, the "Stop" operating mode is active and a brake, if available, will be applied.
  - The action types "positioning" and "homing", however, are active as these two action types trigger the execution of the respective basic function via the corresponding control inputs.
  - The "Stand-by" action type is active when a setpoint has been set for the speed follower or a setpoint position is transferred to the SB LS_MotionControlKernel when "position follower" has been selected.



inputs

Designator Data type	Information/possible settings	
bStart BOOL	Start/continue positioning program	
	FALSE↗TRUE	The positioning program is started/continued. • Counters and outputs are not reset automatically through this. • A positioning program previously interrupted by *bBreak* is continued with the sequence step. • The started positioning program is processed up to its program end even if *bStart* is reset to FALSE again.

Designator Data type		Information/possible settings
bPause BOOL		Interrupt the current step of the positioning program

	TRUE	• An active positioning is interrupted.
		  • The drive is stopped with the current profile deceleration.
		  • A deceleration override is considered.
		• The sequence of an active timing element is stopped.
		• An action of "Standby" type is not interrupted.

	FALSE	• The interrupted positioning is completed.
		  • The distance of the profile already covered is considered here.
		  • Positionings with final speed are not continued when the final speed cannot be reached anymore.
		• The sequence of the stopped timing element is continued with the remaining time.

bNextStep BOOL		Cancel the current step and go to the next step
	FALSE↗TRUE	The current step of the positioning program is cancelled immediately and the positioning program is continued with the next step in the sequence table.
bCancel BOOL		Cancel the current step and go to the step defined in C01404
	FALSE↗TRUE	The current step of the positioning program is cancelled immediately and the positioning program is continued wit the step defined in C01404.
bBreak BOOL		Interrupt positioning program • This input is intended for being connected to drive-relevant status signals (e.g. for controller inhibit or quick stop). • If the interruption is deactivated again by a reset to FALSE, a new FALSE/TRUE edge is required at *bStart* in order that the positioning program is continued with the sequence step.
	FALSE	Condition for program start via *bStart*.
	TRUE	The active positioning program is interrupted. • If a positioning is active, the drive is brought to a standstill with the delay time for stop (C01251/1). • The MCK operating modes Speed follower, Homing, Positioning and Position follower are deactivated. • The digital output signals keep their states.
bReset BOOL		Reset positioning program. • Also possible with interrupted positioning program.
	FALSE↗TRUE	The positioning program is reset. • If a positioning is active, the drive is brought to a standstill with the delay time for stop (C01251/1) without considering an acceleration override. • The program flow is cancelled ("program end"). • The digital output signals, counters and timing elements are reset. • A possibly active action of the "Standby" type is aborted.
wStartStep WORD		Step of the sequence table the positioning program is to start with.
	0	Positioning program starts with step 1 of the sequence table.
	1 ... 100	First step
	Others	Positioning program does not start.
	Not assigned	Positioning program starts with step 1 of the sequence table.

Designator Data type	Information/possible settings	
wBranch1 WORD	Input signal for action 1 of "Variable branching" type • When the positioning program contains an action 1 of "Variable branching" type, branching is carried out in the corresponding step depending on the value available at this input:	
	0	Branching to the next step.
	1	Branching to the step defined in C01418/1.
	2	Branching to the step defined in C01419/1.
	...	...
	20	Branching to the step defined in C01437/1.
	>20	Branching to the next step.
wBranch2 WORD	Input signal for action 2 of "Variable branching" type • When the positioning program contains an action 2 of "Variable branching" type, branching is carried out in the corresponding step depending on the value available at this input:	
	0	Branching to the next step.
	1	Branching to the step defined in C01418/2.
	2	Branching to the step defined in C01419/2.
	...	...
	20	Branching to the step defined in C01437/2.
	>20	Branching to the next step.
wDigitalInputs WORD	Sequencer inputs 1 ... 16 (bit coded) • The action types "positioning", "branching", "waiting" and "stand-by" have the "Input for..." parameter. If this is non-zero, it names the number of the sequencer input where the positioning program expects the respectively defined level before action is executed. • For the transfer of individual control signals, e.g. the converter FB L_ConvBitsToWord_1 can be connected to this input.	
	Bit 0	Sequencer input 1
	Bit 1	Sequencer input 2
	...	...
	Bit 15	Sequencer input 16
wMotionState1 wMotionState2 WORD	Interface to the basic drive functions (MCK) • Connect these inputs to the outputs with the same name of the SB LS_MotionControlKernel. ▶ Connection to the basic drive functions (MCK)	

outputs

Designator Data type	Value/meaning
wMckPosCtrl1 wMckPosCtrl2 wAuxCtrl WORD	Interface to the basic drive functions (MCK) • Connect the outputs *wMckPosCtrl1* and *wMckPosCtrl2* to the inputs *wInMckPosCtrl1* and *wInMckPosCtrl2* des FB L_MckCtrlInterface_1. • Connect the *wAucCtrl* output to the input with the same name of the SB LS_MotionControlKernel. ▶ Connection to the basic drive functions (MCK)

Designator Data type	Value/meaning	
wState WORD	Status (bit coded)	
	Bit 1	Positioning program ready to start/program end reached.
	Bit 2	Positioning program is running.
	Bit 3	Positioning program started, break active.
	Bit 4	Positioning program stopped.
	Bit 5	Positioning program reset.
	Bit 6	Positioning program completed.
	Bit 7	Time monitoring for "positioning" action has been triggered.
	Bit 8	Wrong value at branch input.
	Bit 15	Error active.
wActStep WORD	Current step being processed by the positioning program.	
	0	Positioning program not started.
	1 ... 100	Current step
wDigitalOutputs WORD	Sequencer outputs 1 ... 16 (bit coded) • The sequencer outputs can be set to "0" or "1" by the "switching" action type during the runtime of the positioning program. • For providing individual control signals, e.g. the converter FB L_ConvWordToBits_1 can be connected to this output.	
	Bit 0	Sequencer output 1
	Bit 1	Sequencer output 2
	...	...
	Bit 15	Sequencer output 16
nSet_a INT	Speed setpoint which is output in the "Stand-by" action. • If no "Stand-by" action is active, the value "0" is output.	
wProfileNumber WORD	Display of the profile data set active in the current step	
bProgramBusy BOOL	Status signal "Positioning program is running"	
	FALSE	Positioning program not started or stopped.
	TRUE	Positioning program is running.
bStateReady BOOL	TRUE	Positioning program ready to start/program end reached.
bStateRun BOOL	TRUE	Positioning program is running.
bStatePause BOOL	TRUE	Positioning program started, break active.
bStateBreak BOOL	TRUE	Positioning program stopped.
bStateReset BOOL	TRUE	Positioning program reset.
bStateDone BOOL	TRUE	Positioning program completed.
bWatchdogActive BOOL	TRUE	Time monitoring for "positioning" action has been triggered.
bErrBranch BOOL	TRUE	Wrong value at branch input.
bErrActive BOOL	TRUE	Error active.

General parameters

Note!

The parameters of the different actions for the sequence table are described in the corresponding action subchapter.

Parameters	Possible settings			Information
C01400/1...100				Action in sequence step n • In the subcodes, the calls of the actions required for the positioning program are stored. In this way, the basic sequence (with the exception of the branches) is defined.
	0	Program end		
	xxyyy	Action call • xx ≡ Action type • yyy ≡ Action number		
C01401	1		100	Current step • Read only
C01402				Sequence control: Status • Read only
	0	READY: Positioning program ready to start/program end reached.		
	1	RUN: Positioning program is running.		
	2	PAUSE: Current step interrupted, break is active.		
	3	BREAK: Positioning program interrupted.		
	4	RESET: Positioning program reset.		
C01403				Sequence control: Control word • Bit coded code for controlling the sequencer by a PC or a master control via parameter channel. • By setting a bit to "1", the respective function is activated.
	Bit 0	Start		
	Bit 1	Break		
	Bit 2	Break		
	Bit 3	Cancel		
	Bit 4	Reset		
	Bit 5	NextStep		
C01404				Step takes place when *bCancel* = TRUE • This step is processed if the *bCancel* input is set to TRUE.
	0	**Sequence step**		
	1...100	Step 1 ... 100		
	101	Program end		

19.1.158.1 Connection to the basic drive functions (MCK)

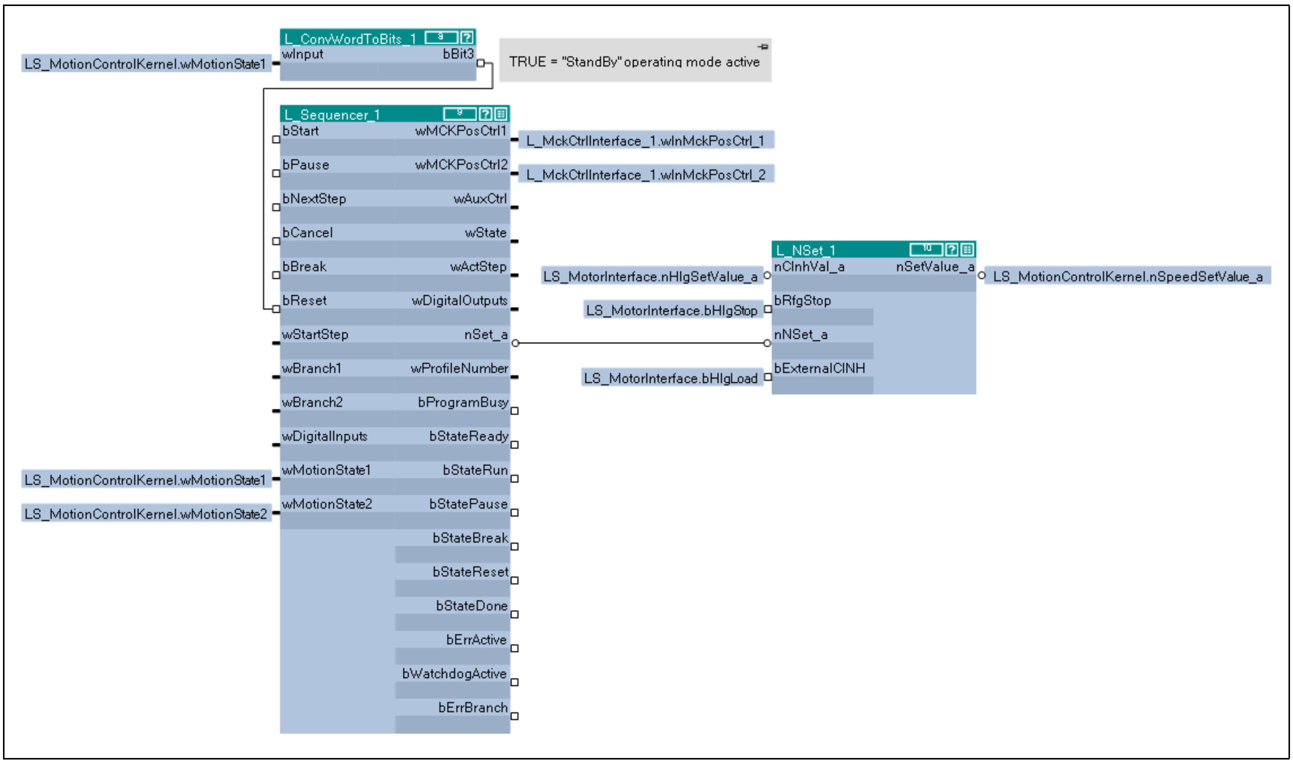
Required connections in the FB Editor

'The recommended starting basis for a connection with the FB **L_Sequencer_1** is the use of the TA "Table positioning".

The following configuration shows the required connections in order that the execution of the prevailing basic function is triggered in LS_MotionControlKernel for the two action types "Positioning" and "Referencing".

With a corresponding parameter setting, the *nSet_a* output serves to transfer a speed setpoint to the basic function "speed follower" if an action of Standby" type is active.

The process is reset in the MCK "Standby" operating mode. The internal "StandBy" operating mode is assumed implicitly if controller inhibit, pulse inhibit, quick stop and/or DC-injection braking are activated. ▶ "StandBy" operating mode (📖 596)



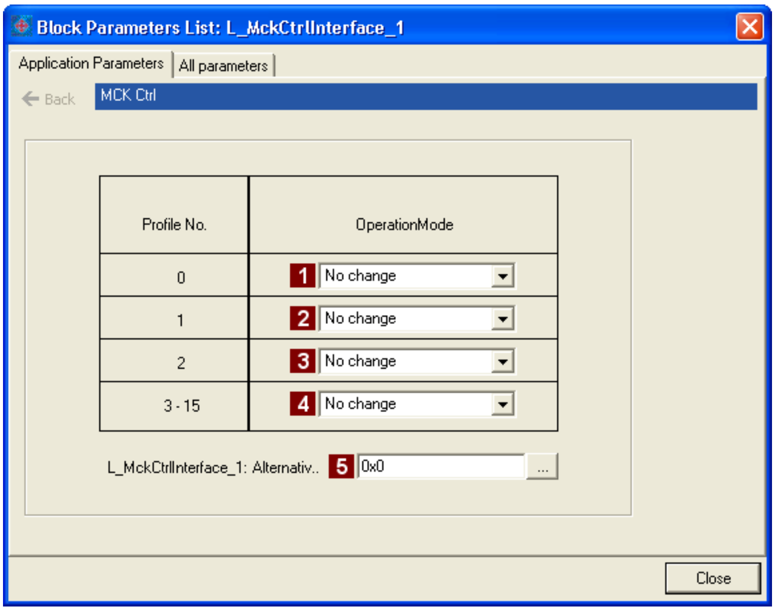
[19-73] Basic integration of the sequence control into the application

Note!

The FB **L_Sequencer_1** has to be called in the processing order before the FB L_MckCtrlInterface_1!

Required parameter settings

For an error-free function of the sequence control, the following parameter settings are required for the FB L_MckCtrlInterface_1:



	Parameters	Information
1	**MCK operating mode at profile no. 0** (C01298/1)	Selection of the operating mode for profile 0 ... 15 at L_MckCtrlInterface_1. **For these four parameters, set the selection "0: No change"** (no assignment of an operating mode to a profile).
2	**MCK operating mode at profile no. 1** (C01298/2)	
3	**MCK operating mode at profile no. 2** (C01298/3)	
4	**MCK operating mode at profile no. 3...15** (C01298/4)	
5	**L_MckCtrlInterface_1: Alternative function** (C01297)	Bit coded selection of the Alternative functions for the bit 16 (PosExecute) in the MCK control word. **For this parameter, set the hexadecimal value "0x0" to deactivate all alternative functions.**

19.1.158.2 Priorities of the control signals

The boolean control signals have the following priorities (1 = highest priority):

1. *bReset* 2. *bBreak* 3. *bPause* 4. *bStart* 5. *bCancel*

19.1.158.3 States of the sequencer

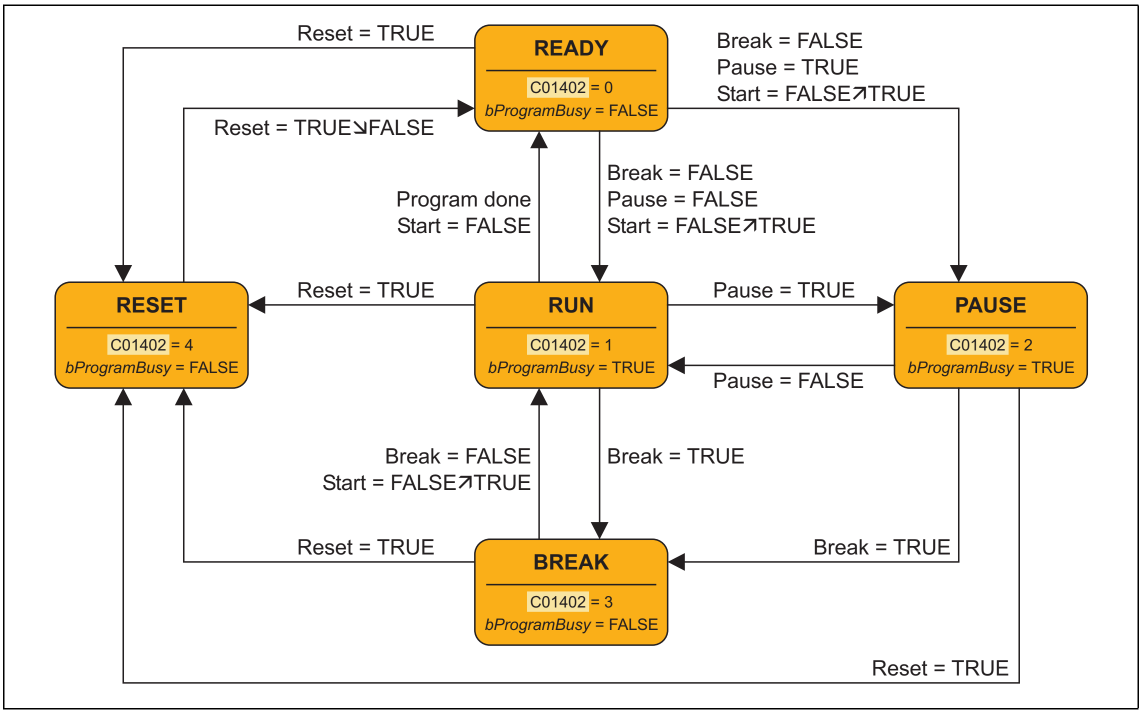
The internal state machine of the sequencer distinguishes the following five states:

State	Display C01402	Impact/meaning
READY	0	When there is no interruption by *bBreak*, the program can be started.
RUN	1	The program is processed.

Break	2	The program is paused and the traversing of a profile is stopped.
		• The outputs keep their states
		• An active positioning is interrupted.
		  • The drive is stopped with the current profile deceleration.
		  • A acceleration override is not considered.
		• After deactivating the break, the positioning program and a previously active positioning are continued at the same position.

Break	3	The traversing of a profile is cancelled. • The drive with the delay time for stop (C01251/1) is brought to a standstill. • The digital output signals keep their states. • After the interruption has been deactivated (*bBreak* = FALSE), another start signal is required to continue the program with the sequence step.

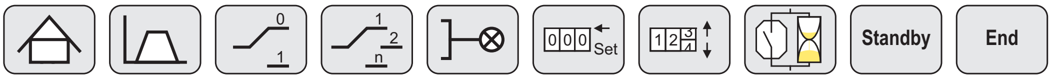
RESET	4	The program is reset.
		• An active positioning is cancelled.
		  • The drive with the delay time for stop (C01251/1) is brought to a standstill.
		  • A acceleration override is not considered.
		• The program flow is cancelled ("program end").
		• The digital output signals and the counter are reset.
		• If a standby operation is active, it will be cancelled.



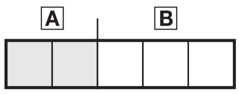
[19-74] Internal state machine of the sequencer

19.1.158.4 Overview: Available action types

The program flow of the sequential positioning control is defined by means of a sequence table which can contain up to 100 references to "actions".



- An action comprises a clear functionality which is described with a few parameters.
- Different action types are available which serve to realise, for instance, program branching, switching operation, waiting times and counters.
- A certain number of actions are available from every action type which can be parameterised individually.
- An action can be called from several positions in the sequence table.
- After an action has been processed, the action entered in the sequence step of the sequence table is automatically processed unless a branching causes a jump to another step in the sequence table.
- Maximally one action can be processed per calculation cycle.
- The sequence table and the actions themselves are represented by parameters (codes with subcodes).
- Action calls are entered in the sequence table as a decimal number that contains the action type A and the action number B:



Overview

Action	Action type A	Action number B	Number of available actions	Parameter range
Program end	0	000	1	-
Positioning	1	001 ... 050	50	C01405 ... C01410
Switch	2	001 ... 016	16	C01411 ... C01414
Branch	3	001 ... 016	16	C01415 ... C01417
Variable branch	4	001 ... 002	2	C01418 ... C01437
Homing	5	000	1	-
Wait	7	001 ... 008	8	C01438 ... C01440
Set counter	8	001 ... 005	5	C01441 ... C01442
Count	9	001 ... 008	8	C01443 ... C01448
Standby	10	001 ... 005	5	C01449 ... C01452

You can find detailed some information with regard to the action types in the following subchapters.

19.1.158.5 Action type "Program end"

End

In order to define the program end within the sequence table, an action of "program end" type is available.

Action call (entry in the sequence table)

0 (Leading zeros can be omitted.)

- When the *bStart* input is reset to FALSE while the positioning program is running, processing is only continued until program end is reached.
- In the Lenze setting, all entries of the sequence table are set to "0" and thus to "program end".

19.1.158.6 Action type "positioning"

For starting a positioning, 50 actions of "positioning" type are available. The Profile entry is made via the parameters of the basic function "Positioning".

Action call (entry in the sequence table)

1	x	x	x

with action number xxx = 001 ... 050

Parameters

The available actions 1 ... 50 are represented via the subcodes 1 ... 50.

Parameters	Possible settings			Information
C01405/1...50				Pos. action n: Starting with
	0	**Waiting function deactivated**		In the default setting, execution of the profile is started immediately.
	1	Sequencer input 1 (Bit 0 of *wDigitalInputs*)		"Wait for level": The execution of the profile is only started when the selected sequencer input has the polarity set in C01406/x.
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)		
	...	...		
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)		
C01406/1...50				Pos. action n: Polarity input • State which the sequencer input selected for the profile start must have.
	0	**Condition is bit state "0"**		
	1	Condition is bit state "1"		
C01407/1...50				Pos. action n: Profile number • A sequence profile can be set in the corresponding profile parameter.
	0	**No profile executed**		
	1...15	Execute profile no. 1 ... 15		
C01408/1...50				Pos. action n: Jump destination • Step within the sequence table which is processed after the profile has been executed.
	0	**Sequence step**		
	1...100	Step 1 ... 100		
C01409/1...50	0.000	s	2147480.000	Pos. action n: Monitoring time • If the positioning process lasts longer than the monitoring time set here, it is branched to the jump destination set in C01410/x.
	When "0.000 s" is set (Lenze setting), the time monitoring function is deactivated.			
C01410/1...50				Pos. action n: Jump destination monitoring • Step which is executed after the monitoring time has been exceeded.
	0	**Sequence step**		
	1...100	Step 1 ... 100		

19.1.158.7 Action type "switching"

For switching digital output signals, 16 actions of the "Switch" type are provided. Each action can set two bits of the *wDigitalOutputs* output signal, which can be selected, to "0" or "1", alternatively and independently of each other.

Action call (entry in the sequence table)

2	x	x	x

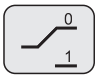
with action number xxx = 001 ... 016

Parameters

The available actions 1 ... 16 are represented via the subcodes 1 ... 16.

Parameters	Possible settings		Information
C01411/1...16			Switching act. n: Output switching A • Selection of the sequencer output which is to be set to the polarity set in C01412/x.
	0	**Deactivated**	
	1	Sequencer output 1 (Bit 0 of *wDigitalOutputs*)	
	2	Sequencer output 2 (Bit 1 of *wDigitalOutputs*)	
	...	...	
	16	Sequencer output 16 (Bit 15 of *wDigitalOutputs*)	
C01412/1...16			Switching act. n: Pol. switching A • State to which the sequencer output selected in C01411/x is to be set.
	0	**Set output bit to "0"**	
	1	Set output bit to "1"	
C01413/1...16			Switching act. n: Output switching B • Selection of the sequencer output which is to be set to the polarity set in C01414/x.
	0	**Deactivated**	
	1	Sequencer output 1 (Bit 0 of *wDigitalOutputs*)	
	2	Sequencer output 2 (Bit 1 of *wDigitalOutputs*)	
	...	...	
	16	Sequencer output 16 (Bit 15 of *wDigitalOutputs*)	
C01414/1...16			Switching act. n: Pol. switching B • State to which the sequencer output selected in C01413/x is to be set.
	0	**Set output bit to "0"**	
	1	Set output bit to "1"	

19.1.158.8 Action type "Branching"



For conditional and unconditional branching (jumps), 16 actions of "Branching" Type are available.

Action call (entry in the sequence table)

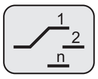
| 3 | x | x | x | with action number xxx = 001 ... 016

Parameters

The available actions 1 ... 16 are represented via the subcodes 1 ... 16.

Parameters	Possible settings		Information
C01415/1...16			Branching action n: Input for jump
	0	**Unconditional branch**	"Unconditional branch": It is always branched to the jump destination set in C01417/x.
	1	Sequencer input 1 (Bit 0 of *wDigitalInputs*)	"Conditional branch": It is only branched to the jump destination set in C01417/x if the selected sequencer input has the polarity set in C01416/x. • Otherwise, the following step in the sequence table will be processed.
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)	
	...	...	
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)	
C01416/1...16			Branching action n: Polarity input • State of the selected sequencer input required for a conditional branch to the jump destination.
	0	**Condition is bit state "0"**	
	1	Condition is bit state "1"	
C01417/1...16			Branching act. n: Jump destination • In case of unconditional branch, it is always branched to the jump destination set here. • In case of conditional branch, it is only branched to the jump destination set here if the sequencer input selected in C01415/x has the polarity set in C01416/x. Otherwise, the following step in the sequence table will be processed.
	0	**Sequence step**	
	1...100	Step 1 ... 100	

19.1.158.9 Action type "Variable branching"



For variable branching (jumps), two actions of "variable branching" type are available.

- Branching to one of 20 possible steps is carried out depending on the *wBranch1* input signal or *wBranch2* input signal at the time of processing.
  - *wBranch1* input signal defines branching for action 1.
  - *wBranch2* input signal defines branching for action 2.
- Example: If the *wBranch2* input signal has the value "15" at the time of processing of the action no. 2, it is branched to the step entered in the parameter "Var. branching n: Jump destination 15" for action no. 2 (C01432/2).

Action call (entry in the sequence table)

4	x	x	x

with action number xxx = 001 ... 002

Parameters

The available actions 1 ... 2 are represented via the subcodes 1 ... 2.

Parameters	Possible settings		Information
C01418/1...2 ... C01437/1...2			Var. branch n: Jump destination 1 ... 20 • Branch destinations 1 ... 20 for input values 1 ... 20 of *wBranch1...2*.
	0	**Branching deactivated** (The following step in the sequence table is processed.)	
	1...100	Step 1 ... 100	

19.1.158.10Action type "Homing"

For starting a homing, an action of "homing" type is available.

- The settings for homing (e.g. homing mode) can be made via the parameters of the basic function "Homing".
- After homing is completed (*bHomingDone* = TRUE), the program flow is continued with the next step.

Action call (entry in the sequence table)

5	0	0	0

Parameters

The "homing" action has no own parameters.

19.1.158.11 Action type "Time waiting"

For inserting waiting times into the program flow, 8 actions of "Time waiting" are available. The sequence step will only be processed after a waiting time has elapsed or when a selectable sequencer input has a certain bit state ("0" or "1").

Action call (entry in the sequence table)

7	x	x	x

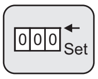
with action number xxx = 001 ... 008

Parameters

The available actions 1 ... 8 are represented via the subcodes 1 ... 8.

Parameters	Possible settings			Information
C01438/1...8	0.000	s	2127480.000	Waiting act. n: Waiting time • Initialisation: 1.000 s
C01439/1...8				Waiting act. n: Input for "Next"
	0	**Input deactivated**		"Wait": Sequence step is only processed after the waiting time set has elapsed.
	1	Sequencer input 1 (Bit 0 of *wDigitalInputs*)		"Wait for level": Sequence step will only be processed when the selected sequencer input has the polarity set in C01440/x. **Note!** However, if a waiting time > 0 s is set, the sequence step is processed at the latest after the waiting time set has elapsed.
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)		
	...	...		
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)		
C01440/1...8				Waiting act. n: Polarity input • State which the sequencer input selected must have so that the sequence step is processed.
	0	**Condition is bit state "0"**		
	1	Condition is bit state "1"		

19.1.158.12Action type "Counter setting"



For setting one of the 5 available counters to a certain starting value, 5 actions of "Counter setting" type are available.

- The 5 actions of the "Counter setting" type are not permanently assigned to the 5 counters.
- You can, e.g., set a counter to a value using an action of "counter setting" type and at a later program time set the same counter to a different value using another action of "counter setting" type.
- The current counter content of the counters 1 ... 5 is displayed in C01443/1...5.

Action call (entry in the sequence table)

| 8 | x | x | x | with action number xxx = 001 ... 005

Parameters

The available actions 1 ... 5 are represented via the subcodes 1 ... 5.

Parameters	Possible settings			Information
C01441/1...5				Counter setting act. n: Counter no.
	0	Counter setting deactivated		
	1	**Counter 1**		
	...	...		
	5	Counter 5		
C01442/1...5	-2147483647		2147483647	Counter setting act. n: Starting value

19.1.158.13 Action type "Counting"

For counting processes, 8 actions of "counting" type are available.

- Every processing of the action causes an increase or decrease of the counter content of the corresponding counter by the set step value (counting up or down).
- If the comparison condition for comparing the counter content to an adjustable comparison value is fulfilled, it can be branched to any step.
- For setting a counter to a starting value, 5 actions of "counter setting" type are available.
- The current counter content of the counters 1 ... 5 is displayed in C01443/1...5.

Action call (entry in the sequence table)

9	x	x	x

with action number xxx = 001 ... 008

Parameters

The available actions 1 ... 8 are represented via the subcodes 1 ... 8.

Parameters	Possible settings			Information
C01443/1...5	-2147483647		2147483647	Counter n: Current counter content • Read only
C01444/1...8				Counting act. n: Counter no.
	0	Counting deactivated (Sequence step is processed.)		
	1	**Counter 1**		
	...	...		
	5	Counter 5		
C01445/1...8	-2147483647		2147483647	Counting act. n: Step value • Value by which the counter is increased or decreased. • Initialisation: 1
C01446/1...8	-2147483647		2147483647	Counting act. n: Comparison value • Value to which the counter is compared.
C01447/1...8				Counting act. n: Jump destination • If the set comparison condition is fulfilled, it is branched to the step set here. • If the condition is not fulfilled or "0" setting, the step followed in the sequence table is processed.
	0	**Sequence step**		
	1...100	Step 1 ... 100		
C01448/1...8				Counting act. n: Comparison operation
	1	**Counter content = comparison value**		
	2	Counter content > comparison value		
	3	Counter content ≥ comparison value		
	4	Counter content < comparison value		
	5	Counter content ≤ comparison value		

19.1.158.14"Stand-by" action type

Standby

For a temporary activation of a setpoint follower, 5 actions of "stand-by" type are available.

- The corresponding setpoint follower remains enabled until the condition for completing stand-by is fulfilled.
- If the "stand-by" step is active, the setpoint set in C01452 is provided at the *nSet_a* output (Lenze setting: 0 %) .

Action call (entry in the sequence table)

1	0	x	x	x

with action number xxx = 001 ... 005

Parameters

The available actions 1 ... 5 are represented via the subcodes 1 ... 5.

Parameters	Possible settings			Information
C01449/1...5				Stand-by act. n: Input for "End" • The setpoint follower selected in C01451/x remains enabled until the sequencer input selected here has the polarity set in C01450/x.
	1	**Sequencer input 1** (Bit 0 of *wDigitalInputs*)		
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)		
	...	...		
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)		
C01450/1...5				Stand-by act. n: Polarity input • State which the sequencer input selected must have so that "Standby" is exited and the sequence step is processed.
	0	**Condition is bit state "0"**		
	1	Condition is bit state "1"		
C01451/1...5				Stand-by act. n: Follower mode • Selection of the follower that is to be activated in standby mode.
	0	Operation Speed follower		
	1	Operation Position follower		
C01452/1...5	-200	%	200	Stand-by act. n: Setpoint • Setpoint for the Speed follower (provided at the output *nSet_a*). • Initialisation: 0 %

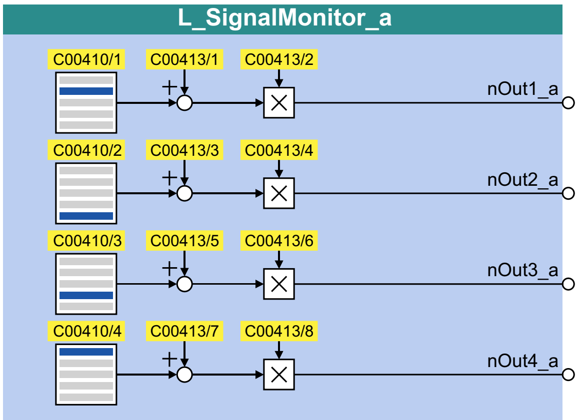
19.1.158.15 Example: Sequence table

Parameters	Entry	≡ Action type/action no.		Meaning
C01400/1	5000		-	Execute homing
C01400/2	3001		1	Process action 1 of "branching" (3) type
C01400/3	9005		5	Process action 5 of "Counting" type (9)
C01400/4	1001		1	Process action 1 of "positioning" type (1)
C01400/5	7003		3	Process action 3 of "Time waiting" type (7)
C01400/6	1002		2	Process action 2 of "positioning" type (1)
C01400/7	3002		2	Process action 2 of type "Branching" (3)
C01400/8	9005		5	Process action 5 of "Counting" type (9)
C01400/9	3004		4	Process action 4 of "branching" (3) type
C01400/...	yyxxx	yy	xxx	Process action *xxx* of *yy* type
C01400/24	1003		3	Process action 3 of "positioning" type (1)
C01400/25	0	End	-	Program end
C01400/26	0	End	-	Program end
C01400/...	...	...	...	...
C01400/100	0	End	-	Program end

19.1.159 L_SignalMonitor_a

This FB outputs four analog signals which can be selected from a list of analog output signals of all function blocks provided in the device.

- Offset and gain of the source signals are adjustable.



outputs

Designator Data type	Value/meaning
nOut1_a INT	Output signal • Internal limitation to ±32767
nOut2_a INT	Output signal • Internal limitation to ±32767
nOut3_a INT	Output signal • Internal limitation to ±32767
nOut4_a INT	Output signal • Internal limitation to ±32767

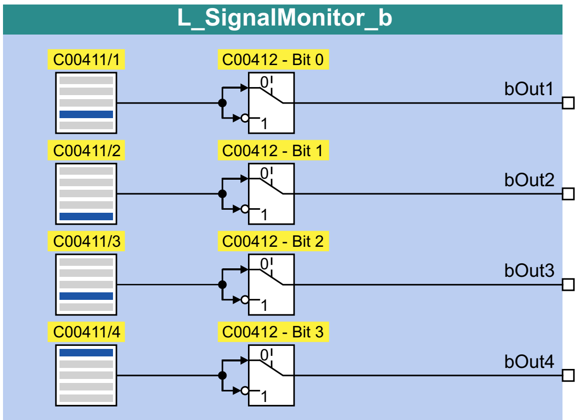
Parameters

Parameters	Possible settings			Information
C00410/1 ... C00410/4	See selection list - analog signals			Selection of the signal sources for *nOut1_a ... nOut4_a*
C00413/1 C00413/3 C00413/5 C00413/7	-199.99	%	+199.99	Offset
C00413/2 C00413/4 C00413/6 C00413/8	-199.99	%	+199.99	Gain

19.1.160 L_SignalMonitor_b

This FB outputs four binary signals which can be selected from a list of binary output signals of all function blocks provided in the device.

- Inversion of the output signals can be set.



outputs

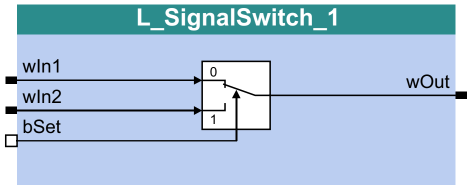
Designator Data type	Value/meaning
bOut1 ... bOut4 BOOL	Output signal FALSE / TRUE

Parameters

Parameters	Possible settings		Information
C00411/1 ... C00411/4	See selection list - digital signals		Selection of the signal sources for *bOut1 ... bOut4*
C00412	Value is bit-coded:		Inversion • Bit set = inversion active
	Bit 0	bOut1 inverted	
	Bit 1	bOut2 inverted	
	Bit 2	bOut3 inverted	
	Bit 3	bOut4 inverted	
	Bit 4	Reserved	
	Bit 5	Reserved	
	Bit 6	Reserved	
	Bit 7	Reserved	

19.1.161 L_SignalSwitch_1

This FB switches between two input signals of the "WORD" data type. The switch-over is controlled by means of a boolean input signal.



inputs

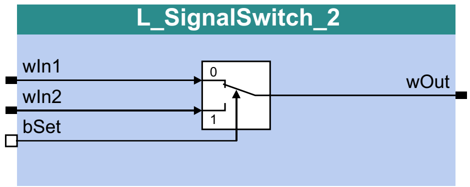
Designator Data type	Information/possible settings	
wIn1 WORD	Input signal 1	
wIn2 WORD	Input signal 2	
bSet BOOL	Selection of the input signal for the output at *wOut*	
	FALSE	*wIn1*
	TRUE	*wIn2*

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

19.1.162 L_SignalSwitch_2

This FB switches between two input signals of the "WORD" data type. The switch-over is controlled by means of a boolean input signal.



inputs

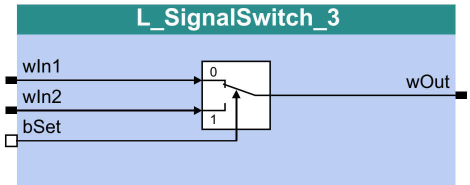
Designator Data type	Information/possible settings	
wIn1 WORD	Input signal 1	
wIn2 WORD	Input signal 2	
bSet BOOL	Selection of the input signal for the output at *wOut*	
	FALSE	*wIn1*
	TRUE	*wIn2*

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

19.1.163 L_SignalSwitch_3

This FB switches between two input signals of the "WORD" data type. The switch-over is controlled by means of a boolean input signal.



inputs

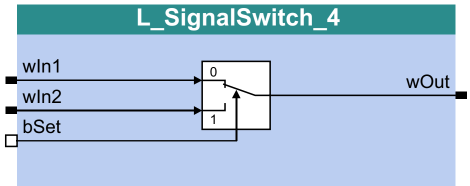
Designator Data type	Information/possible settings	
wIn1 WORD	Input signal 1	
wIn2 WORD	Input signal 2	
bSet BOOL	Selection of the input signal for the output at *wOut*	
	FALSE	*wIn1*
	TRUE	*wIn2*

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

19.1.164 L_SignalSwitch_4

This FB switches between two input signals of the "WORD" data type. The switch-over is controlled by means of a boolean input signal.



inputs

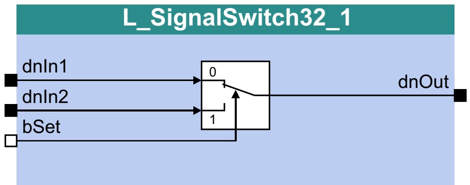
Designator Data type	Information/possible settings	
wIn1 WORD	Input signal 1	
wIn2 WORD	Input signal 2	
bSet BOOL	Selection of the input signal for the output at *wOut*	
	FALSE	*wIn1*
	TRUE	*wIn2*

outputs

Designator Data type	Value/meaning
wOut WORD	Output signal

19.1.165 L_SignalSwitch32_1

This FB switches between two input signals of the "DINT" data type. The switch-over is controlled by means of a boolean input signal.



inputs

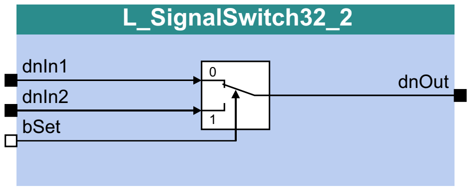
Designator Data type	Information/possible settings	
dnIn1 DINT	Input signal 1	
dnIn2 DINT	Input signal 2	
bSet BOOL	Selection of the input signal for the output at *dnOut*	
	FALSE	*dnIn1*
	TRUE	*dnIn2*

outputs

Designator Data type	Value/meaning
dnOut DINT	Output signal

19.1.166 L_SignalSwitch32_2

This FB switches between two input signals of the "DINT" data type. The switch-over is controlled by means of a boolean input signal.



inputs

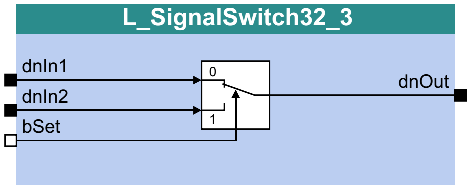
Designator Data type	Information/possible settings	
dnIn1 DINT	Input signal 1	
dnIn2 DINT	Input signal 2	
bSet BOOL	Selection of the input signal for the output at *dnOut*	
	FALSE	*dnIn1*
	TRUE	*dnIn2*

outputs

Designator Data type	Value/meaning
dnOut DINT	Output signal

19.1.167 L_SignalSwitch32_3

This FB switches between two input signals of the "DINT" data type. The switch-over is controlled by means of a boolean input signal.



inputs

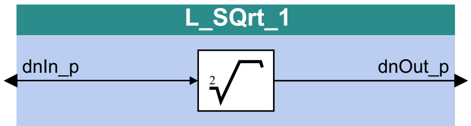
Designator / Data type	Information/possible settings	
dnIn1 / DINT	Input signal 1	
dnIn2 / DINT	Input signal 2	
bSet / BOOL	Selection of the input signal for the output at *dnOut*	
	FALSE	*dnIn1*
	TRUE	*dnIn2*

outputs

Designator / Data type	Value/meaning
dnOut / DINT	Output signal

19.1.168 L_SQrt_1

This FB outputs the square root of the DINT input value.



inputs

Designator Data type	Information/possible settings
dnIn_p DINT	Input signal

outputs

Designator Data type	Value/meaning
dnOut_p DINT	Output signal

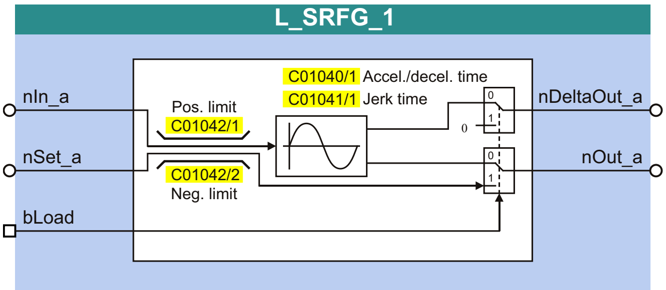
Function

$$dnOut\_p = \sqrt{dnIn\_p}$$

19.1.169 L_SRFG_1

This FB is a ramp function generator with S-shaped ramps for limiting the increase over time of analog signals. The S-shape of the ramps is a result of trapezoidal acceleration.

- The ramp function generator is provided with a setting function so that a value can directly be loaded into the internal ramp generator.
- The balanced acceleration/deceleration time is set in C01040.
- An S-ramp time can be set in C01041 for jerk-free acceleration until maximum acceleration is reached.
- At the *nDeltaOut_a* output, the dy/dt slope of the *nOut_a* output signal is output.



inputs

Designator Data type	Information/possible settings	
nIn_a INT	Input signal	
nSet_a INT	Starting value for the ramp function generator • Will be accepted if *bLoad* = TRUE	
bLoad BOOL	Initialise ramp function generator	
	FALSE	With the set acceleration/deceleration time, the ramp function generator switches over from the value loaded via *nSet_a* to the value at *nIn_a*.
	TRUE	At the *nOut_a* output, *nSet_a* is output. • *nDeltaOut_a* remains at 0 %.

outputs

Designator Data type	Value/meaning
nDeltaOut_a INT	Acceleration of the ramp function generator • Internal limitation to ±100 % (100 % ≡ 16384)
nOut_a INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Parameters

Parameters	Possible settings			Information
C01040/1	0.001	s	999.999	Acceleration/Deceleration time • Initialisation: 100.000 s
C01041/1	0.001	s	50.000	S-ramp time • Initialisation: 0.200 s

Parameters	Possible settings			Information
C01042/1	-199.99	s	199.99	Pos. limit • Initialisation: 100.00 %
C01042/2	-199.99	s	199.99	Neg. limit • Initialisation: -100.00 %

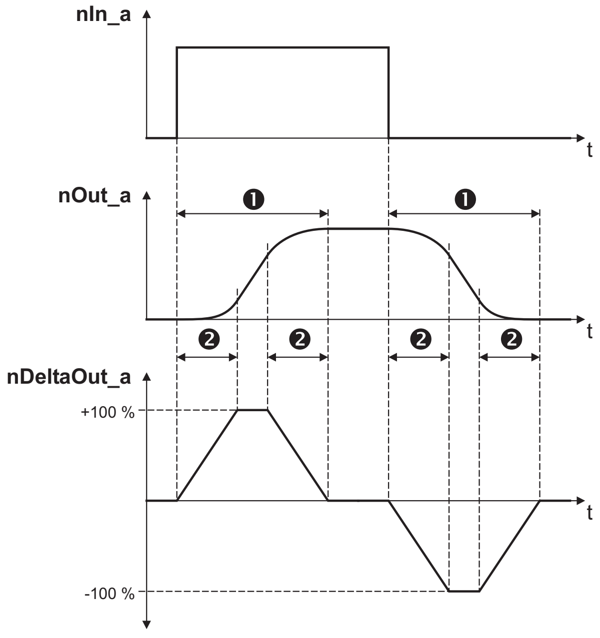
Loading the ramp function generator

When *bLoad* is set to TRUE, the ramp function generator is loaded with the signal at *nSet_a*. This value is accepted immediately and output at *nOut_a*. There is no acceleration or deceleration via an S shape. As long as *bLoad* = TRUE, the ramp function generator remains inhibited.

Acceleration/Deceleration time and S-ramp time

The acceleration/deceleration time and the S-ramp time for jerk-free acceleration can be set separately.

- Acceleration/Deceleration time = Time until the *nOut_a* output value has reached the *nIn_a* input value.
- S-ramp time = Time until the ramp function generator operates at maximum acceleration.



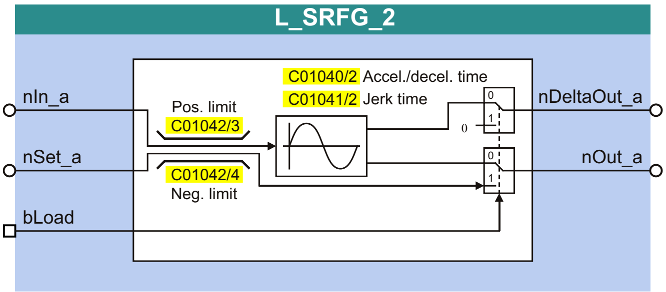
❶ C01040/x: Acceleration/Deceleration time
❷ C01041/x: S-ramp time

[19-75] Signal flow

19.1.170 L_SRFG_2

This FB is a ramp function generator with S-shaped ramps for limiting the increase over time of analog signals. The S-shape of the ramps is a result of trapezoidal acceleration.

- The ramp function generator is provided with a setting function so that a value can directly be loaded into the internal ramp generator.
- The balanced acceleration/deceleration time is set in C01040.
- An S-ramp time can be set in C01041 for jerk-free acceleration until maximum acceleration is reached.
- At the *nDeltaOut_a* output, the dy/dt slope of the *nOut_a* output signal is output.



inputs

Designator Data type	Information/possible settings	
nIn_a INT	Input signal	
nSet_a INT	Starting value for the ramp function generator • Will be accepted if *bLoad* = TRUE	
bLoad BOOL	Initialise ramp function generator	
	FALSE	With the set acceleration/deceleration time, the ramp function generator switches over from the value loaded via *nSet_a* to the value at *nIn_a*.
	TRUE	At the *nOut_a* output, *nSet_a* is output. • *nDeltaOut_a* remains at 0 %.

outputs

Designator Data type	Value/meaning
nDeltaOut_a INT	Acceleration of the ramp function generator • Internal limitation to ±100 % (100 % ≡ 16384)
nOut_a INT	Output signal • Internal limitation to ±199 % (100 % ≡ 16384)

Parameters

Parameters	Possible settings			Information
C01040/2	0.001	s	999.999	Acceleration/Deceleration time • Initialisation: 100.000 s
C01041/2	0.001	s	50.000	S-ramp time • Initialisation: 0.200 s

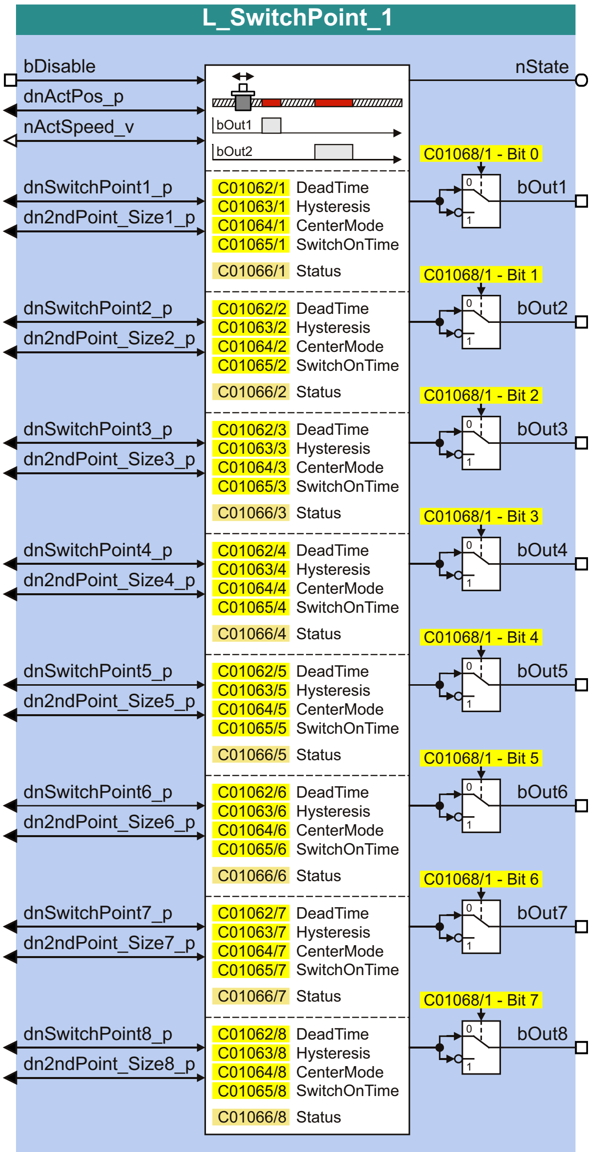
Parameters	Possible settings			Information
C01042/3	-199.99	s	199.99	Pos. limit • Initialisation: 100.00 %
C01042/4	-199.99	s	199.99	Neg. limit • Initialisation: -100.00 %

For a detailed functional description see L_SRFG_1.

19.1.171 L_SwitchPoint_1

This FB provides position switch points, i.e. digital switches the binary statuses (FALSE/TRUE) of which depend on the actual position.

- From version 12.00.00, eight instead of four position switch points are available.
- A position switch point serves to start peripherals as paint nozzles or knifes depending on the tool position.
- Moreover the FB supports the compensation of delay times of external switching elements (dead time compensation).
- By setting a running time, position/time-based cams can be realised as well.



inputs

Designator / Data type	Information/possible settings	
bDisable BOOL	Deactivate position switch points • This input has the highest priority.	
	TRUE	Position switch points are deactivated. • Outputs *bOut1* ... *bOut8* = FALSE
dnActPos_p DINT	Actual position in [increments]	
nActSpeed_v INT	Actual speed in [increments/ms] • Scaling: 16384 ≡ 15000 rpm	
dnSwitchPoint**x**_p DINT	Position switch point **x**: Position of the first switching point in [increments]	
dn2ndPoint_Size**x**_p DINT	Position switch point **x**: Position of the second switching point or size of the switching window depending on the CenterMode set in C01064/x:	
	When CenterMode = FALSE: Position of the second switching point in [increments] • *dn2ndPoint_Size_p* must be higher than *dnSwitchPoint_p*.	
	When CenterMode = TRUE: Size of the switching window in [increments] • Due to the symmetrical arrangement of the window around the first switching point, uneven values are rounded to even values. • Only positive values are permissible.	

outputs

Designator / Data type	Value/meaning	
bOut1 ... bOut8 BOOL	Switching output 1 ... 8	
	TRUE	The actual position is inside the defined switching window.
nState INT	Status	
	1	FB is not active
	2	OK
	3	The data form resulting from the switching points are not plausible (*dnSwitchPoint_p*, *dn2ndPoint_Size_p*, parameterised hysteresis).

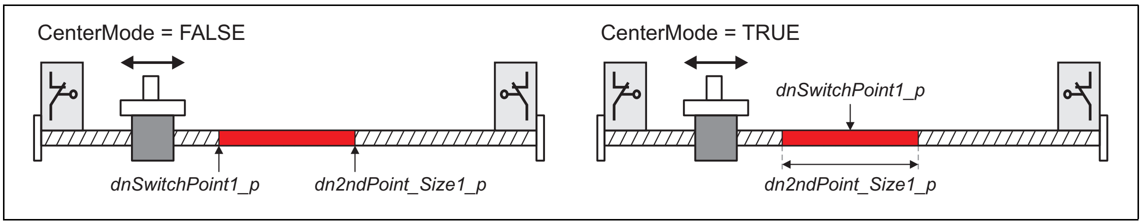
Parameters

Parameters	Possible settings			Information
C01062/1...8	0	µs	65535	Dead time for dead time compensation • The resulting switching position is not subject to a plausibility check. • Lenze setting: 0 µs
C01063/1...8	0	Incr.	65535	Switching hysteresis • Lenze setting: 0 incr.
C01064/1...8				CenterMode • Definition how the *dn2ndPoint_Size_p* selection is interpreted.
	False	*dn2ndPoint_Size_p* defines the second switching point. (Lenze setting)		
	True	*dn2ndPoint_Size_p* defines the size of the switching window.		

Parameters	Possible settings			Information
C01065/1...8	0	ms	60000	Running time for position/time-based cams • Lenze setting: 0 ms
	0 ms = position-based cam			
C01066/1...8				Status 1 ... 4 • Read only.
	0	OK		
	10	FB is not active		
	100	Switching points not plausible		
C01068/1 From version 12.00.00	Setting is bit coded:			Inversion of outputs • Lenze setting: 0x0000
	Bit 0	Inversion of output 1		
	Bit 1	Inversion of output 2		
	Bit 2	Inversion of output 3		
	Bit 3	Inversion of output 4		
	Bit 4	Inversion of output 5		
	Bit 5	Inversion of output 6		
	Bit 6	Inversion of output 7		
	Bit 7	Inversion of output 8		
	Bit 8	Reserved		
	...			
	Bit 15			

19.1.171.1 Definition of the switching range

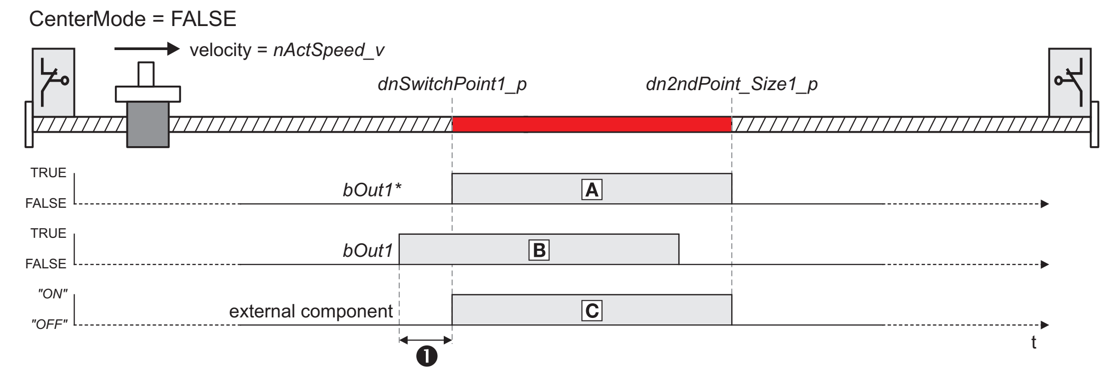
Depending on the setting of the CenterMode (C01064/x), the switching range can be defined via a start and end position or via the data of the center point/size of switching range:



[19-76] Definition of the switching range

19.1.171.2 Dead time compensation

The delay time of external switching elements can be compensated by setting the corresponding delay time in C01062/x. From this input and the current speed ,the FB calculates a preliminary stop, i.e. the *bOut* switching output is set correspondingly earlier.

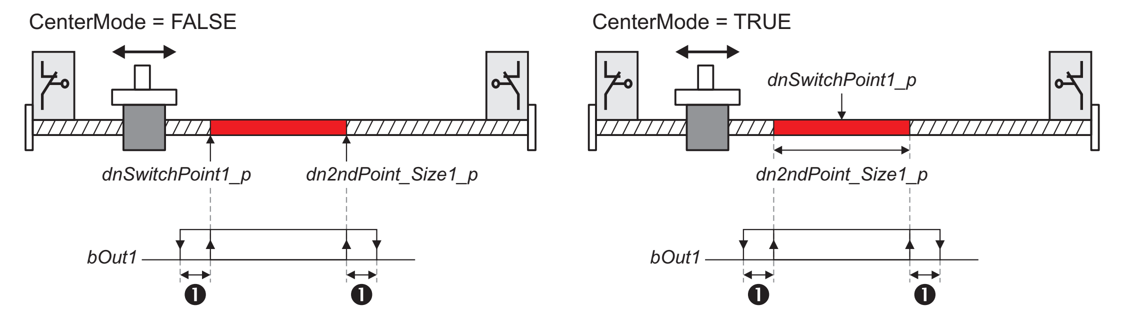


A Switching performance without dead time compensation
B Switching performance with dead time compensation
C Switching performance of the external switching element (with dead time)
❶ C01062/1: Dead time

[19-77] Dead time compensation

19.1.171.3 Switching hysteresis

Setting a hysteresis in C01063/x serves to prevent a permanent state change of the *bOut* switching output that may be caused by actual position value changes due to a mechanical irregularity at the axis.



❶ C01063/1: Hysteresis

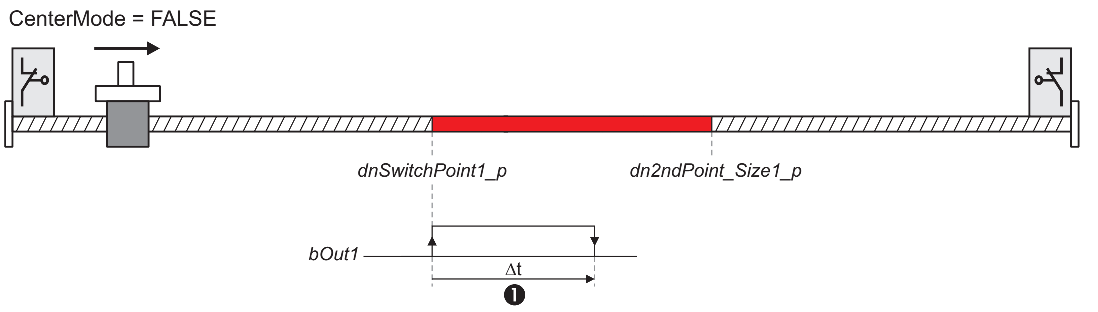
[19-78] Switching hysteresis

Tip!

Please note that a delay time set in C01062/x shifts the switching points including the hysteresis.

19.1.171.4 Position/time-based cams

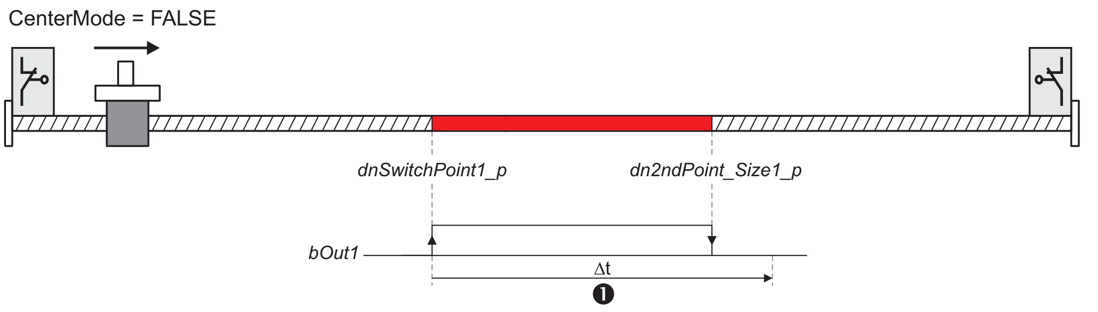
The setting of a running time in C01065/x higher 0 ms serves to realise "position/time-based cams": If the drive reaches the switching range, the *bOut* switching output is set to TRUE and reset to FALSE after the set switching time has elapsed.



❶ C01065/1: Running time

[19-79] Position/time-based cams (here without switching hysteresis and dead time compensation)

If the switching range is left before the running time has expired, the *bOut* switching output is reset to FALSE:



❶ C01065/1: Running time

[19-80] Position/time-based cams: Behaviour when the switching range is left before the running time has expired

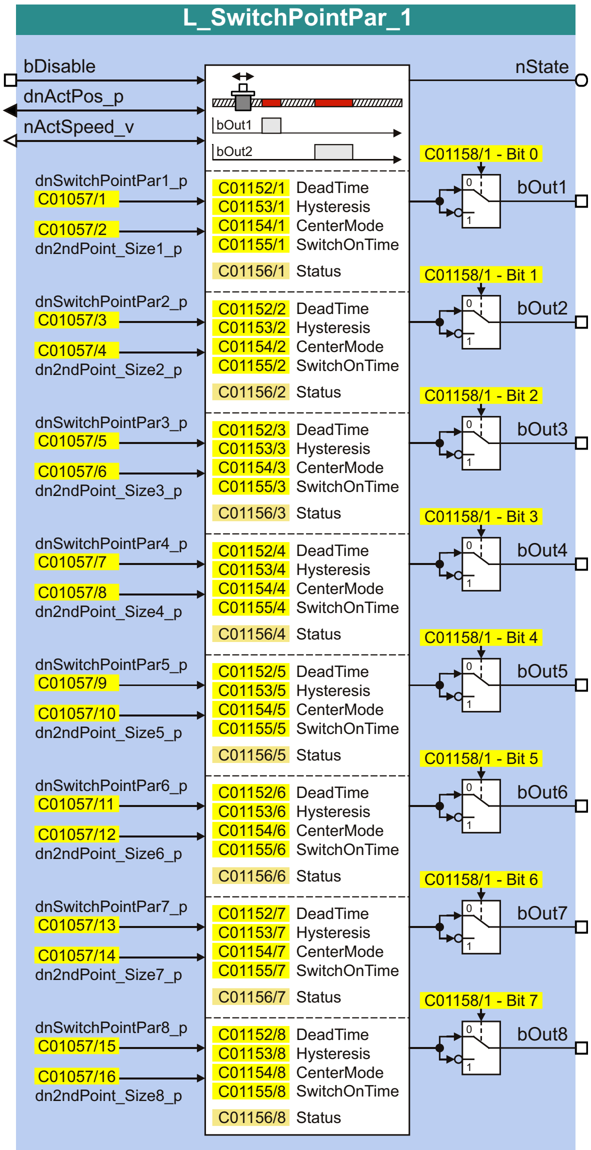
Tip!

Please note that for a resetting of the switching output the switching range must be left first.

19.1.172 L_SwitchPointPar_1

This FB provides position switch points, i.e. digital switches the binary statuses (FALSE/TRUE) of which depend on the actual position.

- This FB has the same function as the FB L_SwitchPoint_1. However, for this FB the position switch point positions must be specified via parameters instead of FB inputs.
- A position switch point serves to start peripherals as paint nozzles or knifes depending on the tool position.
- Moreover the FB supports the compensation of delay times of external switching elements (dead time compensation).
- By setting a running time, position/time-based cams can be realised as well.



inputs

Designator / Data type	Information/possible settings	
bDisable (BOOL)	Deactivate position switch points • This input has the highest priority.	
	TRUE	Position switch points are deactivated. • Outputs *bOut1* ... *bOut8* = FALSE
dnActPos_p (DINT)	Actual position in [increments]	
nActSpeed_v (INT)	Actual speed in [increments/ms] • Scaling: 16384 ≡ 15000 rpm	

outputs

Designator / Data type	Value/meaning	
bOut1 ... bOut8 (BOOL)	Switching output 1 ... 8	
	TRUE	The actual position is inside the defined switching window.
nState (INT)	Status	
	1	FB is not active
	2	OK
	3	The data form resulting from the switching points are not plausible (*dnSwitchPoint_p*, *dn2ndPoint_Size_p*, parameterised hysteresis).

Parameters

Parameters	Possible settings			Information
C01152/1...8	0	µs	65535	Dead time for dead time compensation • The resulting switching position is not subject to a plausibility check. • Lenze setting: 0 µs
C01153/1...8	0	Incr.	65535	Switching hysteresis • Lenze setting: 0 incr.
C01154/1...8				CenterMode • Definition how the *dn2ndPoint_Size_p* selection is interpreted.
	False	*dn2ndPoint_Size_p* defines the second switching point. (Lenze setting)		
	True	*dn2ndPoint_Size_p* defines the size of the switching window.		
C01155/1...8	0	ms	60000	Running time for position/time-based cams • Lenze setting: 0 ms
	0 ms = position-based cam			
C01156/1...8				Status 1 ... 4 • Read only.
	0	OK		
	10	FB is not active		
	100	Switching points not plausible		

Parameters	Possible settings			Information
C01157/1...16	0.0000	units	214748.0000	Position switch point positions • Lenze setting: 0.0000 units

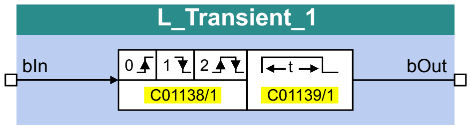
	dnSwitchPoint**x**_p = position of the first switching point for position switch point **x**.
	dn2ndPoint_Size**x**_p = position of the second switching point or size of the switching window for position switch point **x**, depending on the CenterMode set in C01154/x:
	• When CenterMode = FALSE: Position of the second switching point
	  • dn2ndPoint_Size**x**_p must be higher than dnSwitchPoint**x**_p.
	• When CenterMode = TRUE: Size of the switching window
	  • Due to the symmetrical arrangement of the window around the first switching point, uneven values are rounded to even values.
	  • Only positive values are permissible.			

C01158/1	Setting is bit coded:			Inversion of outputs • Lenze setting: 0x0000
	Bit 0	Inversion of output 1		
	Bit 1	Inversion of output 2		
	Bit 2	Inversion of output 3		
	Bit 3	Inversion of output 4		
	Bit 4	Inversion of output 5		
	Bit 5	Inversion of output 6		
	Bit 6	Inversion of output 7		
	Bit 7	Inversion of output 8		
	Bit 8	Reserved		
	...			
	Bit 15			

For a detailed functional description see L_SwitchPoint_1.

19.1.173 L_Transient_1

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01138/1.

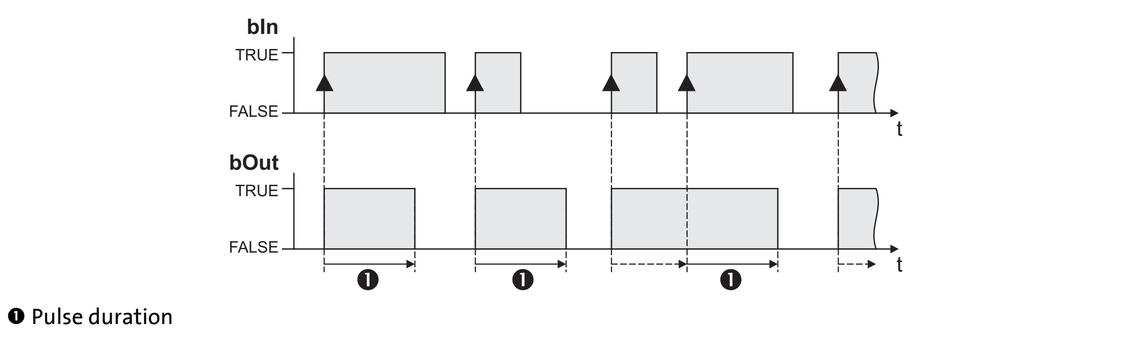
outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

Parameters

Parameters	Possible settings			Information
C01138/1				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01139/1	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

19.1.173.1 Function 0: Evaluate rising signal edges

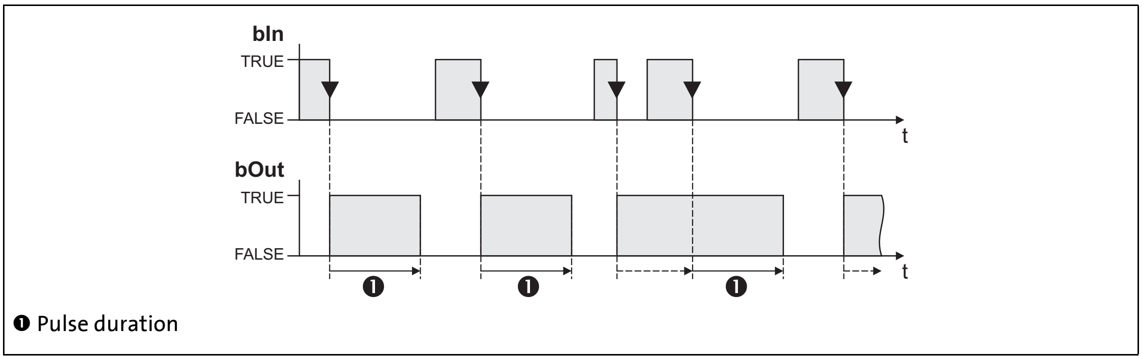


[19-81] Switching performance for function selection "0: High edge"

Functional sequence

1. A FALSE-TRUE edge at the *bIn* input sets the *bOut* output to TRUE.
2. After the parameterised pulse duration has elapsed, the *bOut* output is reset to FALSE unless another FALSE/TRUE edge has been set at the *bIn* input.
   - If an additional FALSE-TRUE edge occurs at the *bIn* input, the pulse duration starts again from the beginning, i.e. the *bOut* output can be retriggered.

19.1.173.2 Function 1: Evaluate falling signal edges

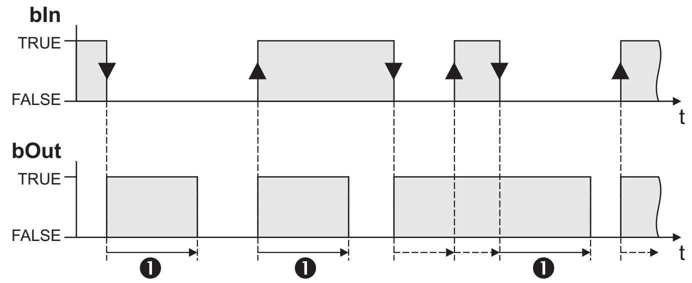


[19-82] Switching performance for function selection "1: Low edge"

Functional sequence

1. A TRUE-FALSE edge at the *bIn* inputs sets the *bOut* output to TRUE.
2. After the parameterised pulse duration has elapsed, the *bOut* output is reset to FALSE unless another TRUE/FALSE edge has been set at the *bIn* input.
   - If an additional TRUE-FALSE edge occurs at the *bIn* input, the pulse duration starts again from the beginning, i.e. the *bOut* output can be retriggered.

19.1.173.3 Function 2: Evaluate rising and falling signal edges



❶ Pulse duration

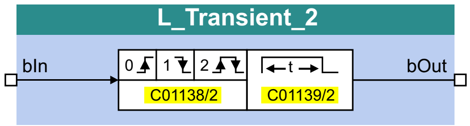
[19-83] Switching performance for function selection "2: High and low edge"

Functional sequence

1. A signal change (FALSE/TRUE edge or TRUE/FALSE edge) at the *bIn* input sets the *bOut* output to TRUE.
2. After the parameterised pulse duration has elapsed, the *bOut* output is reset to FALSE unless another signal change has taken place at the *bIn* input.
   - In case of another signal change at the input *bIn*, the pulse time restarts to elapse, i.e. the output *bOut* can be retriggered.

19.1.174 L_Transient_2

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01138/2.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

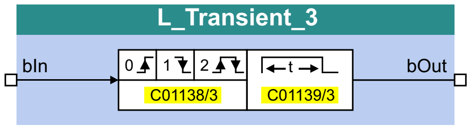
Parameters

Parameters	Possible settings			Information
C01138/2				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01139/2	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

For a detailed functional description see L_Transient_1.

19.1.175 L_Transient_3

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01138/3.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

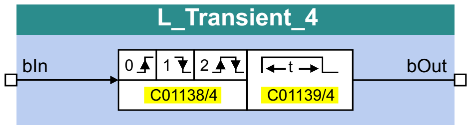
Parameters

Parameters	Possible settings			Information
C01138/3				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01139/3	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

For a detailed functional description see L_Transient_1.

19.1.176 L_Transient_4

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01138/4.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

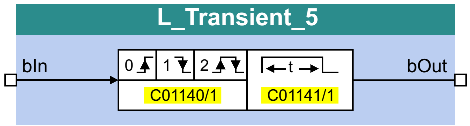
Parameters

Parameters	Possible settings			Information
C01138/4				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01139/4	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

For a detailed functional description see L_Transient_1.

19.1.177 L_Transient_5

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01140/1.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

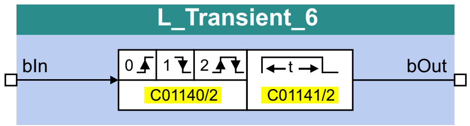
Parameters

Parameters	Possible settings			Information
C01140/1				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01141/1	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

For a detailed functional description see L_Transient_1.

19.1.178 L_Transient_6

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01140/2.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

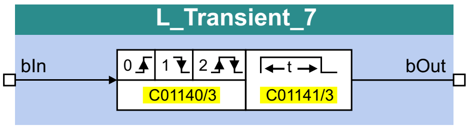
Parameters

Parameters	Possible settings			Information
C01140/2				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01141/2	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

For a detailed functional description see L_Transient_1.

19.1.179 L_Transient_7

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01140/3.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

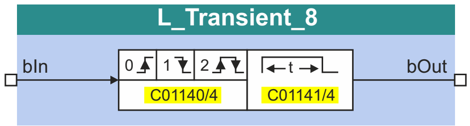
Parameters

Parameters	Possible settings			Information
C01140/3				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01141/3	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

For a detailed functional description see L_Transient_1.

19.1.180 L_Transient_8

This FB serves to evaluate digital signal edges and convert them into timed, retriggerable pulses. Rising signal edges, falling signal edges or both signal edges can be evaluated.



inputs

Designator Data type	Information/possible settings
bIn BOOL	Input for edge evaluation • The function depends on the selection of edge evaluation in C01140/4.

outputs

Designator Data type	Value/meaning
bOut BOOL	Output (retriggerable)

Parameters

Parameters	Possible settings			Information
C01140/4				Function • Selection of edge evaluation
	0	High edge		Lenze setting
	1	Low edge		
	2	High and low edge		
C01141/4	0.001	s	60.000	Pulse duration • Lenze setting: 0.001 s

For a detailed functional description see L_Transient_1.

19.2 System blocks

This chapter describes the system blocks which are available for the inverter in the FB Editor.

The function blocks are described in the chapter "Function blocks". (🕮 1468)

Overview of system blocks available

System block	Function	can be inserted into level: I/O	Appl.
LS_AnalogInput	Interface to the analog input terminals ▸ Analog terminals (🕮 425)	●	
LS_AnalogOutput	Interface to the analog output terminals ▸ Analog terminals (🕮 425)	●	
LS_AxisBusAux	Interface to the axis bus ▸ Axis bus (🕮 891)	●	●
LS_AxisBusIn		●	●
LS_AxisBusIO		●	●
LS_AxisBusOut		●	●
LS_CANManagement	Control of internal functions of the CAN driver and display of the "Operational" status as well as the node address ▸ System bus "CAN on board" (🕮 804)	●	●
LS_DataAccess	*Lenze internal only*		●
LS_DeviceMonitor	Motor control status signals ▸ Motor control (MCTRL) (🕮 143)		●
LS_DFOut	Interface to the digital frequency output (multi-encoder interface X8) • This SB is available from version 12.00.00. ▸ Digital frequency coupling (🕮 365)	●	
LS_DigitalInput	Interface to the digital input terminals ▸ Digital input terminals (🕮 401)	●	
LS_DigitalOutput	Interface to the digital output terminals ▸ Digital output terminals (🕮 422)	●	
LS_DisFree	Display of 8 arbitrary 16-bit signals of the application on display codes	●	●
LS_DisFree_a	Display of 8 arbitrary analog signals of the application on display codes	●	●
LS_DisFree_b	Display of 16 arbitrary digital signals of the application on a bit coded display code	●	●
LS_DisFree_p	Display of 8 arbitrary position signals of the application on display codes	●	●
LS_DriveInterface	Interface to drive control (DCTRL) ▸ Device control (DCTRL) (🕮 105)		●
LS_Keypad	Control via keypad	●	
LS_MotionControlKernel	Interface to the basic drive function implemented in the Motion Control Kernel (MCK) ▸ Basic drive functions (MCK) (🕮 577)		●
LS_MotorInterface	Interface to motor control (MCTRL) ▸ Motor control (MCTRL) (🕮 143)		●
LS_MultiEncoder	Interface to the Multi-Encoder ▸ Multi-Encoder (X8) (🕮 341)	●	●

System block	Function	can be inserted into level: I/O	Appl.
LS_ParFix LS_ParFix_2	Output of frequently used constants (TRUE, FALSE, 100 %, etc.) to be used in the interconnection • LS_ParFix_2 is available from version 02.00.00.	●	●
LS_ParFree LS_ParFree_2	Output of 32 parameterisable 16-bit signals • LS_ParFree_2 is available from version 02.00.00.	●	●
LS_ParFree_a LS_ParFree_a_2	Output of 16 parameterisable analog signals • LS_ParFree_a_2 is available from version 02.00.00.	●	●
LS_ParFree_b	Output of 32 parameterisable digital signals	●	●
LS_ParFree_p	Output of 8 parameterisable position signals	●	●
LS_ParFree_v LS_ParFree_v_2	Output of 8 parameterisable speed signals • LS_ParFree_v_2 is available from version 02.00.00.	●	●
LS_ParFree32	Output of 8 parameterisable 32-bit signals • This SB is available from version 02.00.00.	●	●
LS_ParFreeUnit LS_ParFreeUnit_2	Output of 16 parameterisable position signals with internal conversion of [unit] in [increments] • These SBs are available from version 02.00.00.	●	●
LS_ParReadWrite_1 ... LS_ParReadWrite_6	Reading/Writing of local parameters	●	●
LS_PulseGenerator	Output of 9 fixed frequencies and 1 parameterisable frequency	●	●
LS_Resolver	Interface to the resolver ▶ Resolver (X7) (🕮 335)	●	●
LS_RetainData	Selection and saving of retain data • This SB is available from version 02.00.00.	●	●
LS_SetError_1	Parameterisable responses to user-defined events are tripped ▶ Diagnostics & error management (🕮 719)	●	●
LS_SetError_2		●	●
LS_SyncManagement	Output of status information for synchronising the internal time base ▶ Synchronisation of the internal time base (🕮 912)	●	●
LS_TouchProbe	Interface for touch probe detection ▶ Touch probe detection (🕮 435)	●	●
LS_WriteParamList	Interface to the basic "Parameter change-over" function ▶ Parameter change-over (🕮 914)	●	

Related topics:

▶ Overview of function blocks available (🕮 1468)
▶ Working with the FB Editor (🕮 1416)

19.2.1 LS_AnalogInput

Interface to the analog input terminals.

For a detailed description see the main chapter "I/O terminals":
▸ Internal interfaces | System block "LS_AnalogInput" (📖 433)

19.2.2 LS_AnalogOutput

Interface to the analog output terminals.

For a detailed description see the main chapter "I/O terminals":
▸ Internal interfaces | System block "LS_AnalogInput" (📖 433)

19.2.3 LS_AxisBusAux

Interface to the axis bus.

For a detailed description see main chapter "Axis bus":
▸ Internal interfaces | System block "LS_AxisBusAux" (📖 904)

19.2.4 LS_AxisBusIn

Interface to the axis bus.

For a detailed description see main chapter "Axis bus":
▸ Internal interfaces | System block "LS_AxisBusIn" (📖 903)

19.2.5 LS_AxisBusIO

Interface to the axis bus.

For a detailed description see main chapter "Axis bus":
▸ Internal interfaces | System block "LS_AxisBusIO" (📖 911)

19.2.6 LS_AxisBusOut

Interface to the axis bus.

For a detailed description see main chapter "Axis bus":
- Internal interfaces | System block "LS_AxisBusOut" (902)

19.2.7 LS_CANManagement

Control of internal functions of the CAN driver and display of the "Operational" status as well as the node address.

For a detailed description see the main chapter "System bus CAN on board":
- Internal interfaces | System block "LS_CANManagement" (883)

19.2.8 LS_DataAccess

Only for Lenze-internal use.

19.2.9 LS_DeviceMonitor

Motor control status signals.

For a detailed description see the main chapter "Motor control (MCTRL)":
- Internal status signals | System block "LS_DeviceMonitor" (327)

19.2.10 LS_DFOut

Interface to the digital frequency output (multi-encoder interface X8).

For a detailed description see main chapter "Encoder-/feedback system":
- Internal interfaces | System block "LS_DFOut" (368)

19.2.11 LS_DigitalInput

Interface to the digital input terminals.

For a detailed description see the main chapter "I/O terminals":

▸ Internal interfaces | System block "LS_DigitalInput" (📖 414)

19.2.12 LS_DigitalOutput

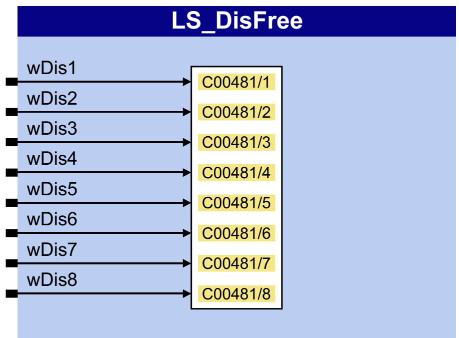
Interface to the digital output terminals.

For a detailed description see the main chapter "I/O terminals":

▸ Internal interfaces | System block "LS_DigitalOutput" (📖 424)

19.2.13 LS_DisFree

This system block displays 8 arbitrary 16-bit signals of the application on display codes.



inputs

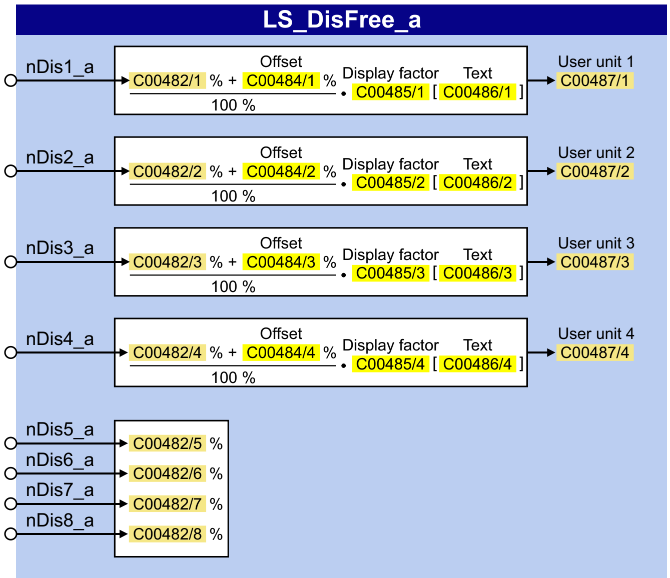
Designator Data type	Information/possible settings
wDis1 ... wDis8 WORD From version 02.00.00: wC481_1 ... wC481_8_a	Inputs for any 16-bit signals of the application **Note:** From version 02.00.00 the inputs are named according to the display parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00481/1...8	0x0000		0xFFFF	Display of the 16-bit signals which are applied at the *wDis1 ... wDis8* inputs

19.2.14 LS_DisFree_a

This system block displays 8 arbitrary analog signals of the application on display codes.



inputs

Designator Data type	Information/possible settings
nDis1_a ... nDis8_a INT From version 02.00.00: nC482_1_a ... nC482_8_a	Inputs for arbitrary analog signals of the application **Note:** From version 02.00.00 the inputs are named according to the display parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00482/1...8	-199.99	%	199.99	Display of the analog signals which are applied at the *nDis1_a ... nDis8_a* inputs
C00484/1...4 ... C00487/1...4	▸ Display of internal process factors in application units			

19.2.14.1 Display of internal process factors in application units

In addition to the display in percent in C00482/1...8, for the first four analog signals *nDis1_a* ... *nDis4_a* the configurable display parameters C00487/1...4 are provided. Via these display parameters, internal process variables can be displayed, e.g. on the keypad, with an individual scaling and an individual unit (from keypad version 02.01).

Configuration of the display parameters (C00487/1...4):

Parameters	Possible settings			Information
C00484/1...4	-199.99	%	199.99	Offset 1 ... 4 • See formula [19-84]. • Lenze setting: 0.00 %
C00485/1...4	-65536.0000		65536.0000	Display factor 1 ... 4 • Scaling of the input variable for the display. • See formula [19-84]. • Lenze setting: 1.0000
C00486/1...4	String (max. 7 digits)			Text 1 ... 4 • For each display value, an individual unit (e.g. "parts") can be set.

$$\text{User unit 1} = \frac{\text{nDis1\_a [\%]} + \text{Offset 1 [\%]}}{\text{100 [\%]}} \cdot \text{Display factor 1 [text 1]}$$

[19-84] Formula for scaling the display

Example 1:

- Input variable *nDis1_a* = 100 %
- Offset 1 (C00484/1) = 0 %
- Display factor 1 (C00485/1) = 123.45
- Text 1 (C00486/1) = "parts"

$$\text{User unit 1} = \frac{100\ [\%] + 0\ [\%]}{100\ [\%]} \cdot 123.45\ [\text{parts}] = 123.45\ \text{parts}$$

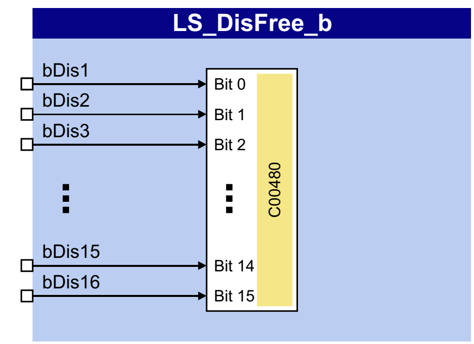
Example 2:

- Input variable *nDis2_a* = 40 %
- Offset 2 (C00484/2) = 35 %
- Display factor 2 (C00485/2) = 20
- Text 2 (C00486/2) = "kg"

$$\text{User unit 2} = \frac{40\ [\%] + 35\ [\%]}{100\ [\%]} \cdot 20\ [\text{kg}] = 15.00\ \text{kg}$$

19.2.15 LS_DisFree_b

This system block displays 16 arbitrary digital signals of the application on a bit coded display code.



inputs

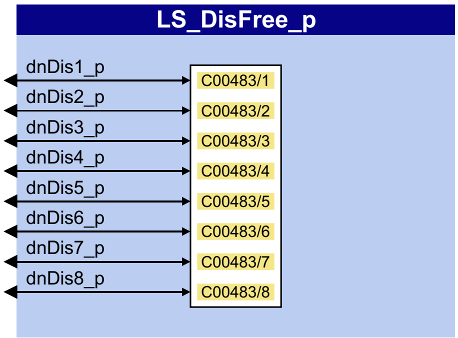
Designator Data type	**Information/possible settings**
bDis1 ... bDis16 BOOL From version 02.00.00: bC480_B0 ... bC480_B16	Inputs for arbitrary digital signals of the application **Note:** From version 02.00.00 the inputs are named according to the display parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00480	0x0000		0xFFFF	Display of the digital signals as hexadecimal values which are applied at the *bDis1* ... *bDis16* inputs
	Bit 0	Signal level at the *bDis1* input		
	Bit 1	Signal level at the *bDis2* input		
	Bit 2	Signal level at the *bDis3* input		
	...	...		
	Bit 15	Signal level at the *bDis16* input		

19.2.16 LS_DisFree_p

This system block displays 8 arbitrary position signals of the application on display codes.



inputs

Designator Data type	Information/possible settings
dnDis1_p ... dnDis8_p DINT From version 02.00.00: dnC483_1_p ... dnC483_8_p	Inputs for arbitrary position signals of the application **Note:** From version 02.00.00 the inputs are named according to the display parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00483/1...8	-2147483647	Incr.	2147483647	Display of the position signals which are applied at the *dnDis1_p* ... *dnDis8_p* inputs

19.2.17 LS_DriveInterface

Interface to internal device control.

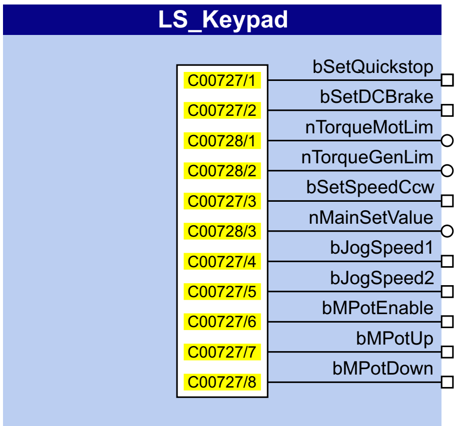
For a detailed description see main chapter "Device control (DCTRL)":

▸ Internal interfaces | "LS_DriveInterface" system block (📖 135)

19.2.18 LS_Keypad

This system block is used on I/O interconnection level if the "Keypad" control mode has been selected in C00007.

In the "Keypad" control mode, the **LS_Keypad** system block passes on various setpoints and control commands to the technology application which can be selected/activated via codes using the keypad.



outputs

Designator Data type	Value/meaning
bSetQuickstop BOOL	C00727/1 = "1" ≡ Request quick stop
bSetDCBrake BOOL	C00727/2 = "1" ≡ Request DC-injection braking
nTorqueMotLim INT	Torque limit in motor mode set in C00728/1 • Lenze setting: 100.00 %
nTorqueGenLim INT	Torque limit in generator mode set in C00728/2 • Lenze setting: 100.00 %
bSetSpeedCcw BOOL	C00727/3 = "1" ≡ Request reversal
nMainSetValue INT	Setpoint speed set in C00728/3 • Lenze setting: 0.00 %
bJogSpeed1 BOOL	C00727/4 = "1" ≡ Request fixed speed setpoint 1
bJogSpeed2 BOOL	C00727/5 = "1" ≡ Request fixed speed setpoint 2
bMPotEnable BOOL	C00727/6 = "1" ≡ Motor potentiometer: Request activation
bMPotUp BOOL	C00727/7 = "1" ≡ Motor potentiometer: Request positive acceleration
bMPotDown BOOL	C00727/8 = "1" ≡ Motor potentiometer: Request negative acceleration

Parameters

Parameters	Possible settings			Information
C00727/1...8	0		1	Keypad digital values • Execution of control commands for keypad operation • See the "Outputs" table for the meaning of the individual subcodes
C00728/1...3	-199.99	%	199.99	Analog values - keypad • Selection of various setpoints for operation via keypad • See the "Outputs" table for the meaning of the individual subcodes

19.2.19 LS_MotionControlKernel

Interface to the basic drive functions implemented in **Motion Control Kernel** (MCK).

For a detailed description see the main chapter "Basic drive functions":

▸ Internal interfaces | System block "LS_MotionControlKernel" (📖 579)

19.2.20 LS_MotorInterface

Interface to internal motor control.

For a detailed description see the main chapterl "Motor control (MCTRL)":

▸ Internal interfaces | System block "LS_MotorInterface" (📖 321)

19.2.21 LS_MultiEncoder

Interface to the Multi-Encoder.

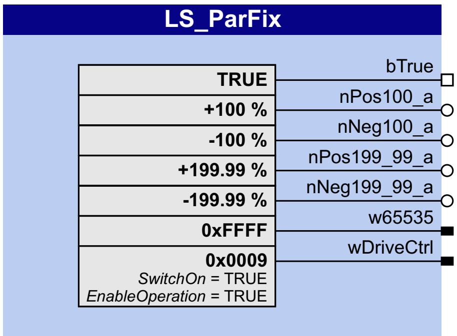
For detailed information please see the main chapter "Encoder/Feedback system":

▸ Internal interfaces | "LS_MultiEncoder" system block (📖 369)

19.2.22 LS_ParFix

This system block outputs various fixed values (constants) to be used in the interconnection.

- From version 02.00.00, another **LS_ParFix_2** system block is available which can be used e.g. in the application level.

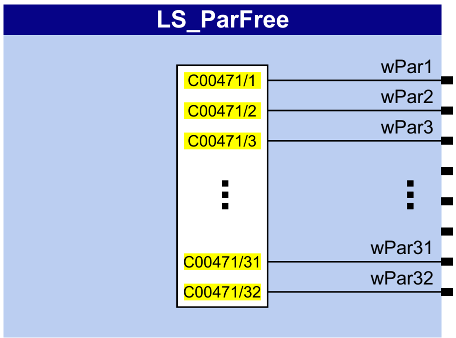


outputs

Designator Data type	**Value/meaning**
bTrue BOOL	1 ≡ TRUE
nPos100_a INT	16384 ≡ + 100 %
nNeg100_a INT	-16384 ≡ - 100 %
nPos199_99_a INT	32767 ≡ + 199.99 %
nNeg199_99_a INT	-32767 ≡ - 199.99 %
w65535 WORD	65535 ≡ 0xFFFF
wDriveCtrl WORD	9 ≡ 0x0009 • Bit 0, SwitchOn = TRUE • Bit 3, EnableOperation = TRUE • All others: FALSE See also: ▸ wCANControl/wMCIControl control words (📖 138)

19.2.23 LS_ParFree

This system block outputs 32 parameterisable 16-bit signals.



outputs

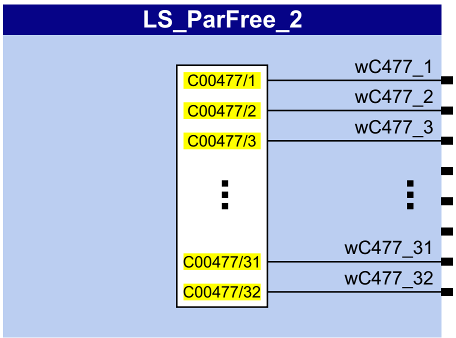
Designator Data type	Value/meaning
wPar1 ... wPar32 WORD From version 02.00.00: wC471_1 ... wC471_32	Output of the 16-bit signals parameterised in C00471/1...32 **Note:** From version 02.00.00 the outputs are named according to the respective setting parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00471/1...32	0x0000		0xFFFF	Setting of the 16-bit signals to be output

19.2.24 LS_ParFree_2

This system block outputs 32 parameterisable 16-bit signals.



outputs

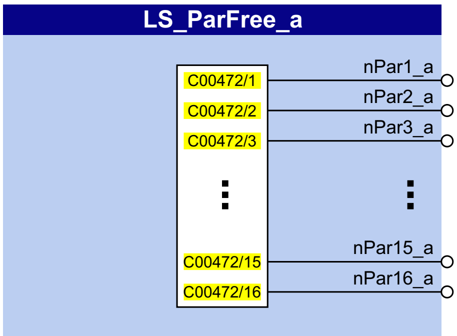
Designator Data type	Value/meaning
wC477_1 ... wC477_32 WORD	Output of the 16-bit signals parameterised in C00477/1...32

Parameters

Parameters	Possible settings			Information
C00477/1...32	0x0000		0xFFFF	Setting of the 16-bit signals to be output

19.2.25 LS_ParFree_a

This system block outputs 16 parameterisable analog signals.



outputs

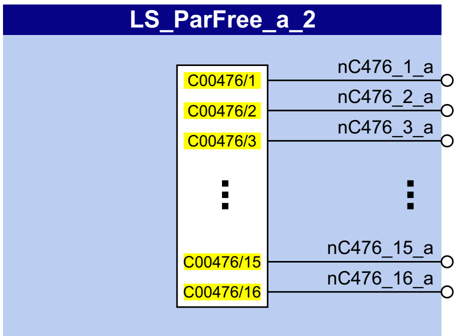
Designator Data type	Value/meaning
nPar1_a ... nPar16_a INT From version 02.00.00: nC472_1_a ... nC472_16_a	Output of the analog signals parameterised in C00472/1...16 **Note:** From version 02.00.00 the outputs are named according to the respective setting parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00472/1...16	-199.99	%	+199.99	Selection of analog signals to be output

19.2.26 LS_ParFree_a_2

This system block outputs 16 parameterisable analog signals.



outputs

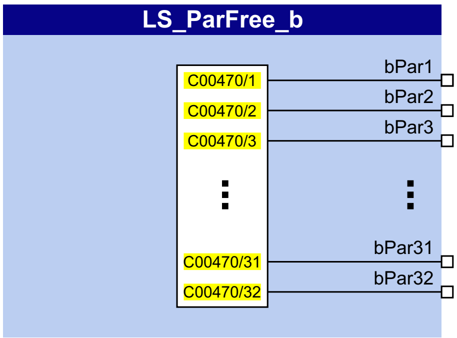
Designator Data type	Value/meaning
nC476_1_a ... nC476_16_a INT	Output of the analog signals parameterised in C00476/1...16

Parameters

Parameters	Possible settings			Information
C00476/1...16	-199.99	%	+199.99	Selection of analog signals to be output

19.2.27 LS_ParFree_b

This system block outputs 32 parameterisable digital signals.



outputs

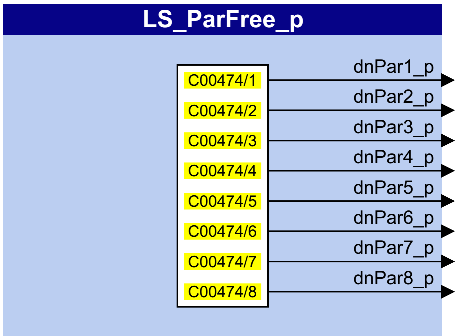
Designator Data type	Value/meaning
bPar1 ... bPar32 BOOL From version 02.00.00: bC470_1 ... bC470_32	Output of the signal levels (FALSE/TRUE) parameterised in C00470/1...32 **Note:** From version 02.00.00 the outputs are named according to the respective setting parameter for an easier allocation.

Parameters

Parameters	Possible settings		Information
C00470/1...32			Selection of signal levels to be output • Bit 0 ... 31 = *bPar1* ... *bPar32*
	0	"FALSE" signal is output	
	1	"TRUE" signal is output	

19.2.28 LS_ParFree_p

This system block outputs 8 parameterisable position signals.



outputs

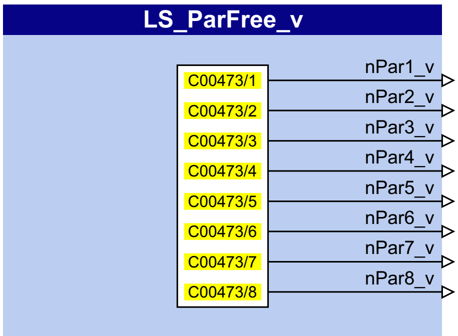
Designator Data type	Value/meaning
dnPar1_p ... dnPar8_p DINT From version 02.00.00: dnC474_1_p ... dnC474_8_p	Output of the position signals parameterised in C00474/1...8 **Note:** From version 02.00.00 the outputs are named according to the respective setting parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00474/1...8	-2147483647	Incr.	2147483647	Setting of the position signals to be output

19.2.29 LS_ParFree_v

This system block outputs 8 parameterisable speed signals.



outputs

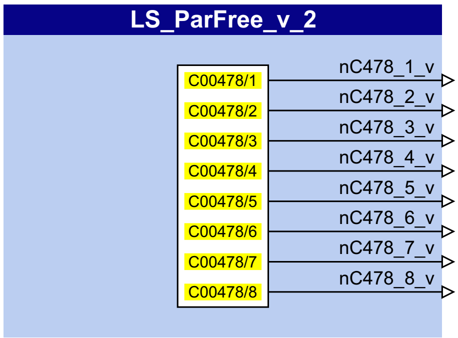
Designator Data type	Information/possible settings
nPar1_v ... nPar8_v INT From version 02.00.00: nC473_1_v ... nC473_8_v	Output of the speed signals parameterised in C00473/1...8 **Note:** From version 02.00.00 the outputs are named according to the respective setting parameter for an easier allocation.

Parameters

Parameters	Possible settings			Information
C00473/1...8	-32767	Incr/ms	+32767	Selection of speed signals to be output

19.2.30 LS_ParFree_v_2

This system block outputs 8 parameterisable speed signals.



outputs

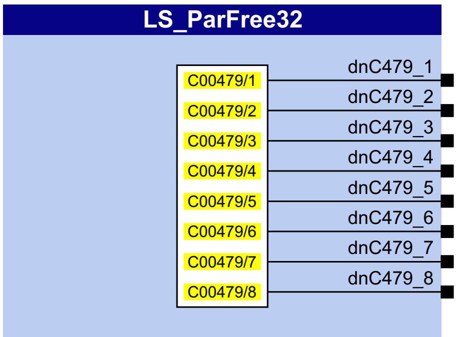
Designator Data type	Information/possible settings
nC478_1_v ... nC478_8_v INT	Output of the speed signals parameterised in C00478/1...8

Parameters

Parameters	Possible settings			Information
C00478/1...8	-32767	Incr/ms	+32767	Selection of speed signals to be output

19.2.31 LS_ParFree32

This system block outputs 8 parameterisable 32-bit signals.



outputs

Designator Data type	Value/meaning
dnC479_1 ... dnC479_8 DINT	Output of the 32-bit signals parameterised in C00479/1...8

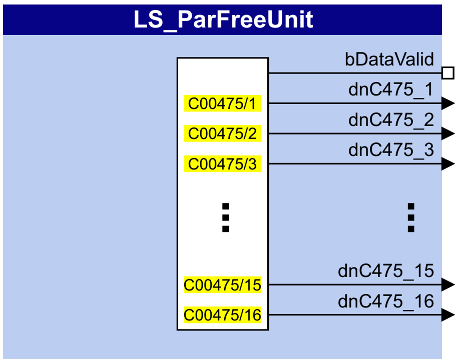
Parameters

Parameters	Possible settings			Information
C00479/1...8	-2147483647		2147483647	Setting of the 32-bit signals to be output

19.2.32 LS_ParFreeUnit

This system block outputs 16 parameterisable position signals.

- In contrast to the SB LS_ParFree_p, the positions are set in the application unit [unit] instead of in [increments].
- The *bDataValid* status signal indicates that the internal position conversion is completed and the output data are consistent.



outputs

Designator Data type	Value/meaning	
bDataValid BOOL	Status signal "Position conversion completed, data consistent"	
	TRUE	The conversion of the positions from [unit] to [increments] is completed.
dnC475_1 ... dnC475_16 DINT	Output of the position signals parameterised in C00475/1...16 in [increments] • Observe *bDataValid* status signal!	

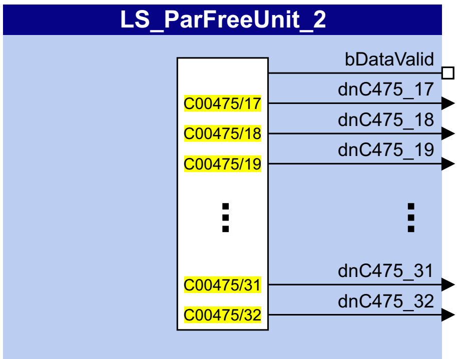
Parameters

Parameters	Possible settings			Information
C00475/1...16	-214748.3647	Unit	214748.3647	Setting of the position signals to be output

19.2.33 LS_ParFreeUnit_2

This system block outputs 16 parameterisable position signals.

- In contrast to the SB LS_ParFree_p, the positions are set in the application unit [unit] instead of in [increments].
- The *bDataValid* status signal indicates that the internal position conversion is completed and the output data are consistent.



outputs

Designator Data type	Value/meaning	
bDataValid BOOL	Status signal "Position conversion completed, data consistent"	
	TRUE	The conversion of the positions from [unit] to [increments] is completed.
dnC475_17 ... dnC475_32 DINT	Output of the position signals parameterised in C00475/17...32 in [increments] • Observe *bDataValid* status signal!	

Parameters

Parameters	Possible settings			Information
C00475/17...32	-214748.3647	Unit	214748.3647	Setting of the position signals to be output

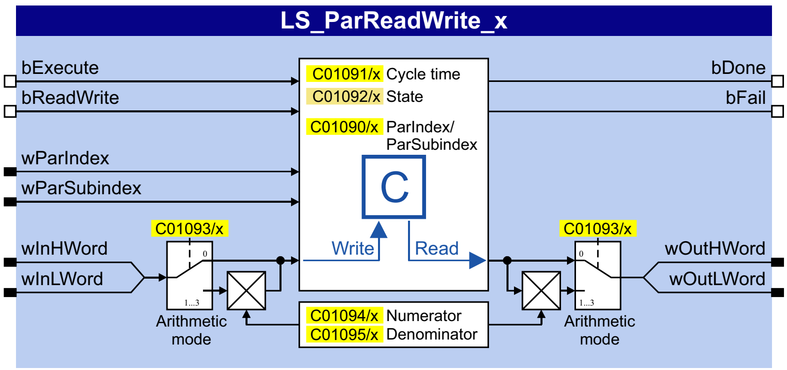
19.2.34 LS_ParReadWrite_1-6

The **LS_ParReadWrite_1** ... **LS_ParReadWrite_6** system blocks are used for reading and writing local parameters.

The system blocks support one-time and cyclic reading/writing in an adjustable time interval.

Note!

The main program, which also contains the parameter communication, runs as a background process with a processing time that can last from several milliseconds to several 100 milliseconds. The outputs *bDone* and *bFail* of the program organisation unit **LS_ParReadWrite** can be used to receive the exact status of the parameter communication.



inputs

Designator Data type	Information/possible settings	
bExecute BOOL	Trip read/write request	
	FALSE↗TRUE	If cycle time (C01091) = "0 ms": <u>One-time</u> reading/writing of the parameter value which has been addressed via the*wParIndex* and *wParSubindex* inputs.
		If cycle time (C01091) > "0 ms": <u>Cyclic</u> reading/writing of the parameter value which has been addressed via the *wParIndex* and *wParSubindex* inputs.
	TRUE↘FALSE	Deactivate cyclic reading/writing again.
bReadWrite BOOL	Selection: Read or write request	
	FALSE	Read request
	TRUE	Write request
wParIndex WORD	Code to be read or written. • The selection can be made alternatively via C01090.	
wParSubindex WORD	Subcode to be read or written. • The selection can be made alternatively via C01090.	
wInHWord wInLWord WORD	Value to be written (DataHigh/DataLow portion)	

outputs

<table>
<tr><th>Designator
Data type</th><th colspan="2">Value/meaning</th></tr>
<tr><td rowspan="3">bDone
BOOL</td><td colspan="2">"Read/Write request successfully completed" status signal
• The output is automatically reset to FALSE if a new request is activated via bExecute or the cycle time (C01091) expires.</td></tr>
<tr><td>TRUE</td><td>Read/Write request successfully completed.</td></tr>
<tr><td>FALSE</td><td>The FALSE status can have the following meanings:
1. There is no active read/write request.
2. The read/write request has not been completed yet.
3. An error has occurred (if bFail = TRUE).</td></tr>
<tr><td rowspan="2">bFail
BOOL</td><td colspan="2">"Error" status</td></tr>
<tr><td>TRUE</td><td>An error has occurred (group signal).
• See display parameter (C01092) for details.</td></tr>
<tr><td>wOutHWord
wOutLWord
WORD</td><td colspan="2">Value which was read (DataHigh/DataLow portion) after read request</td></tr>
</table>

Parameters

<table>
<tr><th>Parameters</th><th colspan="3">Possible settings</th><th>Information</th></tr>
<tr><td rowspan="2">C01090/1...6</td><td>0.000</td><td></td><td>16000,000</td><td rowspan="2">Parameter to be read or written.
• For a setting of "0,000", inputs wParIndex and wParSubindex are effective for addressing purposes instead.
• Lenze setting: 0.000</td></tr>
<tr><td colspan="3">Format:
<Code number>,<subcode number></td></tr>
<tr><td rowspan="13">C01091/1...6</td><td colspan="3"></td><td rowspan="13">Cycle time
• Subcode 1 = LS_ParReadWrite_1
Subcode 2 = LS_ParReadWrite_2
...
Subcode 6 = LS_ParReadWrite_6
• Lenze setting: 0</td></tr>
<tr><td>0</td><td colspan="2">One-time reading/writing at bExecute in case of a FALSE/TRUE edge</td></tr>
<tr><td colspan="3">Cyclic reading/writing:</td></tr>
<tr><td>20</td><td colspan="2">20 ms</td></tr>
<tr><td>50</td><td colspan="2">50 ms</td></tr>
<tr><td>100</td><td colspan="2">100 ms</td></tr>
<tr><td>200</td><td colspan="2">200 ms</td></tr>
<tr><td>500</td><td colspan="2">500 ms</td></tr>
<tr><td>1000</td><td colspan="2">1 s</td></tr>
<tr><td>2000</td><td colspan="2">2 s</td></tr>
<tr><td>5000</td><td colspan="2">5 s</td></tr>
<tr><td>10000</td><td colspan="2">10 s</td></tr>
</table>

Parameters	Possible settings			Information
C01092/1...6				Error status • If *bFail* = TRUE: Error status is displayed. • Subcode 1 = **LS_ParReadWrite_1** Subcode 2 = **LS_ParReadWrite_2** ... Subcode 6 = **LS_ParReadWrite_6**
	0	No error		
	33803	Invalid data type (e.g. STRING)		
	33804	Limit violation		
	33806	Invalid code		
	33813	No element of the selection list		
	33815	Writing of the parameter not permitted		
	33816	Writing of the parameter only permitted if controller is inhibited		
	33829	Invalid subcode		
	33865	No parameter with subcodes		
C01093/1...6				Arithmetic mode • Lenze setting:"0: No arithmetic" ▸ Arithmetic function
	0	No arithmetic		
	1	In16Bit: LW=+/-32767		
	2	In16Bit: HW=+/-; LW=0..65535		
	3	In32Bit: HW_LW=+/-2147483647		
C01094/1...6	-32767		32767	Meters • For internal conversion in arithmetic modes 1 ... 3. • Lenze setting: 1
C01095/1...6	1		32767	Denominator • For internal conversion in arithmetic modes 1 ... 3. • Lenze setting: 1
C01098/1...6	1		32767	Configuration

19.2.34.1 Arithmetic function

The implemented arithmetic function enables easy arithmetic conversion of the process values to be written or read via parameterisable factors into the format of the target parameter without the need of an additional arithmetic function block.

- In C01093, the interpretation of the *wInHWord* and *wInLWord* inputs can be set to be able to write to parameters:

Arithmetic mode		wInHWord	wInLWord	Internal conversion
0	No arithmetic (Lenze setting)	**INTEGER_32** (4 bytes with sign)		No (same behaviour as before)
		DataHigh portion	DataLow portion	
1	In16Bit: LW=+/-32767	-	**INTEGER_16** (2 bytes with sign)	Yes (see the following section)
2	In16Bit: HW=+/-; LW=0..65535	**Sign** (0 ≡ positive value)	**UNSIGNED_16** (2 bytes without sign)	
3	In32Bit: HW_LW= +/-2147483647	**INTEGER_32** (4 bytes with sign)		
		DataHigh portion	DataLow portion	

Internal conversion

If arithmetic modes 1 ... 3 are selected in C01093, the input value / read parameter value is internally converted via parameterisable factors.

- Division is not remainder considered.

$$\text{parameter value to be written} = \text{Input value}_{[32]} \cdot \frac{\text{Meters}_{[16]}}{\text{Denominator}_{[16]}}$$

C01094: Numerator
C01095: Denominator

[19-85] Internal conversion with write access

$$\text{Output value}_{[32]} = \text{Read parameter value} \cdot \frac{\text{Meters}_{[16]}}{\text{Denominator}_{[16]}}$$

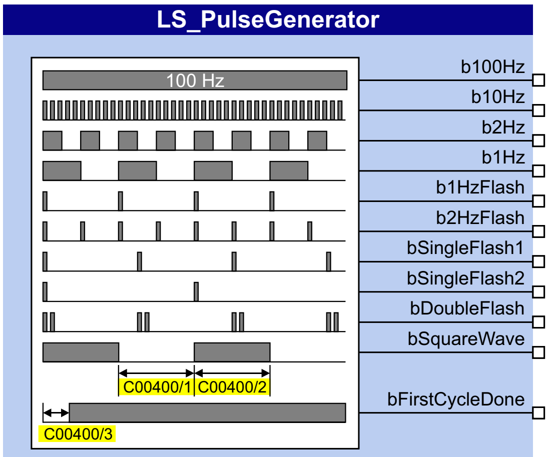
C01094: Numerator
C01095: Denominator

[19-86] Internal conversion with read access

19.2.35 LS_PulseGenerator

This system block outputs 9 different fixed frequencies and 1 frequency with parameterisable pulse/dead time.

The SB provides a TRUE signal at the *bFirstCycleDone* output when the first 1-ms processing cycle is completed and the time set in 1C00400/3 has expired. This status signal can e.g. be used for the delayed enable of peripheral devices or motor control setpoints so that all required initial values are calculated first after the inverter switch-on.



outputs

Designator Data type	Value/meaning
b100Hz BOOL	Rectangular signal 100 Hz
b10Hz BOOL	Rectangular signal 10 Hz
b2Hz BOOL	Rectangular signal 2 Hz
b1Hz BOOL	Rectangular signal 1 Hz
b1HzFlash BOOL	80 ms-pulse, repetition rate every second
b2HzFlash BOOL	80 ms-pulse, repetition rate every 0.5 seconds
bSingleFlash1 BOOL	80 ms pulse, repetition rate every 1.25 seconds
bSingleFlash2 BOOL	80 ms pulse, repetition rate every 3 seconds
bDoubleFlash BOOL	80 ms-double pulse, repetition rate every 1.25 seconds
bSquareWave BOOL	Output frequency with pulse/dead time set in C00400/1...2
bFirstCycleDone BOOL	Status signal "First processing cycle completed" TRUE: The first 1-ms processing cycle has been completed and the time set in C00400/3 has expired (i.e. all FBs have been called at least once).

Parameters

Note!

The real length of the low or high level at the *bSquareWave* output is always 1 ms higher than set in C00400/1 or C00400/2. If "0 ms" is set, a level with a length of 1 ms is created length, for instance, and if "1000 ms" is set, a level with a length of 1001 ms is created.

Parameters	Possible settings			Information
C00400/1	0	ms	60000	Length of the low level (break) + 1 ms • For output *bSquareWave* • Lenze setting: 1000 ms
C00400/2	0	ms	60000	Length of the high level + 1 ms • For output *bSquareWave* • Lenze setting: 1000 ms
C00400/3	0	ms	60000	Delay of the status *bFirstCycleDone* • Lenze setting: 100 ms

19.2.36 LS_Resolver

Interface to the resolver.

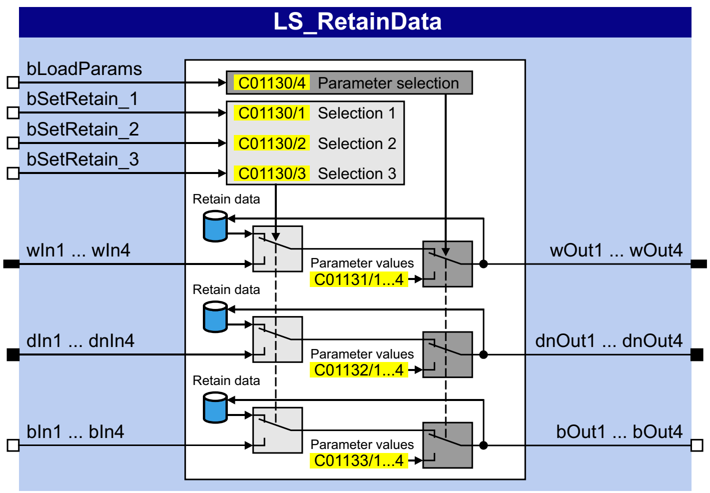
For detailed information please see the main chapter "Encoder/Feedback system":

▸ Internal interfaces | "LS_Resolver" system block (📖 340)

19.2.37 LS_RetainData

This system block serves to select and save retain data similar to the FBs L_SampleHold_1 and L_SampleHold_2.

- Retain data are values that are saved automatically at mains disconnection or disconnection of the 24-V supply in the inverter and are restored when the 24-V supply is switched on again.
- A total of 12 retain data can be stored in the retain memory:
  - 4 WORD signals (16 bits, 0 ... 65535)
  - 4 DINT signals (32 bits, -2147483647 ... 2147483647)
  - 4 BOOL signals (FALSE or TRUE)



inputs

Designator Data type		**Information/possible settings**
bLoadParams BOOL		Set selected outputs & retain values to parameter values. • This input has priority over the *bSetRetain* inputs. • Typical application: Initialisation of the outputs & retain values.
	FALSE	The input or retain values are output, depending on the *bSetRetain* input.
	TRUE	The outputs selected in C01130/4 and the respective retain values are set to the parameter values and held as long as *bLoadParams* is TRUE. If the parameter values change during this time, the respective outputs and retain values change as well.
bSetRetain_1...3 BOOL		Transfer selected input values to the retain memory ("latch")
	FALSE	No transfer. Retain values or parameter values (when *bLoadParams* = TRUE) are provided at the outputs.
	TRUE	The input values selected in C01130/1...3 are stored in the retain memory and provided at the outputs.
wIn1 ... wIn4 WORD		Input values
dnIn1 ... dnIn4 DINT		
bIn1 ... bIn4 BOOL		

outputs

Designator Data type	Value/meaning
wOut1 ... wOut4 WORD	Output of the input or retain values, depending on the *bSetRetain* input
dnOut1 ... dnOut4 DINT	
bOut1 ... bOut4 BOOL	

Parameters

Parameters	Possible settings			Information
C01130/1...4	Setting is bit coded:			Subcodes 1 ... 3: Selection 1 ... 3 • Select which input values are to be stored in the retain memory when setting the respective *bSetRetain*-input to TRUE. • Lenze setting: 0x0000 Subcode 4: Parameter selection • Select which parameter values are to be stored in the retain memory when setting *bLoadParams* to TRUE. • Lenze setting: 0x0000
		Subcodes 1 ... 3:	Subcode 4:	
	Bit 0	wIn1	C01131/1	
	Bit 1	wIn2	C01131/2	
	Bit 2	wIn3	C01131/3	
	Bit 3	wIn4	C01131/4	
	Bit 4	dnIn1	C01132/1	
	Bit 5	dnIn2	C01132/2	
	Bit 6	dnIn3	C01132/3	
	Bit 7	dnIn4	C01132/4	
	Bit 8	bIn1	C01133/1	
	Bit 9	bIn2	C01133/2	
	Bit 10	bIn3	C01133/3	
	Bit 11	bIn4	C01133/4	
	Bit 12	Reserved		
	Bit 13			
	Bit 14			
	Bit 15			
C01131/1...4	0		65536	Parameter value for retain memory (outputs *wOut1 ... wOut4*) • Lenze setting: 0
C01132/1...4	-2147483647		2147483647	Parameter values for retain memory (outputs *dnOut1 ... dnOut4*) • Lenze setting: 0
C01133/1...4				Parameter values for retain memory (outputs *bOut1 ... bOut4*) • Lenze setting: 0: FALSE
	0	FALSE		
	1	TRUE		

19.2.38 LS_SetError_1

Parameterisable responses to user-defined events are tripped.

For a detailed description see the main chapter "Diagnostics & error management":
- "LS_SetError_1" system block (782)

19.2.39 LS_SetError_2

Parameterisable responses to user-defined events are tripped.

For a detailed description see the main chapter "Diagnostics & error management":
- "LS_SetError_2" system block (783)

19.2.40 LS_SyncManagement

Output of status information for synchronising the internal time base.

For a detailed description see main chapter "Synchronisation of the internal time base of the inverter":
- Internal interfaces | System block "LS_SyncManagement" (913)

19.2.41 LS_TouchProbe

Interface for touch probe detection.

For a detailed description see the main chapter "I/O terminals":
- Touch probe detection (435)

19.2.42 LS_WriteParamList

Writing to a configurable list which contains up to 32 local parameters.

For a detailed description see the main chapterl "Parameter change-over". (914)

19.3 Application blocks

19.3.1 Technology application "Actuating drive speed"

This technology application serves to solve speed-controlled drive tasks, e.g. conveyor drives (interconnected), extruders, test benches, vibrators, travelling drives, presses, machining systems, metering units.

For a detailed description see TA "Actuating drive speed". (📖 454)

19.3.2 Technology application "actuating drive speed (AC Drive profile)"

This technology application available from version 13.00.00 provides a speed and torque control by means of "AC Drive Profile". The fieldbuses EtherNet/IP™ and system bus (CANopen) are supported.

For a detailed description see TA "actuating drive speed (AC Drive Profile)". (📖 481)

19.3.3 Technology application "Table positioning"

This technology application serves to solve position-controlled drive tasks which are normally controlled by a higher-level control via a fieldbus, e.g. transport facilities, rotary tables, storage and retrieval units, feed drives, metering units, hoists.

For a detailed description see TA "Table positioning". (📖 514)

19.3.4 "Switch-off positioning" technology application

This technology application is used to solve speed-controlled drive tasks which require a pre-switch off or stopping at certain positions, e.g. roller conveyors and conveying belts. The pre-switch off is implemented by connecting switch-off sensors.

For a detailed description see TA "Switch-off positioning". (📖 544)

19.4 Port blocks

19.4.1 Port block "LP_CanIn1"

The **LP_CanIn1** port block maps process data object RPDO1 in the FB Editor.

For a detailed description see RPDO1 | Port block "LP_CanIn1". (🕮 821)

19.4.2 Port block "LP_CanIn2"

The **LP_CanIn2** port block maps process data object RPDO2 in the FB Editor.

For a detailed description see RPDO2 | "LP_CanIn2" port block. (🕮 823)

19.4.3 Port block "LP_CanIn3"

The **LP_CanIn3** port block maps process data object RPDO3 in the FB Editor.

For a detailed description see RPDO3 | "LP_CanIn3" port block. (🕮 825)

19.4.4 Port block "LP_CanIn4"

The **LP_CanIn4** port block maps process data object RPDO4 in the FB Editor.

For a detailed description see RPDO4 | "LP_CanIn4" port block. (🕮 827)

19.4.5 Port block "LP_CanOut1"

The **LP_CanOut1** port block maps process data object TPDO1 in the FB Editor.

For a detailed description see TPDO1 | "LP_CanOut1" port block. (🕮 829)

19.4.6 Port block "LP_MciIn"

The **LP_MciIn** port block maps the received MCI-PDOs in the FB Editor.

For a detailed description see Port block "LP_MciOut". (📖 889)

19.4.7 Port block "LP_MciOut"

The **LP_MciOut** port block maps the sent MCI-PDOs in the FB Editor.

For a detailed description see Port block "LP_MciIn". (📖 888)

19.4.8 Port block "LP_CanOut2"

The **LP_CanOut2** port block maps process data object TPDO2 in the FB Editor.

For a detailed description see TPDO2 | "LP_CanOut2" port block. (📖 830)

19.4.9 Port block "LP_CanOut3"

The **LP_CanOut3** port block maps process data object TPDO3 in the FB Editor.

For a detailed description see TPDO3 | "LP_CanOut3" port block. (📖 831)

19.4.10 Port block "LP_CanOut4"

The **LP_CanOut4** port block maps process data object TPDO4 in the FB Editor.

For a detailed description see TPDO4 | "LP_CanOut4" port block. (📖 832)

E

M

S

T

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

Lenze Drives GmbH
Postfach 10 13 52, D-31763 Hameln
Breslauer Straße 3, D-32699 Extertal
Germany
HR Lemgo B 6478
+49 5154 82-0
+49 5154 82-2800
sales.de@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service.de@lenze.com

Inverter

8400

"Actuator Torque" technology application
for 8400 HighLine C and 8400 TopLine C _ _ _

Software Manual EN

13460495

Lenze

Contents

1 About this documentation

This documentation described the software-based solution of a task. The transferability of the described solution to the respective application case needs to be checked by the user. If required, the user has to adapt the solution accordingly. Thus, physical aspects as e.g. drive dimensioning is not part of this documentation.

Danger!

The controller is a source of danger which may lead to death or severe injury of persons.

To protect yourself and others against these dangers, observe the safety instructions before switching on the controller.

Please read the safety instructions provided in the **8400 mounting instructions** and in the **8400 hardware manual**. Both documents are supplied with the controller.

Target group

This documentation is intended for all persons who want to use the "Actuator Torque" technology application for the 8400 HighLine/TopLine controller.

Validity

The information in this documentation are valid for the following technology applications:

Technology application	from version
Actuator Torque	01.00

Screenshots/application examples

All screenshots provided in this documentation are application examples. Depending on the software version of the controller and the version of the installed »Engineer« software, the screenshots in this documentation may differ from the representation in the »Engineer«.

Tip!

Information and tools for Lenze products are provided in the download area at

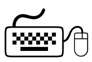
http://www.lenze.com → Download

1.1 Document history

Version			Description
1.1	07/2014	TD05	Error corrections
1.0	03/2014	TD05	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is generally used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the prefix "0x" is used. Example: 0x60F4
Binary number	0b	For binary numbers, the prefix "0b" is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that only applies to a certain controller software version or higher is identified accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software ...
Window	*italics*	The *Message* window ... / The *Options* dialog box...
Variable name		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Press **<F1>** to open the online help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...
Hyperlink	underlined	Optically highlighted reference to another topic. In this documentation activated by mouse-click.
Icons		
Page reference	(📖 4)	Optically highlighted reference to another page. In this documentation activated by mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Engineering Tools	Software solutions for simple engineering at all stages
	»EASY Navigator« – Ensures easy operator guidance • All practical Lenze engineering tools at a glance • Tools can be selected quickly • Clearly arranged, simplifying the engineering process from the start
	»EASY Starter« – Simple tool for service technicians • Especially developed for the commissioning and maintenance of Lenze devices • Graphical user interface with few buttons • Simple online diagnostics, parameterisation and commissioning • No risk of accidentally changing the application • Ready applications can be loaded to the device
	»Engineer« – Multi-device engineering • For all products from our L-force portfolio • Practice-oriented user interface • Easy handling due to graphical user interfaces • Suitable for all project stages (configuration, commissioning, production) • Parameter setting and configuration
Code	Parameter used for controller parameterisation or monitoring. The term is usually called "index".
Display code	Parameter that displays the current status or value of an input/output of a system block.
FB Editor	Abbreviation for function block editor. Graphic interconnection tool which is available in the »Engineer« for function block interconnections on the **FB Editor**.
Function block	General designation of a function block for free interconnection in the FB Editor. A function block (short: FB) can be compared with an integrated circuit that contains a specific control logic and delivers one or several values when being executed. Example: "L_Arithmetik_1" (FB for arithmetic operations)
Lenze setting	This setting is the default factory setting of the device.
LP	Abbreviation for **L**enze **P**ort block Example: "LP_CanIn1" (CAN1 port block)
LS	Abbreviation for **L**enze **S**ystem block Example: "LS_DigitalInput" (system block for digital input signals)
MCI	Abbreviation for **M**otionbus **C**ommunication **I**nterface (fieldbus interface) The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system.
Port block	Block for implementing the process data transfer via a fieldbus
Subcode	If a code contains several parameters, these are stored in "subcodes". This Manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). The term is usually called "subindex".
System block	System blocks provide interfaces to basic functions and to the hardware of the controller in the FB Editor of the »Engineer« (e.g. to the digital inputs).
USB diagnostic adapter	The USB diagnostic adapter is used for the operation, parameterisation, and diagnostics of the controller. Data are exchanged between the PC (USB connection) and the controller (diagnostic interface on the front) via the diagnostic adapter. Order designation: E94AZCUS

1.4 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

> **Pictograph and signal word!**
>
> (characterise the type and severity of danger)
>
> **Note**
>
> (describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling

2 Features of the technology application

The "Actuator Torque" technology application serves to set a selectable torque at the motor independent of the speed ("torque-controlled operation"). A superimposed speed limitation prevents the drive from accelerating in an uncontrolled way. If the positive or negative speed limit value (for CW and CCW rotation) is reached, the drive changes from the torque-controlled to the speed-controlled operation. The set speed limit values will not be exceeded. The torque setpoint and the speed limit value are the main setpoint values of the application.

The setpoint for the motor torque can be specified for both directions. Hence, the drive has a driving or braking effect. In case of a unipolar setpoint, the effective direction can be switched over via a boolean signal.

2.1 Functional overview

Function	8400 HighLine	8400 TopLine
Two modes for motor control switchable via a boolean signal: • Torque control with speed limitation • Speed control with torque limitation	●	●
Offset, gain and ramp generator for the torque setpoint	●	●
Inching mode with ramp generator: • Hard" inching: Normal speed setting without torque limitation • "Smooth" inching: Operated with speed limitation in torque-controlled mode	●	●
Signal processing functions for speed limitations in torque-controlled mode or the speed setpoint in case of speed control: • Ramp function generator with linear ramps or S-shaped ramp (PT1 rounding) • Fixed setpoints (JOG setpoints)	●	●
Offset and gain for speed limitations	●	●
Forward motion for speed setpoint in torque-controlled mode	●	●
Inversion of the effective direction for the torque setpoint via a boolean signal	●	●
Web break/slip monitoring	●	●

2.2 Application ranges

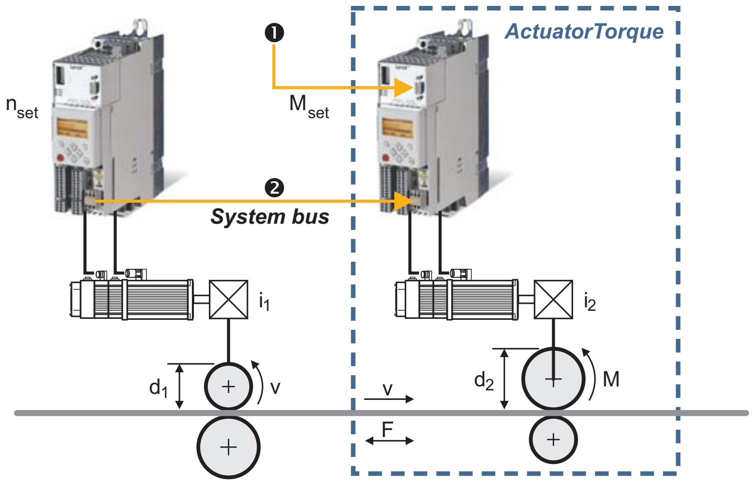
- Material transport with torque distribution (supporting drives)
- Conveyor drives with adjustable tension
- Chain conveyors
- S frame structure
- Bilateral tandem drives
- Test benches for tensile stress
- Test benches for motors and gearboxes
- Brake assemblies
- Traction-controlled winders with superimposed setpoint conditioning (e.g. diameter calculator in the PLC)

Example: Conveyor drive with adjustable web tension

This application can be directly implemented using the "Actuator Torque" technology application.

The torque setpoint results from a setpoint tensile force for the web. This setpoint is defined as process date, e.g. by a control or an HMI. The value of the speed limitation derives from the actual speed of the speed-determining drive of the total machine plus a forward motion or offset value.

Conveyor drive with adjustable web tension



❶	Setpoint tensile force for web = torque setpoint for torque follower (right drive)
❷	Actual speed for left drive = speed limitation plus forward motion/offset value for the drive on the right

2.3 System requirements

The technology application was created with the L-force »Engineer« V2.19 and can only be used with the versions V2.19 or higher.

Software

Product	Order designation	from version
L-force »Engineer« HighLevel	ESPEV-EHNNN	2.19

Hardware

Product	Order designation	from hardware version	from software version
Inverter Drives 8400 HighLine C	E84AVHCxxxxx	VD	13.00
Inverter Drives 8400 TopLine C	E84AVTCxxxxx	VD	13.00

2.4 Influencing factors of the torque setting range

Typically, the torque setting range of the drive is in the range 1:20 to 1:40. The following factors influence the setting range in a positive way:

- Motor model adjusted in the best possible way by entering extended motor type data or by the motor parameter identification.
- Compensation of the change in motor resistance as a function of the winding temperature. For this purpose, a temperature feedback (KTY) is required via the motor encoder cable. An evaluation of KTY sensors for temperature detection is only possible for the 8400 TopLine via the encoder interface X7 and X8.
- In case of permanent-magnet synchronous motors: Compensation of the change in flux formation as a function of the winding temperature. An evaluation of KTY sensors for temperature detection is only possible for the 8400 TopLine via the encoder interface X7 and X8.

STOP

Stop!

Torque setting without motor feedback

In principle, a torque can also be set for asynchronous motors using the sensorless vector control (SLVC). Theoretically, this could serve to also use motors without speed feedback in applications with reduced torque setting range requirements. However, the following aspects have to be considered here:

- The sensorless vector control does not enable the torque to be controlled in a stable way in the range of zero field frequency.
- An operation in generator mode with low field frequencies shall be prevented when using sensorless vector control as the motor cannot be controlled then.
- The field frequency has to amount to double the rated motor slip in order to set a torque in a stable way.

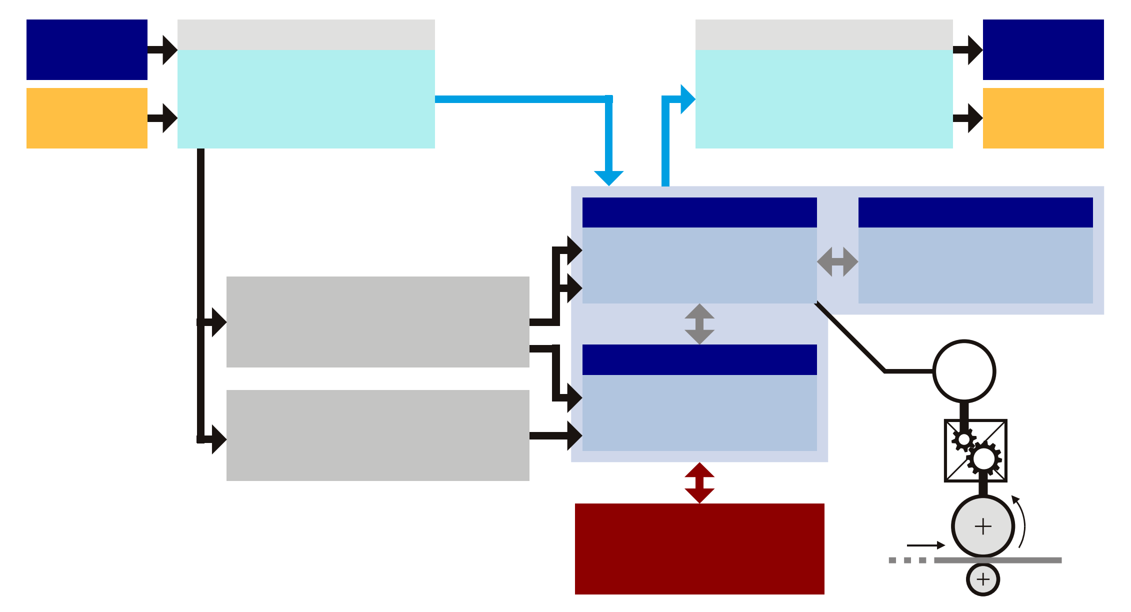
The V/f characteristic control (VFCplus) is not suitable to set a torque.

2.5 Basic signal flow

In the technology application, function and system blocks are interconnected such that typical basic functions of the torque controllers can be implemented from the application ranges mentioned before.

The following describes the principal signal flow with the essential functions.

Principal signal flow of the "Actuator Torque" technology application



2.6 Parameter setting in the FB Editor view

You can make the settings of the application-specific parameters directly in the FB Editor. This has the advantage that the signal flow can be traced. The interaction of the modules becomes clear. Moreover, you can reconfigure the I/O interconnection using the FB Editor and carry out an online monitoring of the application running in the device (e.g. for diagnostic purposes).

- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.
- Colour codes and comments support you in handling the FB Editor.
  - The areas highlighted in turquoise represent the "user interface". If required, the pre-assignment of the I/O terminals can be adapted here and a control via the fieldbus interface (MCI) can be established.
  - In the areas highlighted in yellow, application-specific settings are required.

Detailed information on how to work with the FB Editor can be found in the reference manual/online help of the controller in the chapter "Working with the FB Editor".

2.7 Pre-assignment of the I/O terminals

<table>
<tr><th>Terminal</th><th colspan="3">Function</th></tr>
<tr><td colspan="4">Digital input terminals</td></tr>
<tr><td>X5/RFR</td><td colspan="3">Controller enable</td></tr>
<tr><td>X5/DI1</td><td colspan="3">- (reserved for HTL encoder)</td></tr>
<tr><td>X5/DI2</td><td colspan="3">- (reserved for HTL encoder)</td></tr>
<tr><td rowspan="4">X5/DI3</td><td colspan="3">Switch on torque-controlled operation</td></tr>
<tr><td colspan="2">DI3</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Speed control with torque limitation</td></tr>
<tr><td colspan="2">HIGH</td><td>Torque control with speed limitation</td></tr>
<tr><td rowspan="4">X5/DI4</td><td colspan="3">Activate quick stop</td></tr>
<tr><td colspan="2">DI4</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Deactivate quick stop</td></tr>
<tr><td colspan="2">HIGH</td><td>Activate quick stop
• The motor control is decoupled from the setpoint selection (speed and torque) and within the deceleration time parameterised in C00105, the motor is brought to standstill (n_{act} = 0).
• A pulse inhibit is set if the auto-DCB function has been activated via C00019.</td></tr>
<tr><td rowspan="6">X5/DI5
X5/DI6</td><td colspan="3">Selection of fixed setpoints for inching mode
• For "soft inching", DI3 has to be set to HIGH.</td></tr>
<tr><td>DI5</td><td>DI6</td><td>Function</td></tr>
<tr><td>LOW</td><td>LOW</td><td>Speed selection via analog input 2</td></tr>
<tr><td>HIGH</td><td>LOW</td><td>Selection of fixed setpoint 1 = C00039/1 = 10 % reference speed (C00011)
→ Manual inching in positive direction</td></tr>
<tr><td>LOW</td><td>HIGH</td><td>Selection of fixed setpoint 2 = C00039/2 = -10 % reference speed (C00011)
→ Manual inching in negative direction</td></tr>
<tr><td>HIGH</td><td>HIGH</td><td>Selection of fixed setpoint 3 = C00039/3 = 0 % reference speed (C00011)</td></tr>
<tr><td>X5/DI7</td><td colspan="3">Reset error message</td></tr>
<tr><td colspan="4">Analog input terminals</td></tr>
<tr><td>X3/A1U</td><td colspan="3">Torque setpoint
• Scaling: 10 V ≡ 100 % maximum torque (C00057)</td></tr>
<tr><td>X3/A2U</td><td colspan="3">Speed setpoint/speed limitations (in torque-controlled operation)
• Scaling: 10 V ≡ 100 % reference speed (C00011)</td></tr>
<tr><td colspan="4">Digital output terminals</td></tr>
<tr><td>X4/DO1</td><td colspan="3">Status "Drive is ready"</td></tr>
<tr><td>X4/DO2</td><td colspan="3">- (not assigned, can be used freely)</td></tr>
<tr><td>X4/DO3</td><td colspan="3">- (not assigned, can be used freely)</td></tr>
<tr><td>X107/BD1, BD2</td><td colspan="3">Control of a holding brake by the basic function "holding brake control"</td></tr>
<tr><td>X101/COM, NO</td><td colspan="3">Status "Error is pending"</td></tr>
<tr><td colspan="4">Analog output terminals</td></tr>
<tr><td>X3/O1U</td><td colspan="3">Actual speed value
• Scaling: 10 V ≡ 100 % reference speed (C00011)</td></tr>
<tr><td>X3/O2U</td><td colspan="3">Actual torque
• Scaling: 10 V ≡ 100 % maximum torque (C00057)</td></tr>
</table>

3 Short setup of the technology application

3.1 Preconditions

For the execution of the short setup described in the following, the setting of the most important parameters (motor, feedback system, etc.) is assumed.

The "commissioning wizard 8400" serves to carry out a guided commissioning of the controller based on the Lenze setting of the parameters.

How to proceed:

1. Before switch-on: Make sure that the controller is inhibited (input RFR open).
2. Switch on voltage supply of the controller.
   For parameter setting and diagnostics of the controller without motor operation, an external 24-V supply through a safely separated power supply unit (SELV/PELV) is sufficient.
3. Establish a communication link between controller and Engineering PC, e.g. via USB diagnostic adapter (E94AZCUS):
   - connect the USB diagnostic adapter to the X6 diagnostic interface.
   - establish a connection between the USB diagnostic adapter and the PC via a free USB port.
4. Start »Engineer« on the Engineering PC, e.g. via the Windows® start menu:
   Start → All programs → Lenze → Engineering → L-force Engineer...
   After the program start, no project has been loaded first and the *start-up wizard* is displayed.

> You can find detailed information on the options of the start-up wizard and on the general use of the »Engineer« in the online help for the program which you can call with **[F1]**.

5. Create a new project or open a project already available.
6. Go to *Project View* and select the 8400 controller.
7. Click the icon to go online.
   After a connection to the controller has been established, the following status is displayed in the *Status line*:

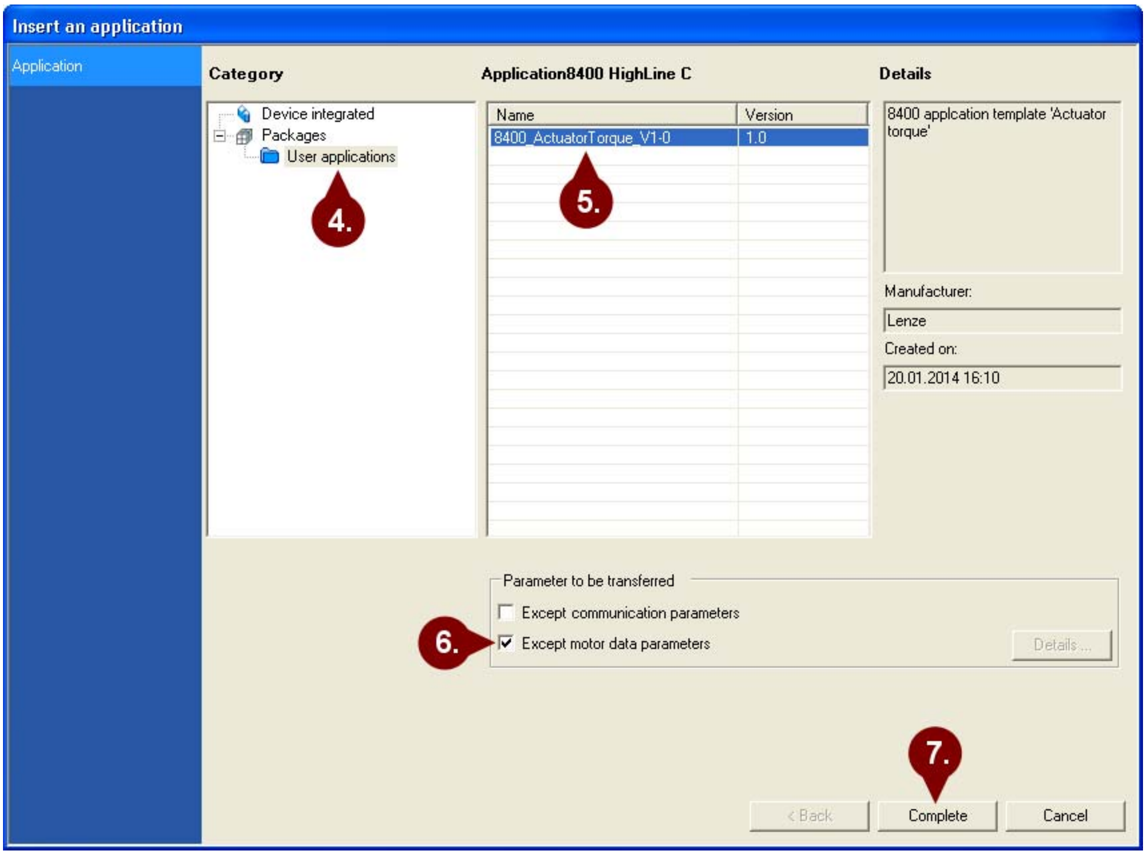
8. Click the icon to start the *commissioning wizard 8400*.
   - Now the commissioning wizard guides you step by step through the setting of the important parameters for a quick commissioning.
   - The **Next** button can only be activated again after all parameter settings in the device have been reset via the **Load Lenze setting** button.

3.2 Step 1: Load "Actuator Torque" technology application

In the Lenze setting, the controller uses the "speed actuating drive" technology application integrated in the device. Execute the following steps to use the "Actuator Torque" technology application instead:

1. Select the controller in the *Project view*.
2. If there is still an online connection to the controller:

   Click the icon to go offline again.
   (the application can only be selected offline.)
3. Click the icon to select another application.
   The *Insert application* dialog box appears:



4. In the left field, select the category "Packages" ➜ "User applications".
5. In the right field, select the "8400_ActuatorTorque_V1-0" application.
6. Activate the **Except motor data parameters** option in order that the settings of the motor data parameters made before will not be overwritten.
7. Press **Complete** to close the dialog box again and load the selected application into the »Engineer« project.
8. Confirm the request whether the current application is to be replaced by the "8400_ActuatorTorque_V1-0" application with **Yes**.

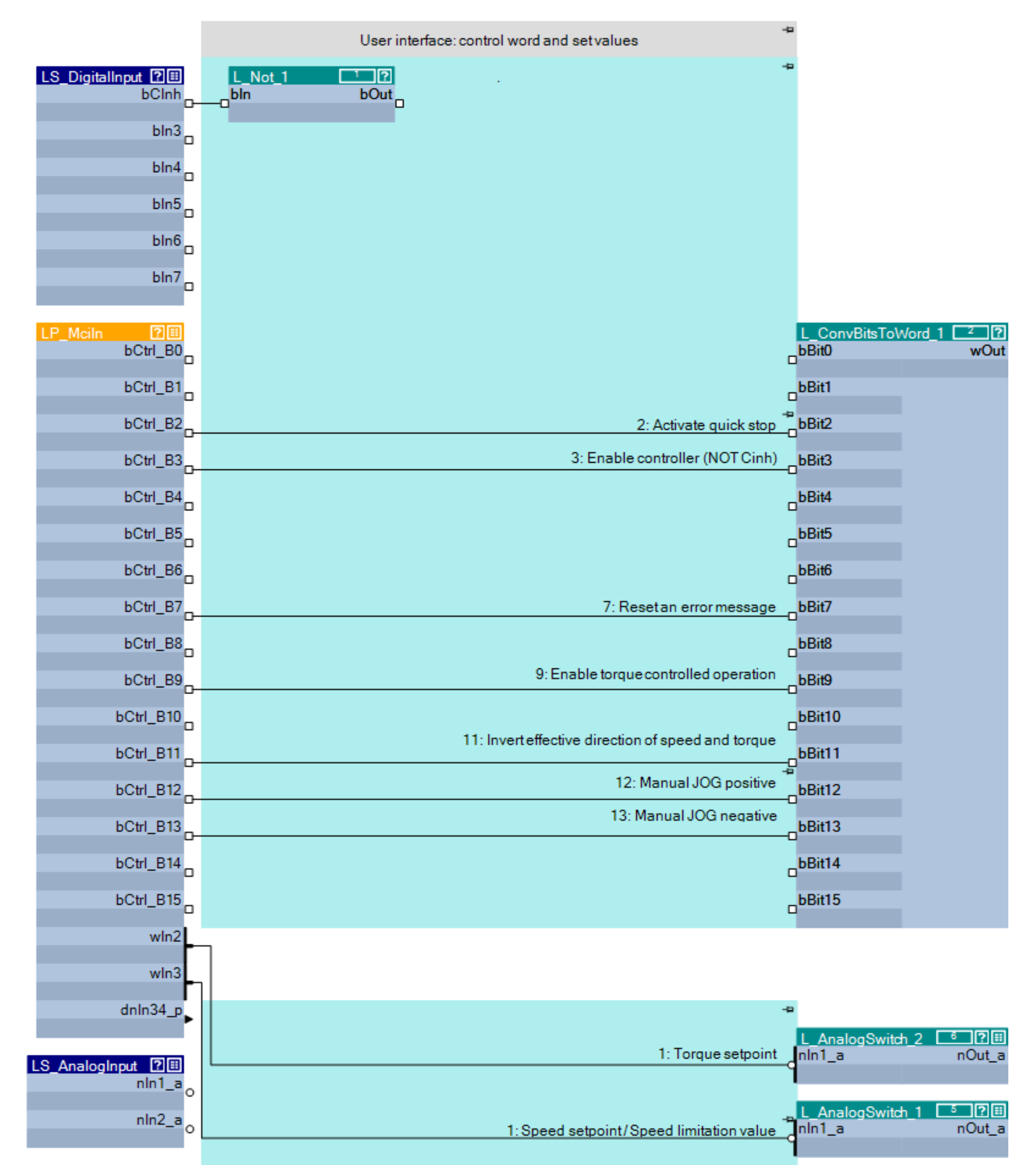
3.3 Step 2 (optional): Establish control via the fieldbus interface (MCI)

In the default setting, the setpoints are selected via the analog input terminals and the application is controlled via the digital input terminals. For the control via the fieldbus interface (MCI), the user interface (highlighted in turquoise) in the FB Editor has to be adapted for the application control and the setpoints.

- The assignment of the outputs on the left to the inputs on the right can be changed at will.
- The inputs on the right are permanently linked to functions of the application.

The following sample illustration shows the interconnection required for achieving the Pre-assignment of the process data input words described in the following subchapter:

Adaptation of the user interface for a control via the fieldbus interface (MCI)



[3-1] User interface for application control word and setpoints in the FB Editor

3.3.1 Pre-assignment of the process data input words

After adapting the interconnection according to the [3-1] illustration, the process data input words for a control via the fieldbus interface (MCI) are assigned as follows:

Input words	Assignment
Word 1	Control word (for bit assignment see the following table)
Word 2	Torque setpoint • Scaling: 16384 ≡ 100 % maximum torque (C00057)
Word 3	Speed setpoint/speed limitations (in torque-controlled operation) • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 4 ... 16	- (not preconfigured)

<table>
<tr><th>Control word</th><th colspan="3">Function</th></tr>
<tr><td>Bit 0</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 1</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 2</td><td colspan="3">1 ≡ Activate quick stop (QSP)</td></tr>
<tr><td>Bit 3</td><td colspan="3">1 ≡ Enable controller (RFR)</td></tr>
<tr><td>Bit 4</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 5</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 6</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 7</td><td colspan="3">1 ≡ Reset fault (trip reset)</td></tr>
<tr><td>Bit 8</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 9</td><td colspan="3">0 ≡ Speed control with torque limitation
1 ≡ Torque control with speed limitation (torque-controlled operation)</td></tr>
<tr><td>Bit 10</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 11</td><td colspan="3">1 ≡ Invert effective direction
• Inversion of the torque setpoint, the speed limitations and the parameterised fixed setpoints.</td></tr>
<tr><td rowspan="6">Bit 12 ... 13</td><td colspan="3">Selection of fixed setpoints for inching mode
• For "soft inching", bit 9 has to be set to "1" in addition.</td></tr>
<tr><th>Bit 12</th><th>Bit 13</th><th>Function</th></tr>
<tr><td>0</td><td>0</td><td>Speed selection via process data word 3
(with a correspondingly adapted interconnection, see above)</td></tr>
<tr><td>1</td><td>0</td><td>Selection of fixed setpoint 1 = C00039/1 = 10 % reference speed (C00011)
➜ Manual inching in positive direction</td></tr>
<tr><td>0</td><td>1</td><td>Selection of fixed setpoint 2 = C00039/2 = -10 % reference speed (C00011)
➜ Manual inching in negative direction</td></tr>
<tr><td>1</td><td>1</td><td>Selection of fixed setpoint 3 = C00039/3 = 0 % reference speed (C00011)</td></tr>
<tr><td>Bit 14</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 15</td><td colspan="3">- (not preconfigured)</td></tr>
</table>

3.3.2 Pre-assignment of the process data output words

The **LP_MciOut** port block is already implemented in the technology application. The process data output words are assigned as follows:

Output words	Assignment
Word 1	Status word (for bit assignment see the following table)
Word 2	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	Actual torque • Scaling: 16384 ≡ 100 % maximum torque (C00057)
Word 4 ... 16	- (not preconfigured)

<table>
<tr><th>Status word</th><th colspan="6">Status</th></tr>
<tr><td>Bit 0</td><td colspan="6">1 ≡ Group error active (configurable in C00148)</td></tr>
<tr><td>Bit 1</td><td colspan="6">1 ≡ Inverter control inhibited (pulse inhibit is active)</td></tr>
<tr><td>Bit 2</td><td colspan="6">1 ≡ Drive controller is ready for operation</td></tr>
<tr><td>Bit 3</td><td colspan="6">1 ≡ Quick stop is active</td></tr>
<tr><td>Bit 4</td><td colspan="6">1 ≡ Setpoint torque is in the limitation</td></tr>
<tr><td>Bit 5</td><td colspan="6">1 ≡ Speed controller is in the limitation</td></tr>
<tr><td rowspan="2">Bit 6</td><td colspan="6">During open-loop operation:
1 ≡ Speed setpoint < Comparison value (C00024)</td></tr>
<tr><td colspan="6">During closed-loop operation:
1 ≡ Actual speed value < Comparison value (C00024)</td></tr>
<tr><td>Bit 7</td><td colspan="6">1 ≡ Controller inhibited (controller inhibit is active)</td></tr>
<tr><td rowspan="14">Bit 8 ... 11</td><td colspan="6">Bit coded display of the active device status</td></tr>
<tr><td>Bit 11</td><td>Bit 10</td><td>Bit 9</td><td>Bit 8</td><td>Device status</td><td>Meaning</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>FirmwareUpdate</td><td>Firmware update function is active</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Init</td><td>Initialisation active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Ident</td><td>Identification active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>ReadyToSwitchOn</td><td>Device is ready to start</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>SwitchedOn</td><td>Device is switched on</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>OperationEnabled</td><td>Operation</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>Trouble</td><td>Trouble active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>Fault</td><td>Fault active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>TroubleQSP</td><td>TroubleQSP is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>SafeTorqueOff</td><td>Safe torque off is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>SystemFault</td><td>System fault active</td></tr>
<tr><td>Bit 12</td><td colspan="6">1 ≡ A warning is indicated</td></tr>
<tr><td>Bit 13</td><td colspan="6">1 ≡ Controller is in the "Trouble" device status</td></tr>
<tr><td>Bit 14</td><td colspan="6">1 ≡ Torque-controlled operation active</td></tr>
<tr><td>Bit 15</td><td colspan="6">1 ≡ Speed limitation active (web break/slip monitoring has tripped)</td></tr>
</table>

3.4 Step 3: Set commissioning parameters

For a quick commissioning, only the following application-specific parameters have to be set or their default setting has to be checked!

- In order that you quickly find the respective parameterisation dialog in the FB Editor, the following table lists the block related to each parameter.
- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

Parameter (Block)	**Possible settings**			**Info**
C00470/1 (LS_ParFree_b)				1 ≡ Invert effective direction • Inversion of the torque setpoint, the speed limitations and the parameterised fixed setpoints. • Lenze setting: 0
	0	Effective direction CW		
	1	Effective direction CCW		
Condition speed setpoint/speed limitations				
C00660/2 (L_FixSet_a_1)	-199.99	%	199.99	Common speed offset value for both speed limitations • Only effective in the torque-controlled operation. • Lenze setting: 5 % reference speed (C00011)
C00039/1 (L_NSet_1)	-199.99	%	199.99	Fixed setpoint 1 • For manual inching in positive direction. • Lenze setting: 10 % reference speed (C00011)
C00039/2 (L_NSet_1)	-199.99	%	199.99	Fixed setpoint 2 • For manual inching in negative direction. • Lenze setting: -10 % reference speed (C00011)
C00039/3 (L_NSet_1)	-199.99	%	199.99	Fixed setpoint 3 • Fixed setpoint 3 is selected if bit 12 **AND** bit 13 of the application control word are set to "1". • Lenze setting: 0 % reference speed (C00011)
Condition torque setpoint				
C00472/1 (LS_ParFree_a)	-199.99	%	199.99	Gain factor for torque setpoint • Lenze setting: 100 %
C01040/1 (L_SRFG_1)	0.001	s	999.999	Linear ramp time for the torque setpoint • Lenze setting: 0.010 s
Creation of the "torque-controlled operation active" status (bit 14 in the status word)				
C00825/12 (L_DigitalLogic5_1)				Influence of the "brake released" status signal on the "torque-controlled operation active" status • Lenze setting: 0 (no influence)
	0	*bMBrakeReleased* is ignored		The "torque-controlled operation active" status is set irrespective of the brake status.
	1	*bMBrakeReleased* is additionally AND'ed		The "torque-controlled operation active" status also requires a released brake.

3.5 Step 4: Go online and transmit parameter set to the controller

In order to set the current parameter settings in the controller to the settings in the project, transmit the parameter set to the controller.

1. Click the icon to go online.
2. Click the icon to transmit the parameter set to the controller.
3. After a successful transmission, click the icon to save the parameter set safe against mains failure in the integrated Memory Module.

3.6 Step 5: Enable controller and select setpoints

After the parameter set has been transmitted to the controller, the controller can now be enabled and the setpoints can be selected via the corresponding interfaces.

▸ Pre-assignment of the I/O terminals (📖 11)
▸ Pre-assignment of the process data input words (📖 15)

3.7 Step 6 (optional): Set optimisation parameters

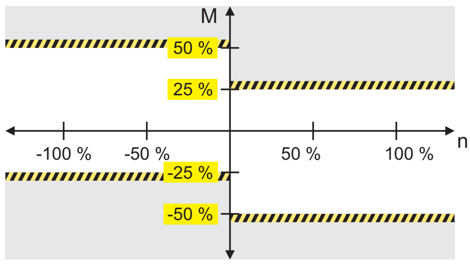
The following application-specific parameters are used for optimisation and can also be adapted during operation.

Stop!

If you change parameters in the »Engineer« during an online connection to the device, the changes are directly transferred to the device!

Tip!

Do not forget to save the executed parameter changes safe against mains failure in the integrated Memory Module!

Parameter (Block)	Possible settings			Info
Torque limitation in motor mode/in generator mode				
The set torque limitation is always effective, even in torque-controlled operation.				Example: Definition of the torque limitations C00472/3 = 25 % C00472/4 = 50 % 
C00472/3 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in motor mode • Lenze setting: 100 % maximum torque (C00057)
C00472/4 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in generator mode • Lenze setting: 100 % maximum torque (C00057)
Condition speed setpoint/speed limitations				
C00721/2 (L_DigitalDelay_2)	0.000	s	3600.000	Delay for the change-over of acceleration/deceleration time for manual jog back to the acceleration/deceleration time for the speed setpoints/speed limitations if manual jog is not requested anymore. The delay has to be set according to the following equation: Delay = fixed setpoint (C39/x) * deceleration time (C103/x) • Example for fixed setpoint 1: Delay = 10 [%] * 1 [s] = 10 [s] • Lenze setting: 0.1 s
C00012 (L_NSet_1)	0.000	s	999.99	Acceleration time for the speed setpoint/speed limitations • Lenze setting: 0 s
C00013 (L_NSet_1)	0.000	s	999.99	Deceleration time for the speed setpoint/speed limitations • Lenze setting: 0 s
C00101/1...3 (L_NSet_1)	0.000	s	999.999	Acceleration times for fixed setpoints 1 ... 3 • Lenze setting: 1 s

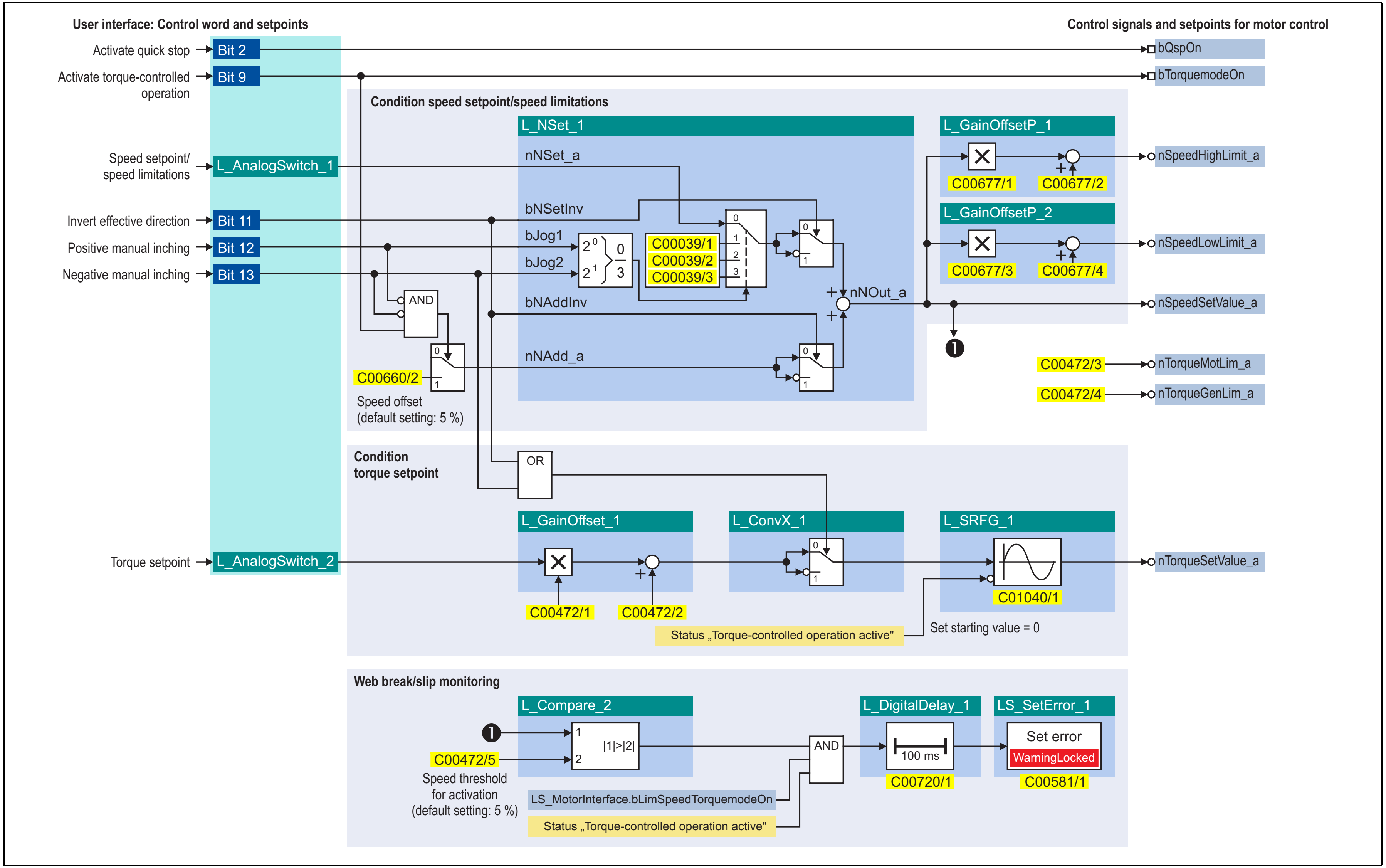
Parameter (Block)	Possible settings			Info
C00103/1...3 (L_NSet_1)	0.000	s	999.999	Deceleration times for fixed setpoints 1 ... 3 • Lenze setting: 1 s
C00677/1 (L_GainOffsetP_1)	-199.99	%	199.99	Separate gain factor for the positive speed limitation • Lenze setting: 100 %
C00677/2 (L_GainOffsetP_1)	-199.99	%	199.99	Separate offset value for the positive speed limitation • Lenze setting: 0 % reference speed (C00011)
C00677/3 (L_GainOffsetP_2)	-199.99	%	199.99	Separate gain factor for the negative speed limitation • Lenze setting: 100 %
C00677/4 (L_GainOffsetP_2)	-199.99	%	199.99	Separate offset value for the negative speed limitation • Lenze setting: 0 % reference speed (C00011)
Condition torque setpoint				
C00472/2 (LS_ParFree_a)	-199.99	%	199.99	Offset value for torque setpoint • Lenze setting: 0 % maximum torque (C00057)
Web break/slip monitoring				
C00472/5 (LS_ParFree_a)	-199.99	%	199.99	Speed threshold for activating the web break/slip monitoring • Lenze setting: 5 % reference speed (C00011) ▸ Web break/slip monitoring (📖 30)
C00581/1 (LS_SetError_1)				Response when web break/slip monitoring has tripped • Lenze setting: "4: WarningLocked"
	0	No Reaction		
	1	Fault		
	2	Trouble		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	Warning		
	6	Information		
C00720/1 (L_DigitalDelay_1)	0.000	s	3600.000	Tripping delay for web break/slip monitoring • Lenze setting: 0.1 s

4 Detailed functions of the technology application

This chapter describes the functions implemented in the "Actuator Torque" technology application with the possible settings relevant for the application.

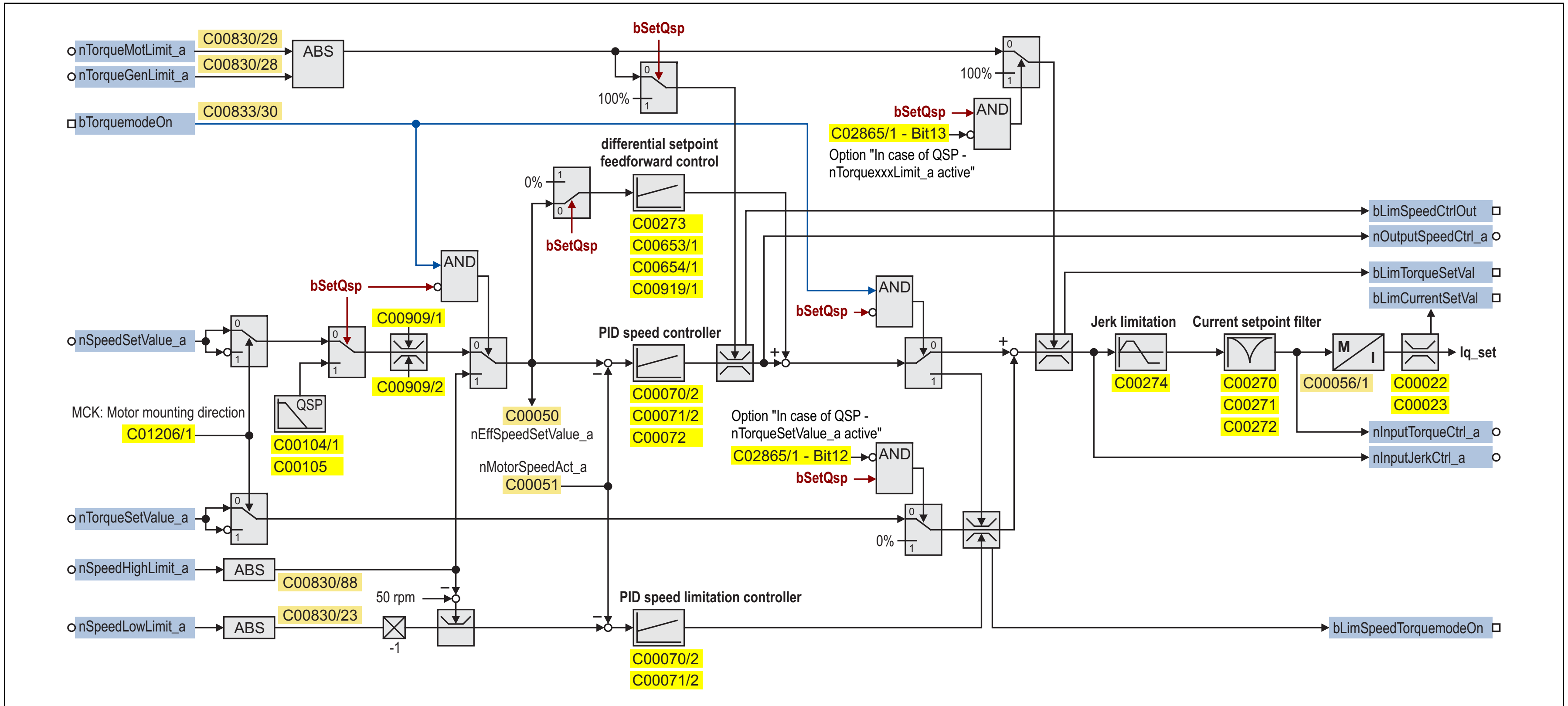
Detailed information on the function and parameterisation of the functions described in the following can be found in the reference manual/online help of the controller.

4.1 Signal flow of the technology application



4.2 Signal flow of the motor control

The following basic signal flow makes it easier to understand the setpoint generation in the torque-controlled operation. The change-over between speed and torque control is carried out via the *bTorquemodeOn* input of the **LS_MotorInterface** system block.



Inputs	Info
nTorqueMotLimit_a	Torque limitation in motor mode
nTorqueGenLimit_a	Torque limitation in generator mode
bTorquemodeOn	FALSE = speed control with torque limitation
	TRUE = torque control with speed limitation
nSpeedSetValue_a	Speed setpoint
nTorqueSetValue_a	Torque setpoint
nSpeedHighLimit_a	Upper speed limit for the speed limitation
nSpeedLowLimit_a	Lower speed limit for speed limitation

Outputs	Info
bLimSpeedCtrlOut	TRUE = speed controller output is limited internally
nOutputSpeedCtrl_a	Speed controller output
bLimTorqueSetVal	TRUE = torque setpoint is limited internally
bLimCurrentSetVal	TRUE = current setpoint is limited internally
nInputTorqueCtrl_a	Input value of the torque control (torque setpoint)
nInputJerkCtrl_a	Input value of the jerk limitation
bLimSpeedTorquemodeOn	TRUE = speed limitation for torque control is active

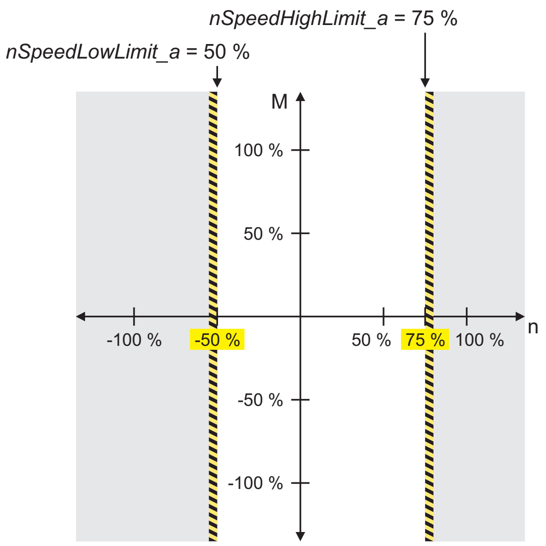
4.3 Basic drive functions (MCK)

4.3.1 Torque-controlled operation with the basic "speed follower" drive function

The torque-controlled operation is not implemented by an own operating mode in the Motion Control Kernel (MCK) of the 8400. Thus, the basic drive function "speed follower" is used in the technology application for the torque-controlled operation.

- With reference to the maximum torque (C00057), the torque setpoint is defined in a scaled way via the *nTorqueSetValue_a* input of the **LS_MotionControlKernel** system block.
  - *nTorqueSetValue_a* always has an additive effect on the torque setpoint in the motor control, even in case of speed control. Thus, this interface can also be used for defining an acceleration torque.
  - In the technology application, it is ensured that the value at *nTorqueSetValue_a* is = 0 if no torque-controlled operation is requested (*bTorquemodeOn* = FALSE).
- With reference to the reference speed (C00011), the speed limitations are defined in a scaled way via the two inputs *nSpeedHighLimit_a* and *nSpeedLowLimit_a* of the **LS_MotorInterface** system block.
  - *nSpeedHighLimit_a* and *nSpeedLowLimit_a* act as an amount, the sign is neglected.
  - The compliance with the upper and lower speed limitation is ensured via two independent speed controllers. Both speed controllers work with the parameters Vp (C00070) and Ti (C00071).
  - If speed limitation is active, the drive can generate up to 100% of its torque depending on the deviation of the speed setpoint from the actual speed value!

Example: Definition of the speed limitations



nSpeedHighLimit_a	Upper (positive) speed limitation
nSpeedLowLimit_a	Lower (negative) speed limitation

▸ Signal flow of the motor control (📖 23)

4.3.2 Holding brake control

This basic function is used for low-wear control of a holding brake.

Danger!

Please note that the holding brake is an important element of the safety concept of the entire machine.

Thus, proceed very carefully when commissioning this system part!

Detailed information on how to parameterise the holding brake control can be found in the reference manual/online help of the controller in the chapter "Basic drive functions (MCK)".

The documentation of the holding brake control contains safety instructions which must be observed!

Application-specific notes on the holding brake control:

- In the Lenze setting, the mode 0 (brake control off) is preset in C02580.
- The technology application is prepared for the control of a 24-V holding brake via the high current output.
- When the holding brake control is used, the *nSpeedSetValue_a* input of the **LS_MotionControlKernel** system block has to be connected in a plausible way. In the automaticallly controlled operation (C02580 = "12: Autom. controlled") the brake is released/closed based on the speed setpoint provided at this input and the set switching thresholds:
  - Release brake:
    nSpeedSetValue_a > switching threshold (C02581/1) + hysteresis for release (C02581/2)
  - Close brake:
    nSpeedSetValue_a < switching threshold (C02581/1) - hysteresis for closing (C02581/3)
- The current torque setpoint *nTorqueSetValue_a* does not influence the holding brake control.
  - The brake would also be released if *nTorqueSetValue_a* = 0 % if the *nSpeedSetValue_a* speed setpoint exceeds the switching threshold!
  - The brake is also released if the torque setpoint has the inverted sign of the speed setpoint.
- In the technology application, the holding brake control in the Lenze setting works with the same setpoints as the speed limitation. A separate adaptation of the limitations is possible by parameterising the blocks **L_GainOffsetP_1** and **L_GainOffsetP_2**.
- When the holding brake control (C02580 ≠ 0) is used, the drive can only follow an external torque if the brake is released.
  - The **LS_MotionControlKernel** system block signals the "brake released" status via its *bMBrakeReleased* output.
  - When C00825/12 = "1: True" in the FB **L_DigitalLogic5_1**, the "torque-controlled operation active"status (bit 14 in the Status word) additionally requires the brake to be released (*bMBrakeReleased* is AND'ed in addition).

4.4 Manual jog (inching mode)

The "manual jog (inching mode)" function is implemented via the **L_NSet_1** ramp generator.
▶ Signal flow of the technology application (📖 22)

The FB **L_NSet_1** supports up to 15 fixed setpoints (JOG setpoints) with individually adjustable acceleration and deceleration times.

- For a manual inching in positive direction, the fixed setpoint 1 (C00039/1) is used.
- For a manual inching in negative direction, the fixed setpoint 2 (C00039/2) is used.
- The respective function is activated via bit 12 and bit 13 of the application control word (FB **L_ConvBitsToWord_1**). In the default setting, bit 12 and bit 13 are linked to the digital inputs DI5 and DI6.
  An activation via the fieldbus interface (MCI) is possible after a corresponding adaptation of the user interface. See commissioning; Step 2 (optional): Establish control via the fieldbus interface (MCI). (📖 14)

Parameter (Block)	Possible settings			Info
C00039/1 (L_NSet_1)	-199.99	%	199.99	Fixed setpoint 1 • For manual inching in positive direction. • Lenze setting: 10 % reference speed (C00011)
C00039/2 (L_NSet_1)	-199.99	%	199.99	Fixed setpoint 2 • For manual inching in negative direction. • Lenze setting: -10 % reference speed (C00011)
C00039/3 (L_NSet_1)	-199.99	%	199.99	Fixed setpoint 3 • Fixed setpoint 3 is selected if bit 12 **AND** bit 13 of the application control word are set to "1". • Lenze setting: 0 % reference speed (C00011)
C00101/1...3 (L_NSet_1)	0.000	s	999.999	Acceleration times for fixed setpoints 1 ... 3 • Lenze setting: 1 s
C00103/1...3 (L_NSet_1)	0.000	s	999.999	Deceleration times for fixed setpoints 1 ... 3 • Lenze setting: 1 s

"Hard" inching

In case of speed control, the torque results from the load and the moment of inertia to be accelerated. It can assume maximum values. Thus, the inching mode at speed control can also be called "hard inching".

"Soft" inching

In parallel to the preselection of the respective fixed setpoint for the inching mode, the torque-controlled operation can be switched on via bit 9 of the application control word (FB **L_ConvBitsToWord_1**). In the default setting, bit 9 is linked to the digital input DI3.

In contrast to the inching mode at speed control, only the defined torque setpoint is available. Thus, the inching mode in the torque-controlled operation can also be called "soft" inching.

4.5 Functions for the torque-controlled operation

4.5.1 Offset and gain of the torque setpoint

The torque setpoint can be increased via the FB **L_GainOffset_1** and an offset can be applied.
▶ Signal flow of the technology application (📖 22)

- The gain factor (C00472/1) serves to adapt the percentage setpoint to the reference value maximum torque (C00057) of the controller.
  Example: The technology application is the basis for a torque-supporting drive. The current torque of a master drive is the setpoint for the supporting drive. Both drives have different maximum torques (C00057) for the scaled torque selection. The torque, however, is to split 1:1.
  - Maximum torque - master = 20 Nm
  - Maximum torque - slave = 30 Nm

  → This results in a gain factor of 20 Nm/30 Nm = 66.66 %
- The setting of an offset value (C00472/2) may be required to precontrol a constant torque. An offset, for instance, serves to compensate the constant part of a friction torque.

Parameter (Block)	Possible settings			Info
C00472/1 (LS_ParFree_a)	-199.99	%	199.99	Gain factor for torque setpoint • Lenze setting: 100 %
C00472/2 (LS_ParFree_a)	-199.99	%	199.99	Offset value for torque setpoint • Lenze setting: 0 % maximum torque (C00057)

4.5.2 Ramp generator for the torque setpoint

After the FB **L_GainOffset_1**, the torque setpoint is applied to the ramp generator **L_SRFG_1**.
▶ Signal flow of the technology application (📖 22)

- A ramp generation in the torque setpoint path may be required to smooth setpoint step-changes.
- In case of typical torque followers – distribution of the load torque to several axes - no ramp generation is required.
- The ramp generator is loaded with the setpoint "zero" if
  - the Motion Control Kernel (MCK) is in "StandBy" mode
    ("StandBy" = internal operating mode at quick stop, pulse inhibit and DC-injection braking)
    OR
  - if no torque-controlled operation has been requested.

Parameter (Block)	Possible settings			Info
C01040/1 (L_SRFG_1)	0.001	s	999.999	Linear ramp time for the torque setpoint • Lenze setting: 0.010 s

4.5.3 Speed limitation/forward motion for the speed setpoint

The speed of torque-controlled motors is determined by external factors, e.g. via the material in case of a withdrawal. For this purpose, the set speed limitations have to be considerably higher than the current speed of the motor. In order to ensure this, an offset or forward motion value is added to the speed setpoint.

The technology application provides two options to adapt the speed limitations for the torque-controlled operation:

A. Common offset for both limit values via the FB **L_FixSet_a_1**.

B. Separate offset and gain factor via the two FBs **L_GainOffsetP_1** and **L_GainOffsetP_2**.

If the speed control is active instead, both adaptations do not affect the speed setpoint.

Parameter (Block)	**Possible settings**			**Info**
C00660/2 (L_FixSet_a_1)	-199.99	%	199.99	Common speed offset value for both speed limitations • Only effective for torque followers. • Lenze setting: 5 % reference speed (C00011)
C00677/1 (L_GainOffsetP_1)	-199.99	%	199.99	Separate gain factor for the positive speed limitation • Lenze setting: 100 %
C00677/2 (L_GainOffsetP_1)	-199.99	%	199.99	Separate offset value for the positive speed limitation • Lenze setting: 0 % reference speed (C00011)
C00677/3 (L_GainOffsetP_2)	-199.99	%	199.99	Separate gain factor for the negative speed limitation • Lenze setting: 100 %
C00677/4 (L_GainOffsetP_2)	-199.99	%	199.99	Separate offset value for the negative speed limitation • Lenze setting: 0 % reference speed (C00011)

Details on the signal flow

In the torque-controlled operation, the selection for both speed limitations is led via the **L_NSet_1** setpoint generator. this serves to apply the functions for signal conditioning to the speed limitations.

In the speed-controlled operation, not the speed limitations but the speed setpoint is led via the **L_NSet_1** setpoint generator. All functions for signal conditioning can now be applied to the speed-controlled operation.

▶ Signal flow of the technology application (📖 22)

4.5.4 Inversion of the effective direction for the torque setpoint

A simple inversion of the effective direction of the torque setpoint via bit 11 of the application control word (FB **L_ConvBitsToWord_1**) is implemented in the technology application. In the default setting, bit 11 is linked to C00470/1.

Parameter (Block)	Possible settings		Info
C00470/1 (LS_ParFree_b)			1 ≡ Invert effective direction • Inversion of the torque setpoint, the speed limitations and the parameterised fixed setpoints. • Lenze setting: 0
	0	Effective direction CW	
	1	Effective direction CCW	

The inversion can be applied if the effective direction of the torque is to be inverted as requested by the process.

STOP **Stop!**

An inversion of the torque setpoint causes a reversal of the speed limitations if no further measures are taken! (The speed limitations act with regard to the sign of the motor speed.) Depending on the application case, the set values for the two speed limitations have to be inverted as well.

As a second condition, the request of manual inching in negative direction also causes an inversion of the torque setpoint. This is required to enable "soft" inching in negative direction. ▶ Manual jog (inching mode) (📖 26)

Details on the signal flow

For signal inversion of the torque setpoint, the FB **L_ConvX_1** is used.

The signal inversion of the speed limitations, the fixed setpoints and the speed offset value is carried out in the **L_NSet_1** setpoint generator via the functions *bNSetInv* and *bNAddInv*.

▶ Signal flow of the technology application (📖 22)

4.6 Monitoring functions

4.6.1 Web break/slip monitoring

In the torque-controlled operation (*bTorqueModeOn* = TRUE), the normal state is that the motor follows the torque setpoint and no speed limitation is active. If the load is omitted (e.g. by web break or slip), the axis is accelerated by the torque until the speed limitation takes effect.

In the default setting, the monitoring function implemented in the technology application triggers the user error 1 with the "WarningLocked" response after a tripping delay of 100 ms if:

1. the torque-controlled mode is active **AND**
2. the motor control signals that a speed limitation is active in the torque-controlled operation (*bLimSpeedTorquemodeOn* = TRUE) **AND**
3. the amount of the speed setpoint is higher than the speed threshold set in C00472/5.

The web break/slip monitoring can be adapted via the following parameters:

Parameter (Block)	**Possible settings**			**Info**
C00472/5 (LS_ParFree_a)	-199.99	%	199.99	Speed threshold for activating the web break/slip monitoring • Lenze setting: 5 % reference speed (C00011)
C00581/1 (LS_SetError_1)				Response when web break/slip monitoring has tripped • Lenze setting: "4: WarningLocked"
	0	No Reaction		
	1	Fault		
	2	Trouble		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	Warning		
	6	Information		
C00720/1 (L_DigitalDelay_1)	0.000	s	3600.000	Tripping delay for web break/slip monitoring • Lenze setting: 0.1 s

▸ Signal flow of the technology application (📖 22)

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

Lenze Drives GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service

Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Inverter

8400

"Electrical Shaft Master/Slave"
technology application
for 8400 TopLine C

Software manual EN

13467297

Lenze

Contents

1 About this documentation

This documentation described the software-based solution of a task. The transferability of the described solution to the respective application case needs to be checked by the user. If required, the user has to adapt the solution accordingly. Thus, physical aspects as e.g. drive dimensioning is not part of this documentation.

Danger!

The controller is a source of danger which may lead to death or severe injury of persons.

To protect yourself and others against these dangers, observe the safety instructions before switching on the controller.

Please read the safety instructions provided in the **8400 mounting instructions** and in the **8400 hardware manual**. Both documents are supplied with the controller.

Target group

This documentation addresses to all persons

- who want to use the "Electrical Shaft Master" or "Electrical Shaft Slave" technology application for the 8400 TopLine inverter, and
- who are familiar with handling the device and the »Engineer« software.

Validity

The information in this documentation are valid for the following technology applications:

Technology application	from version
Electrical Shaft Master	01.00
Electrical Shaft Slave	01.00

Screenshots/application examples

All screenshots provided in this documentation are application examples. Depending on the software version of the controller and the version of the installed »Engineer« software, the screenshots in this documentation may differ from the representation in the »Engineer«.

Tip!

Information and tools for Lenze products are provided in the download area at

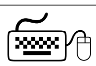
http://www.lenze.com → Download

1.1 Document history

Version			Description
1.0	07/2014	TD05	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is generally used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the prefix "0x" is used. Example: 0x60F4
Binary number	0b	For binary numbers, the prefix "0b" is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that only applies to a certain controller software version or higher is identified accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software ...
Window	*italics*	The *Message* window ... / The *Options* dialog box...
Variable name		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Press **<F1>** to open the online help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...
Hyperlink	underlined	Optically highlighted reference to another topic. In this documentation activated by mouse-click.
Icons		
Page reference	(🕮 5)	Optically highlighted reference to another page. In this documentation activated by mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Engineering Tools	Software solutions for simple engineering at all stages
	»EASY Navigator« – Ensures easy operator guidance • All practical Lenze engineering tools at a glance • Tools can be selected quickly • Clearly arranged, simplifying the engineering process from the start
	»EASY Starter« – Simple tool for service technicians • Especially developed for the commissioning and maintenance of Lenze devices • Graphical user interface with few buttons • Simple online diagnostics, parameterisation and commissioning • No risk of accidentally changing the application • Ready applications can be loaded to the device
	»Engineer« – Multi-device engineering • For all products from our L-force portfolio • Practice-oriented user interface • Easy handling due to graphical user interfaces • Suitable for all project stages (configuration, commissioning, production) • Parameter setting and configuration
Code	Parameter used for controller parameterisation or monitoring. The term is usually called "index".
Subcode	If a code contains several parameters, these are stored in "subcodes". This Manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). The term is usually called "subindex".
Lenze setting	This setting is the default factory setting of the device.
FB Editor	Abbreviation for function block editor. Graphic interconnection tool which is available in the »Engineer« for function block interconnections on the **FB Editor**.
Function block	General designation of a function block for free interconnection in the FB Editor. A function block (short: "FB") can be compared with an integrated circuit that contains a specific control logic and delivers one or several values when being executed. Example: "L_Arithmetic_1" (FB for arithmetic operations) Many function blocks are available several times (e.g. L_And_1, L_And_2, and L_And_3).
System block	In the function block editor of the »Engineer«, system blocks provide interfaces to basic functions, "free codes", and to the hardware of the inverter (e.g. to the digital inputs). Each system block is available only once.
Port block	Block for implementing the process data transfer via a fieldbus
LP	Abbreviation for **L**enze **P**ort block Example: "LP_CanIn1" (CAN1 port block)
LS	Abbreviation for **L**enze **S**ystem block Example: "LS_DigitalInput" (system block for digital input signals)
MCI	Abbreviation for **M**otionbus **C**ommunication **I**nterface (fieldbus interface) The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system.
Technology application	A technology application is a drive solution based on the experience and know-how of Lenze in which function blocks interconnected to a signal flow form the basis for implementing typical drive tasks.
USB diagnostic adapter	The USB diagnostic adapter is used for the operation, parameterisation, and diagnostics of the controller. Data are exchanged between the PC (USB connection) and the controller (diagnostic interface on the front) via the diagnostic adapter. Order designation: E94AZCUS

1.4 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling

2 Features of the technology application

The "Electrical Shaft" technology application serves to easily establish a so-called "electrical shaft" between several 8400 TopLine inverters, so that two or more devices can be used in angular synchronism.

The axis bus (terminal block X10) is used as transmission medium. A modification to the system bus (CAN) can be effected; however, this is not covered by this description.

2.1 Functional overview

General

- Electrical shaft in line topology (all drives receive the same master value)
- Transmission of the master value via axis bus
- 1 master drive and up to 61 slave drives

Master functions ("Electrical Shaft Master" technology application)

- Setpoint selection optionally via
  - plugged-in communication module/fieldbus interface (MCI)
  - "CAN on board" system bus (X1)
  - analog input (X3)
  - multi encoder input (X8)
- ramp function generator for main setpoint from the fieldbus or analog input
  The ramp function generator with adjustable ramps prevents jerks during the changeover or in the case of abrupt setpoint changes. By this, a master control can easily select a setpoint without any ramps.
- Stop function with an individual deceleration ramp for the electrical shaft (bring whole interconnection synchronously to a standstill)
- Quick stop function (for stop in the event of a breakdown)
- Adjustable gearbox factor
- Speed trimming
  For the connection of correction values, e.g. from a higher-level control loop. This permits an acceleration or deceleration of the drive.
- Position offset (angular adjustment)
  For the selection of a fixed angular offset for the setpoint. The angular offset is controlled via a profile generator and can therefore be continuously adjusted during master-slave operation.
- Use of the state machine of the Motion Control Kernel (MCK) and the following basic drive functions:
  - Homing (homing/reference setting)
  - Manual jog (optionally with the following slave drives)
- Optionally setpoint- or actual value-based master value for slave drive(s)
- Optionally master function for slave drive(s) with "Position Follower" application
- Central reset (acknowledgement) of errors

Slave functions ("Electrical Shaft Slave" technology application)

- Setpoint selection optionally by the master drive or multi-encoder
- Quick stop function (for stop in the event of a breakdown)
- Adjustable stretch factor (mapping of the procedural speed/angle ratio to the master value)
  This serves to vary the speed of the slave drive in an adjustable ratio to the master drive (e.g. for the purpose of material stretching)
- Adjustable gearbox factor (mapping of the mechanics of the respective slave)
- Speed trimming
- Position offset (angular adjustment)
- Mark correction via touch probe sensor
- Use of the state machine of the Motion Control Kernel (MCK) and the following basic drive functions:
  - Homing (homing/reference setting)
  - Manual jog
- Angle correction for transmission errors of the axis bus
- Error message from the slave to the master via I/O axis bus for central error handling

2.2 Application ranges

- Conveyors
- Stretching lines
- Wire drawing machines
- Transport devices
- Printing machines
- Packaging machine
- ...

2.3 System requirements

The technology application was created with the L-force »Engineer« V2.20 and can only be used with the versions V2.20 or higher.

Software

Product	Order designation	from version
L-force »Engineer« HighLevel	ESPEV-EHNNN	2.20

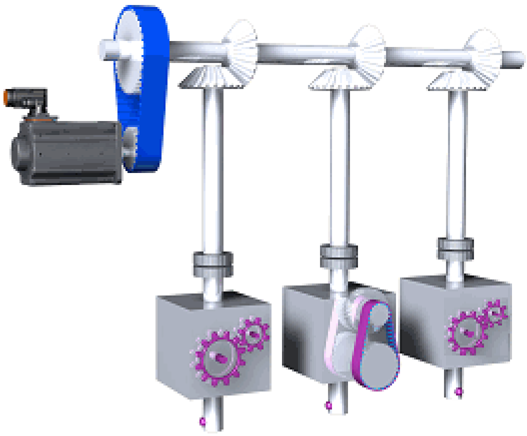
Hardware

Product	Order designation	from hardware version	from software version
Inverter Drives 8400 TopLine C	E84AVTCxxxxx	VD	14.00

2.4 Basics of the electrical shaft

2.4.1 Synchronisation of the drives via a master angle

Coupling the drives via a master angle results in a fixed position assignment for all drives of the electrical shaft like for a mechanical shaft:



The master drive provides the master angle and transfers it to the other (slave) drives following this master angle.

Advantages of this type of synchronisation

- Communication between the drives is kept very simple. A time-consuming evaluation of the status signals of each individual drive and the control signals to be generated for each individual drive is dispensed with.
- The flexible signal structures provide for an easy execution of trimming functions. This makes it possible to easily synchronise and optimise the motion sequences in machines.
- By variation of the master angle speed, the number of cycles or production speed of the machine is/are altered. In this process, the drives' position assignment towards each other is maintained.

2.4.2 Transmission of the master angle via axis bus

In the case of the 8400 TopLine, the integrated axis bus can be used for the transmission of the master angle.

The axis bus serves to couple several 8400 TopLine inverters in an axis interconnection. It is the main task of the axis bus to carry out simple cross-data exchange from axis to axis.

A distinction between the two transmission channels "data transfer axis bus" and "IO axis bus" is made, which can be used as follows:

- The data transfer axis bus is based on CAN physics and enables a high-performance data transfer of master values and control signals to other 8400 TopLine controllers. For this purpose, a synchronisation of the internal time base of the controllers via the IO axis bus is always required.
- The IO axis bus is a 1-wire bus with open-collector circuitry (5 V isolated). The IO axis bus can either be used to transfer controller errors in the interconnection ("release cord" principle) or as a pure open-collector IO function. In the first case, the internal time base of the controllers can be synchronised via the IO axis bus.

Max. 62 controllers can be connected to the axis bus.

Stop!

The data transfer axis bus of the 8400 TopLine controller is designed especially for performance and simplicity. HMIs and other peripherals as well as the »Engineer« are not supported at the axis bus.

The IO axis bus of the 8400 TopLine controller is not compatible to the state bus of the 9300/9400 device series due to different voltage levels!

Some detailed information about the axis bus can be found in the reference manual/in the online help of the inverter in the "Axis bus" chapter of the same name.

2.4.3 Master value or actual value transmission?

In contrast to the transmission of the actual value, the transmission of the master value results in a considerably smoother machine operation; however, mark corrections or higher-level controls as well as interference of the leading drive have no impact on the interconnection.

The transmission of the actual value causes a slightly more uneven machine operation. Higher-level controls as well as interference of the leading drives have an impact on the interconnection.

The selection of the value that is to serve as speed setpoint for the slave drives is defined in the master drive:

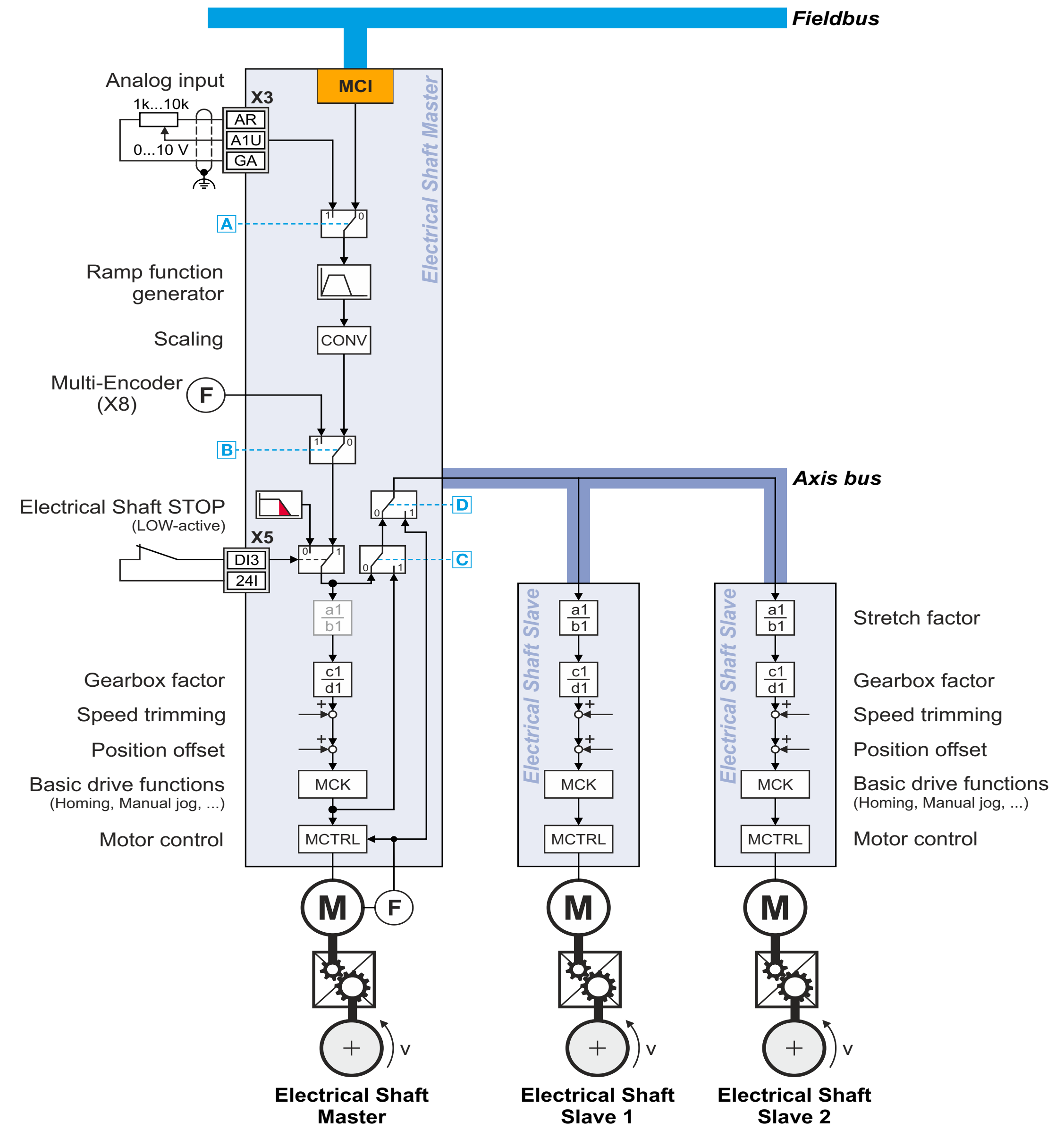
- Mere master value: speed setpoint irrespective of the master
- Master value of the master: speed setpoint of the master drive's motion control
  (speed setpoint of the Motion Control Kernel; including manual jog, homing, ...)
- Actual master value: exact master position including following errors

2.5 Basic signal flow

In the technology application, function blocks and system blocks are interconnected so that an electrical shaft can be implemented for the application ranges mentioned before.

In the following, the basic signal flow between a master drive and two slave drives with the basic functions is shown.

Basic signal flow of the "Electrical Shaft" technology application



A	C00470/1: selection of the sources for the speed setpoint (MCI interface or analog input 1)
B	C00470/2: selection of the multi-encoder at X8 as source for the speed/angle setpoint
C	C00470/5: selection of the master value for the slave drives
D	C00470/6: selection actual master value for the slave drives

2.6 Parameter setting in the FB Editor view

You can make the settings of the application-specific parameters directly in the FB Editor. This has the advantage that the signal flow can be traced. The interaction of the modules becomes clear. Moreover, you can reconfigure the I/O interconnection using the FB Editor and carry out an online monitoring of the application running in the device (e.g. for diagnostic purposes).

- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.
- Colour codes and comments support you in handling the FB Editor.
  - The areas highlighted in turquoise represent the "user interface". If required, the pre-assignment of the I/O terminals can be adapted here and a control via the fieldbus interface (MCI) can be established.
  - In the areas highlighted in yellow, application-specific settings are required.

Detailed information on how to work with the FB Editor can be found in the reference manual/online help of the controller in the chapter "Working with the FB Editor".

2.7 Pre-assignment of the user interface for the master drive

2.7.1 I/O terminals

<table>
<tr><th>Terminal</th><th colspan="3">Function</th></tr>
<tr><td colspan="4">Digital input terminals</td></tr>
<tr><td>X5/RFR</td><td colspan="3">Controller enable</td></tr>
<tr><td rowspan="5">X5/DI1</td><td colspan="3">Start/stop homing</td></tr>
<tr><td colspan="2">DI1</td><td>Function</td></tr>
<tr><td colspan="2" rowspan="2">HIGH</td><td>For selection of homing modes "4" ... "15" in C01221:
Start reference search</td></tr>
<tr><td>For selection of homing modes "100: SetRef" in C01221:
Set home position manually</td></tr>
<tr><td colspan="2">HIGH↘LOW</td><td>Stop homing</td></tr>
<tr><td rowspan="4">X5/DI2</td><td colspan="3">Activate quick stop (e.g. in the case of a fault</td></tr>
<tr><td colspan="2">DI2</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Deactivate quick stop</td></tr>
<tr><td colspan="2">HIGH</td><td>Activate quick stop
• The motor control is decoupled from the setpoint selection (speed and torque) and within the deceleration time parameterised in C00105, the motor is brought to standstill (n_{act} = 0).
• A pulse inhibit is set if the auto-DCB function has been activated via C00019.</td></tr>
<tr><td rowspan="4">X5/DI3</td><td colspan="3">Stop electrical shaft
• The connection is configured in a fail-safe fashion (LOW = Electrical Shaft STOP).</td></tr>
<tr><td colspan="2">DI3</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Stop electrical shaft:
synchronously bring whole drive system to a standstill</td></tr>
<tr><td colspan="2">HIGH</td><td>Electrical shaft running</td></tr>
<tr><td>X5/DI4</td><td colspan="3">Connection of pre-switch sensor for reference search
• The edge sensitivity of this input and the response to the pre-switch signal depend on the homing mode selected.</td></tr>
<tr><td rowspan="6">X5/DI5
X5/DI6</td><td colspan="3">Manual jog</td></tr>
<tr><td>DI5</td><td>DI6</td><td>Function</td></tr>
<tr><td>LOW</td><td>LOW</td><td>-</td></tr>
<tr><td>HIGH</td><td>LOW</td><td>Manual jog in positive direction</td></tr>
<tr><td>LOW</td><td>HIGH</td><td>Manual jog in negative direction</td></tr>
<tr><td>HIGH</td><td>HIGH</td><td>- / Manual jog in the direction selected first</td></tr>
<tr><td rowspan="5">X5/DI7</td><td colspan="3">Reset error messages</td></tr>
<tr><td colspan="2">DI7</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>No reset</td></tr>
<tr><td colspan="2">1. LOW↗HIGH</td><td>With the first positive edge, the error messages of all slave drives are reset.</td></tr>
<tr><td colspan="2">2. LOW↗HIGH</td><td>With the second positive edge, the error message in the master drive is reset.
(The bus runtimes between the master and slave make a double LOW-HIGH edge necessary for the reset!)</td></tr>
</table>

Terminal	Function
Analog input terminals	
X3/A1U	Speed setpoint • If this input is to be used as setpoint source, C00470/1 is to be set to the value "1: True". • Scaling: 10 V ≡ 100 % reference speed (C00011)
X3/A2U	- (not assigned, can be used freely)
Digital output terminals	
X4/DO1	Status "Drive is ready"
X4/DO2	- (not assigned, can be used freely)
X4/DO3	Status "Home position is known"
X107/BD1, BD2	Control of a holding brake by the basic function "holding brake control"
X101/COM, NO	Status "Error is pending"
Analog output terminals	
X3/O1U	Actual speed value • Scaling: 10 V ≡ 100 % reference speed (C00011)
X3/O2U	Actual torque • Scaling: 10 V ≡ 100 % maximum torque (C00057)

2.7.2 Process data input words

Fieldbus interface (MCI); port block **LP_MciIn**

Input words	Assignment
Word 1	Control word (for bit assignment see the following table)
Word 2	Speed setpoint • Scaling: 16384 ≡ 100 % reference speed (C00011)
Words 3 ... 16	- (not preconfigured)

Control word	Function
Bit 0	- (not preconfigured)
Bit 1	- (not preconfigured)
Bit 2	- (not preconfigured)
Bit 3	- (not preconfigured)
Bit 4	- (not preconfigured)
Bit 5	- (not preconfigured)
Bit 6	- (not preconfigured)
Bit 7	- (not preconfigured)
Bit 8	- (not preconfigured)
Bit 9	- (not preconfigured)
Bit 10	- (not preconfigured)
Bit 11	- (not preconfigured)
Bit 12	- (not preconfigured)
Bit 13	- (not preconfigured)
Bit 14	- (not preconfigured)
Bit 15	- (not preconfigured)

2.7.3 Process data output words

Fieldbus interface (MCI); port block **LP_MciOut**

Output words	Assignment
Word 1	Status word (for bit assignment see the following table)
Word 2	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	Actual torque • Scaling: 16384 ≡ 100 % maximum torque (C00057)
Word 4 ... 16	- (not preconfigured)

<table>
<tr><th>Status word</th><th colspan="6">Status</th></tr>
<tr><td>Bit 0</td><td colspan="6">1 ≡ Group error active (configurable in C00148)</td></tr>
<tr><td>Bit 1</td><td colspan="6">1 ≡ Inverter control inhibited (pulse inhibit is active)</td></tr>
<tr><td>Bit 2</td><td colspan="6">1 ≡ Drive controller is ready for operation</td></tr>
<tr><td>Bit 3</td><td colspan="6">1 ≡ Quick stop is active</td></tr>
<tr><td>Bit 4</td><td colspan="6">1 ≡ Setpoint torque is in the limitation</td></tr>
<tr><td>Bit 5</td><td colspan="6">- (not preconfigured)</td></tr>
<tr><td rowspan="2">Bit 6</td><td colspan="6">During open-loop operation:
1 ≡ Speed setpoint < Comparison value (C00024)</td></tr>
<tr><td colspan="6">During closed-loop operation:
1 ≡ Actual speed value < Comparison value (C00024)</td></tr>
<tr><td>Bit 7</td><td colspan="6">1 ≡ Controller inhibited (controller inhibit is active)</td></tr>
<tr><td rowspan="14">Bit 8 ... 11</td><td colspan="6">Bit coded display of the active device status</td></tr>
<tr><td>Bit 11</td><td>Bit 10</td><td>Bit 9</td><td>Bit 8</td><td>Device status</td><td>Meaning</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>FirmwareUpdate</td><td>Firmware update function is active</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Init</td><td>Initialisation active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Ident</td><td>Identification active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>ReadyToSwitchOn</td><td>Device is ready to start</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>SwitchedOn</td><td>Device is switched on</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>OperationEnabled</td><td>Operation</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>Trouble</td><td>Trouble active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>Fault</td><td>Fault active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>TroubleQSP</td><td>TroubleQSP is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>SafeTorqueOff</td><td>Safe torque off is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>SystemFault</td><td>System fault active</td></tr>
<tr><td>Bit 12</td><td colspan="6">1 ≡ A warning is indicated</td></tr>
<tr><td>Bit 13</td><td colspan="6">1 ≡ A fault is active. The inverter is in the "Trouble" device state.
• The motor has no torque (is coasting) due to the inhibit of the inverter.
• The "Trouble" device status is automatically abandoned if the error cause has been removed.</td></tr>
<tr><td>Bit 14</td><td colspan="6">- (not preconfigured)</td></tr>
<tr><td>Bit 15</td><td colspan="6">1 ≡ Home position is known</td></tr>
</table>

2.8 Pre-assignment of the user interface for the slave drive

2.8.1 I/O terminals

Terminal	Function	
Digital input terminals		
X5/RFR	Controller enable	
X5/DI1	Speed trimming positive	
	DI1	Function
	HIGH	Speed trimming by the value set in C00476/1 (Lenze setting: +3 %)
X5/DI2	Speed trimming negative	
	DI2	Function
	HIGH	Speed trimming by the value set in C00476/2 (Lenze setting: -3 %)
X5/DI3	- (Reserved)	
X5/DI4	Option 1: Connection of a pre-switch sensor for reference search (Only relevant for the selection of homing modes "4" ... "7" in C01221)	
	DI4	Function
	LOW↗HIGH	Activate search profile data set for further reference search
	HIGH↘LOW	Enable home position detection
	Option 2: Connection of a touch probe sensor for mark synchronisation/TP correction • If the TP correction is to be used, C00470/3 is to be set to the value "1: True".	
X5/DI5	Connection of a master pulse for mark synchronisation/TP correction • If the TP correction is to be used, C00470/3 is to be set to the value "1: True".	
X5/DI6	Start/stop homing	
	DI6	Function
	HIGH	For selection of homing modes "4" ... "15" in C01221: Start reference search
		For selection of homing modes "100: SetRef" in C01221: Set home position manually
	HIGH↘LOW	Stop homing
X5/DI7	Reset error message	
	DI7	Function
	LOW	No reset
	LOW↗HIGH	Reset local error message in the slave drive
Analog input terminals		
X3/A1U	- (not assigned, can be used freely)	
X3/A2U	- (not assigned, can be used freely)	

Terminal	Function
Digital output terminals	
X4/DO1	Status "Drive is ready"
X4/DO2	- (not assigned, can be used freely)
X4/DO3	Status "Home position is known"
X107/BD1, BD2	Control of a holding brake by the basic function "holding brake control"
X101/COM, NO	Status "Error is pending"
Analog output terminals	
X3/O1U	Actual speed value • Scaling: 10 V ≡ 100 % reference speed (C00011)
X3/O2U	Current following error • Scaling: 10 V ≡ 1 revolution

2.8.2 Process data input words

Fieldbus interface (MCI); port block **LP_MciIn**

Input words	Assignment
Word 1	Control word (for bit assignment see the following table)
Words 2 ... 16	- (not preconfigured)

<table>
<tr><th>Control word</th><th colspan="3">Function</th></tr>
<tr><td>Bit 0</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 1</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 2</td><td colspan="3">1 ≡ Activate quick stop (QSP)</td></tr>
<tr><td>Bit 3</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 4</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 5</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 6</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 7</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 8</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 9</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 10</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 11</td><td colspan="3">1 ≡ deactivate X offset</td></tr>
<tr><td rowspan="6">Bit 12 ... 13</td><td colspan="3">Manual jog</td></tr>
<tr><td>Bit 12</td><td>Bit 13</td><td>Function</td></tr>
<tr><td>0</td><td>0</td><td>-</td></tr>
<tr><td>1</td><td>0</td><td>Manual jog in positive direction</td></tr>
<tr><td>0</td><td>1</td><td>Manual jog in negative direction</td></tr>
<tr><td>1</td><td>1</td><td>- / Manual jog in the direction selected first</td></tr>
<tr><td>Bit 14</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 15</td><td colspan="3">- (not preconfigured)</td></tr>
</table>

2.8.3 Process data output words

Fieldbus interface (MCI); port block **LP_MciOut**

Output words	Assignment
Word 1	Status word (for bit assignment see the following table)
Word 2	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	Actual torque • Scaling: 16384 ≡ 100 % maximum torque (C00057)
Word 4 ... 16	- (not preconfigured)

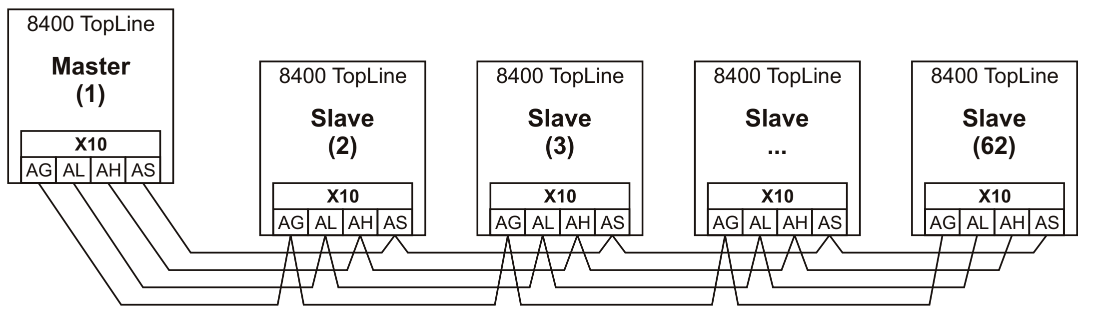
<table>
<tr><th>Status word</th><th colspan="6">Status</th></tr>
<tr><td>Bit 0</td><td colspan="6">1 ≡ Group error active (configurable in C00148)</td></tr>
<tr><td>Bit 1</td><td colspan="6">1 ≡ Inverter control inhibited (pulse inhibit is active)</td></tr>
<tr><td>Bit 2</td><td colspan="6">1 ≡ Drive controller is ready for operation</td></tr>
<tr><td>Bit 3</td><td colspan="6">1 ≡ Quick stop is active</td></tr>
<tr><td>Bit 4</td><td colspan="6">1 ≡ Setpoint torque is in the limitation</td></tr>
<tr><td>Bit 5</td><td colspan="6">- (not preconfigured)</td></tr>
<tr><td rowspan="2">Bit 6</td><td colspan="6">During open-loop operation:
1 ≡ Speed setpoint < Comparison value (C00024)</td></tr>
<tr><td colspan="6">During closed-loop operation:
1 ≡ Actual speed value < Comparison value (C00024)</td></tr>
<tr><td>Bit 7</td><td colspan="6">1 ≡ Controller inhibited (controller inhibit is active)</td></tr>
<tr><td rowspan="14">Bit 8 ... 11</td><td colspan="6">Bit coded display of the active device status</td></tr>
<tr><td>Bit 11</td><td>Bit 10</td><td>Bit 9</td><td>Bit 8</td><td>Device status</td><td>Meaning</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>FirmwareUpdate</td><td>Firmware update function is active</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Init</td><td>Initialisation active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Ident</td><td>Identification active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>ReadyToSwitchOn</td><td>Device is ready to start</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>SwitchedOn</td><td>Device is switched on</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>OperationEnabled</td><td>Operation</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>Trouble</td><td>Trouble active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>Fault</td><td>Fault active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>TroubleQSP</td><td>TroubleQSP is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>SafeTorqueOff</td><td>Safe torque off is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>SystemFault</td><td>System fault active</td></tr>
<tr><td>Bit 12</td><td colspan="6">1 ≡ A warning is indicated</td></tr>
<tr><td>Bit 13</td><td colspan="6">1 ≡ A fault is active. The inverter is in the "Trouble" device state.
• The motor has no torque (is coasting) due to the inhibit of the inverter.
• The "Trouble" device status is automatically abandoned if the error cause has been removed.</td></tr>
<tr><td>Bit 14</td><td colspan="6">1 ≡ "Electrical Shaft" operation switched on</td></tr>
<tr><td>Bit 15</td><td colspan="6">1 ≡ Home position is known</td></tr>
</table>

3 Short setup of the technology application

3.1 Hardware structure required

The "Electrical Shaft" technology application requires at least two 8400 TopLine inverters that are connected to each other via axis bus. A maximum of 62 nodes on the axis bus can be actuated:

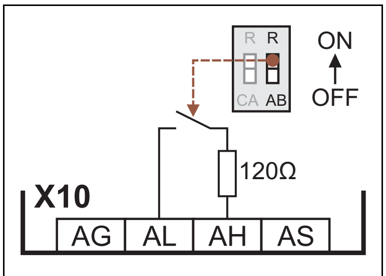
Axis bus topology



X10/AL, AH	Data transfer axis bus
X10/AS	IO axis bus
X10/AG	GND, reference potential of axis bus

The axis bus must be terminated between axis bus low (AL) and axis bus high (AH) at the first and last physical node each by a resistor (120 Ω).

The 8400 controller is provided with an integrated bus terminating resistor, which can be activated via the DIP switch labelled with "AB":



- OFF = bus terminating resistor is inactive
- ON = bus terminating resistor is active

3.2 Preconditions

For the execution of the short setup described in the following, the setting of the most important parameters (motor, feedback system, etc.) for each node on the axis bus is assumed.

The "commissioning wizard 8400" serves to carry out a guided commissioning of the controller based on the Lenze setting of the parameters.

How to proceed:

1. Before switch-on: Make sure that the controller is inhibited (input RFR open).
2. Switch on voltage supply of the controller.
   For parameter setting and diagnostics of the controller without motor operation, an external 24-V supply through a safely separated power supply unit (SELV/PELV) is sufficient.
3. Establish a communication link between controller and Engineering PC, e.g. via USB diagnostic adapter (E94AZCUS):
   - connect the USB diagnostic adapter to the X6 diagnostic interface.
   - establish a connection between the USB diagnostic adapter and the PC via a free USB port.
4. Start »Engineer« on the Engineering PC, e.g. via the Windows® start menu:
   Start → All programs → Lenze → Engineering → L-force Engineer...
   After the program start, no project has been loaded first and the *start-up wizard* is displayed.

> You can find detailed information on the options of the start-up wizard and on the general use of the »Engineer« in the online help for the program which you can call with **[F1]**.

5. Create a new project or open a project already available.
6. Go to *Project View* and select the 8400 controller.
7. Click the icon to go online.
   After a connection to the controller has been established, the following status is displayed in the *Status line*:

8. Click the icon to start the *commissioning wizard 8400*.
   - Now the commissioning wizard guides you step by step through the setting of the important parameters for a quick commissioning.
   - The **Next** button can only be activated again after all parameter settings in the device have been reset via the **Load Lenze setting** button.

3.3 Short setup of the master drive

The following steps only have to be carried out for the inverter which is to be the master in the network.

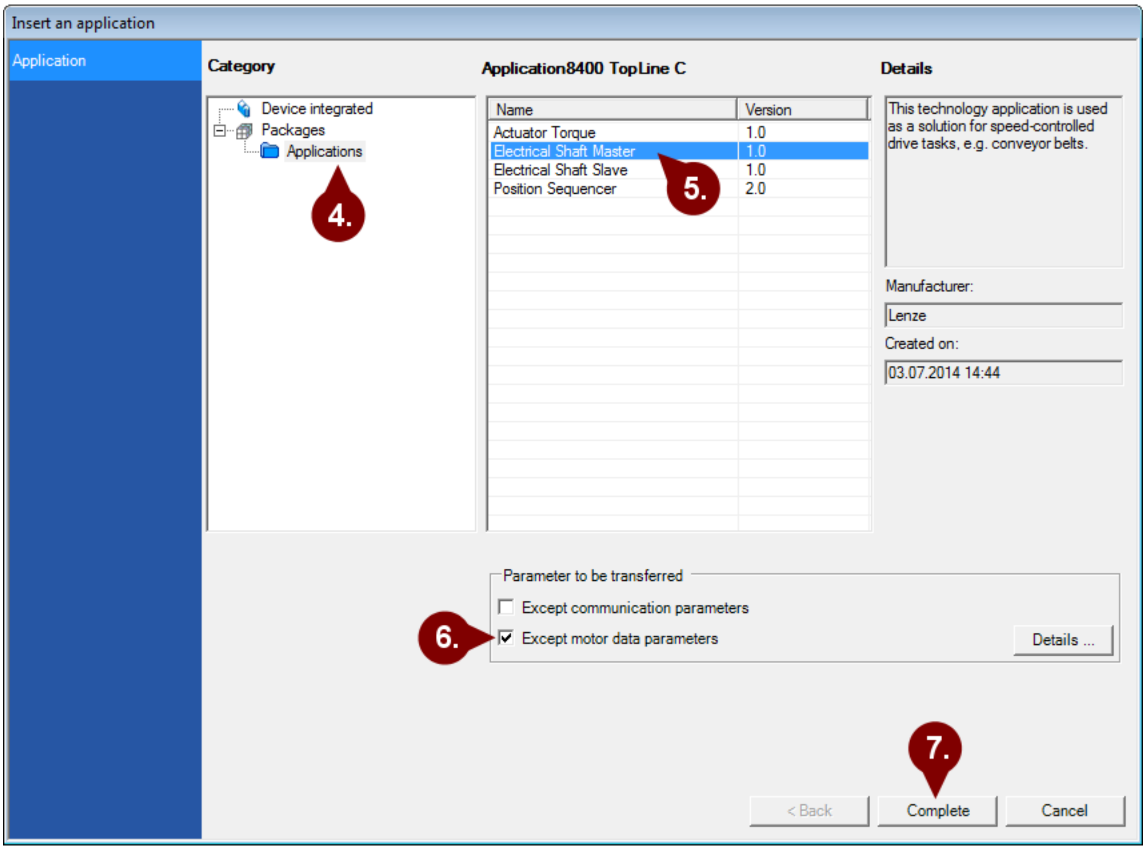
3.3.1 Step 1: Load "Electrical Shaft Master" technology application

In the Lenze setting, the inverter uses the "speed actuating drive" technology application integrated in the device. Execute the following steps to use the "Electrical Shaft Master" technology application instead:

1. Select the controller in the *Project view*.
2. If there is still an online connection to the controller:

   Click the icon to go offline again.
   (the application can only be selected offline.)

3. Click the icon to select another application.
   The *Insert application* dialog box appears:



4. In the left field, select the "Packages" → "Applications" category.
5. Select the "Electrical Shaft Master" application in the right field.
6. Activate the **Except motor data parameters** option in order that the settings of the motor data parameters made before will not be overwritten.
7. Press **Complete** to close the dialog box again and load the selected application into the »Engineer« project.

8. Confirm the prompt on whether the current application is to be replaced by the "Electrical Shaft Master" application with **Yes**.

3.3.2 Step 2: Axis bus settings for the master

Carry out the following settings for the inverter which is to be the master in the network:

- Axis bus address (C02430/1) = "1"
  - With this setting, the controller automatically takes over the control in the network.
  - Only one master is allowed in the network.
- Axis bus IO function (C02440/1) = "1: Master"
  - With this setting, the controller outputs a synchronisation cycle to the I/O axis bus to which the slaves can orient themselves.
- If the network is created by more than two inverters:
  Set the number of inverters connected to the axis bus in C02430/2.

3.3.3 Step 3 (optional): Set up control system via the fieldbus interface (MCI)

In the default setting, the application is controlled via the digital input terminals. For higher automated systems, mostly data bus systems are used for controlling the drives.

For control via the fieldbus interface (MCI), the user interface (area highlighted in turquoise) is to be adapted accordingly in the function block editor.

- The assignment of the outputs on the left to the inputs on the right can be changed at will.
- The inputs on the right are permanently linked to functions of the application.

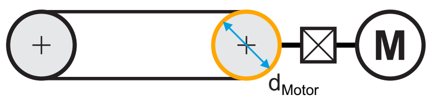
Tip!

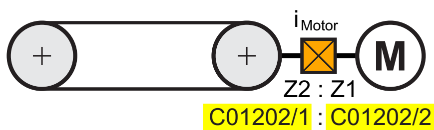
Control via the integrated CANopen interface ("CAN on board") can also be attained by means of simple interconnection changes. For this purpose, the **LP_CanIn** port block has to be inserted in the interconnection and connected to the corresponding inputs of the user interface.

3.3.4 Step 4: Set commissioning parameters

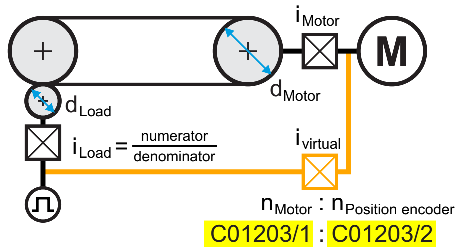
For a quick commissioning, only the following application-specific parameters have to be set or their default setting has to be checked!

- In order that you quickly find the respective parameterisation dialog in the FB Editor, the following table lists the block related to each parameter.
- The ▤ icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

Parameter (Block)	**Possible settings** (Lenze setting printed in bold)			**Info**
Machine parameters/axis settings				
C01201/1 (LS_MotionControl Kernel)	0.0000	units	214748.3647	Cycle length • For a modulo measuring system, the length of one cycle to the overflow is to be set. • For a limited measuring system, the Lenze setting "0.0000" is to be retained.
	Lenze setting: 0.0000 units			
C01204 (LS_MotionControl Kernel)	0.0001	units/rev.	214748.3647	Feed constant • The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft. • The value is entered in application units relating to one gearbox revolution. • Schematic diagram of a conveyor drive: $C01204 = \pi * d_{Motor}$ 
	Lenze setting: 360.0000 units/U			
C01206/1 (LS_MotionControl Kernel)				Motor mounting direction
	0	**not inverted**		Motor is mounted directly
	1	inverted		Motor is mounted with rotation by 180°

C01202/1..2
(LS_MotionControl Kernel)
	1		65535	Gearbox ratio motor - load
				• Set the gearbox ratio with mathematical precision in the two subcodes:
				  • Subcode 1: numerator term (Z2)
				  • Subcode 2: denominator term (Z1)
				$i_{motor} = \frac{C01202/1}{C01202/2} = \frac{Z2}{Z1}$
				• Schematic diagram of a conveyor drive:
				

	Lenze setting: 1:1			

Parameter (Block)	**Possible settings** (Lenze setting printed in bold)			**Info**
Position encoder				
If the motor encoder is used as position encoder, you do not need to parameterise a separate position encoder. Keep the Lenze setting for the following codes.				
C01206/2 (LS_MotionControl Kernel)				Position encoder mounting direction
	0	**not inverted**		Position encoder is mounted directly
	1	inverted		Position encoder mounted with rotation by 180
C01203/1..2 (LS_MotionControl Kernel)	1		65535	Speed ratio for motor - position encoder • Set the "virtual" speed ratio between the motor and the external position encoder with mathematical precision in the two subcodes: • Subcode 1: numerator term (motor speed) • Subcode 2: denominator term (encoder speed) • Schematic diagram of a conveyor drive:  • The "virtual" speed ratio can be calculated as follows: $i_{virtual} = \frac{C01203/1}{C01203/2} = \frac{i_{motor}}{i_{load}} \cdot \frac{\pi \cdot d_{load}}{\pi \cdot d_{motor}}$ $i_{virtual} = \frac{C01202/1}{C01202/2} \cdot \frac{denominator}{numerator} \cdot \frac{\pi \cdot d_{load}}{C01204}$
	Lenze setting: 1:1			
Selection of the speed setpoint for the application				
C00470/1 (LS_ParFree_b)				Selection of the source for the speed setpoint (also see Basic signal flow)
	0	**MCI interface**		Speed setpoint = process data input word 2 • Scaling: 16384 ≡ 100 % reference speed (C00011)
	1	Analog input 1		Speed setpoint = voltage signal at X3/A1U • Scaling: 10 V ≡ 100 % reference speed (C00011)
C00470/2 (LS_ParFree_b)				Selection of the source for the speed setpoint (also see Basic signal flow)
	0	**Master value**		Source set in C00470/1
	1	Multi encoder		Multi-Encoder at X8
C00470/7 (LS_ParFree_b)				Setpoint direction • Not relevant if multi-encoder is set as speed setpoint source in C00470/2.
	0	**not inverted**		
	1	inverted		
Settings for main setpoint via MCI interface or analog input 1				
C00012 (L_NSet_1)	0.000	s	999.999	Acceleration time - main setpoint • Not relevant if multi-encoder is set as speed setpoint source in C00470/2.
	Lenze setting: 2.000 s			
C00013 (L_NSet_1)	0.000	s	999.999	Deceleration time - main setpoint • Not relevant if multi-encoder is set as speed setpoint source in C00470/2.
	Lenze setting: 2.000 s			

Parameter (Block)	**Possible settings** (Lenze setting printed in bold)			**Info**
"Electrical shaft STOP" function				
C01040/1 (L_SRFG_1)	0.001	s	999.999	Linear ramp time for Electrical Shaft STOP • t_{ramp} = C01040/1 * speed [%] + C01041/1
	Lenze setting: 0.5 s			
C01041/1 (L_SRFG_1)	0.001	s	50.000	S-ramp time for Electrical Shaft STOP
	Lenze setting: 0.1 s			
Stretch factor				
C00471/1..2 (LS_ParFree)	0		65535	Stretch factor • Subcode 1: numerator term • Subcode 2: denominator term
	Lenze setting: 1000:1000			
Gearbox factor				
C00471/3..4 (LS_ParFree)	0		65535	Gearbox factor • Subcode 1: numerator term • Subcode 2: denominator term
	Lenze setting: 1000:1000			
Speed trimming				
C00476/1 (LS_ParFree_a_2)	-199.99	%	199.99	Speed trimming positive • Scaling: 100 % ≡ reference speed (C00011)
	Lenze setting: 3 %			
C00470/9 (LS_ParFree_b)				Activate positive speed trimming
	0	**No speed trimming**		
	1	Speed trimming by the value set in C00476/1		
C00476/2 (LS_ParFree_a_2)	-199.99	%	199.99	Speed trimming negative • Scaling: 100 % ≡ reference speed (C00011)
	Lenze setting: -3 %			
C00470/10 (LS_ParFree_b)				Activate negative speed trimming
	0	**No speed trimming**		
	1	Speed trimming by the value set in C00476/2		
Position offset (angular adjustment)				
C00475/1 (LS_ParFreeUnit_1)	-214748.3647	units	214748.3647	Position offset (X offset) The angular adjustment is retained as long as the position offset is present.
	Lenze setting: 0.0000 units			
C01060/2 (L_PosCtrlLin_1)	0.010	s	130.000	Acceleration ramp for the higher-level positioning of the X axis via the position offset set in C00475/1.
	Lenze setting: 1.000 s			
C01060/3 (L_PosCtrlLin_1)	0.010	s	130.000	Deceleration ramp for the higher-level positioning of the X axis via the position offset set in C00475/1.
	Lenze setting: 1.000 s			

Parameter (Block)	Possible settings (Lenze setting printed in bold)		Info
Selection of the signal for slave drives to be output via axis bus			
C00470/4 (LS_ParFree_b)			Operation as position follower master
	0	**Off**	The signals are output via axis bus according to the settings in C00470/5 and C00470/6.
	1	On	This setting is only required if the drive is to be used as master for a slave drive with the "Position Follower" technology application. The following signals are then output via axis bus: Line data words 1 & 2 = current position Line data word 3 = current speed **Note:** With this setting, homing of the master causes a step in the slave position follower if the current position is set to the home position! Remedy: Inhibit the slave position follower when you are referencing the master!
C00470/5 (LS_ParFree_b)			Selection of the speed setpoint for slave drives (also see Basic signal flow)
	0	**Mere master value**	Speed setpoint set in C00470/2
	1	Master value	Speed setpoint of the master drive's motion control (speed setpoint of the Motion Control Kernel). With this setting, the slave drives follow the master drive even if it is executing manual jog or homing.
C00470/6 (LS_ParFree_b)			Selection of the speed setpoint for slave drives (also see Basic signal flow)
	0	**Master value**	Master value set in C00470/5
	1	Actual speed	Current speed

3.3.5 Step 5: Go online and transfer parameter set to the inverter

In order to set the current parameter settings in the controller to the settings in the project, transmit the parameter set to the controller.

1. Click the icon to go online.
2. Click the icon to transmit the parameter set to the controller.
3. After a successful transmission, click the icon to save the parameter set safe against mains failure in the integrated Memory Module.

3.4 Short setup of the slave drive

The following steps are to be carried out for all slave drives of the interconnection.

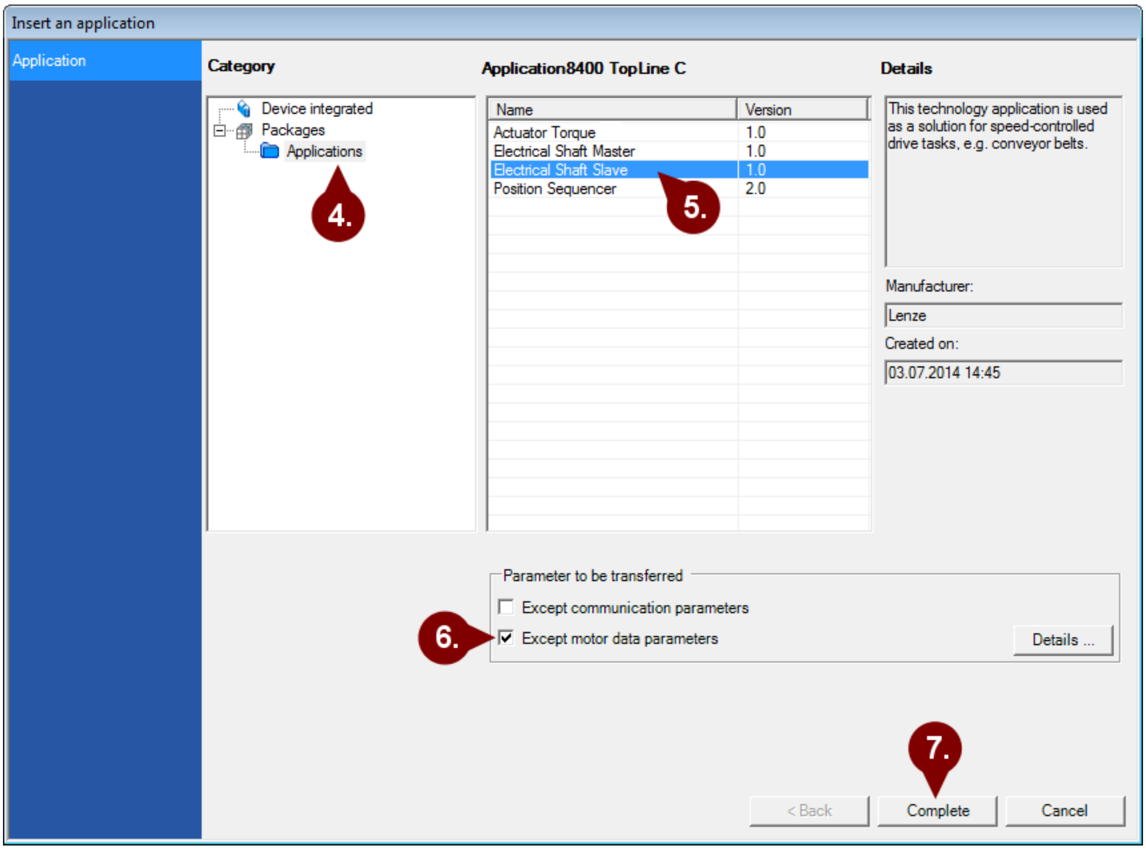
3.4.1 Step 1: Load "Electrical Shaft Slave" technology application

In the Lenze setting, the inverter uses the "speed actuating drive" technology application integrated in the device. Execute the following steps to use the "Electrical Shaft Slave" technology application instead:

1. Select the controller in the *Project view*.
2. If there is still an online connection to the controller:

   Click the icon to go offline again.
   (the application can only be selected offline.)

3. Click the icon to select another application.
   The *Insert application* dialog box appears:



4. In the left field, select the "Packages" → "Applications" category.
5. Select the "Electrical Shaft Slave" application in the right field.
6. Activate the **Except motor data parameters** option in order that the settings of the motor data parameters made before will not be overwritten.
7. Press **Complete** to close the dialog box again and load the selected application into the »Engineer« project.

8. Confirm the prompt on whether the current application is to be replaced by the "Electrical Shaft Slave" application with **Yes**.

3.4.2 Step 2: Axis bus settings for the slave

Make the following settings for every slave controller in the network:

- Axis bus address (C02430/1) = "2" ... "62"
  - Make sure that all controllers connected to the axis bus have different axis bus addresses.
- Sync signal source (C01120) = "2: AxisBusIO"
  - With this setting, the synchronisation cycle output by the master is used as synchronisation source. The parameters C01121 ... C01123 for the synchronisation of the internal time base are automatically set to fixed reasonable values to provide for a technically perfect operation of the axis bus!
  - Basically, only one source is allowed to synchronise the internal time base.
- Axis bus IO function (C02440/1) = "2: Slave"
  - This setting serves to define the slave.

3.4.3 Step 3 (optional): Set up control system via the fieldbus interface (MCI)

In the default setting, most functions are controlled via the digital input terminals.

Via the fieldbus interface (MCI), manual jog can be executed for the slave drive, and the X offset can be deactivated. If further functions are to be controlled via the fieldbus interface (MCI), the user interface (area highlighted in turquoise) must be adapted accordingly in the function block editor.

- The assignment of the outputs on the left to the inputs on the right can be changed at will.
- The inputs on the right are permanently linked to functions of the application.

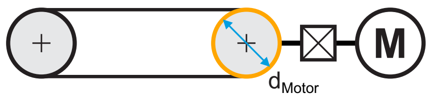
Tip!

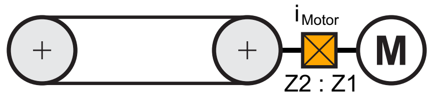
Control via the integrated CANopen interface ("CAN on board") can also be attained by means of simple interconnection changes. For this purpose, the **LP_CanIn** port block has to be inserted in the interconnection and connected to the corresponding inputs of the user interface.

3.4.4 Step 4: Set commissioning parameters

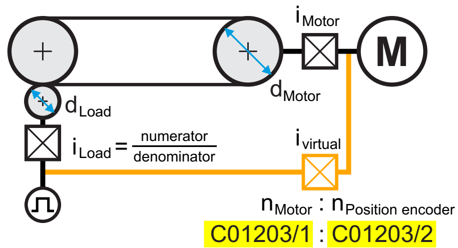
For a quick commissioning, only the following application-specific parameters have to be set or their default setting has to be checked!

- In order that you quickly find the respective parameterisation dialog in the FB Editor, the following table lists the block related to each parameter.
- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

Parameter (Block)	**Possible settings** (Lenze setting printed in bold)			**Info**
Machine parameters/axis settings				
C01201/1 (LS_MotionControl Kernel)	0.0000	units	214748.3647	Cycle length
	Lenze setting: 0.0000 units			• For a modulo measuring system, the length of one cycle to the overflow is to be set. • For a limited measuring system, the Lenze setting "0.0000" is to be retained.
C01204 (LS_MotionControl Kernel)	0.0001	units/rev.	214748.3647	Feed constant
	Lenze setting: 360.0000 units/U			• The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft. • The value is entered in application units relating to one gearbox revolution. • Schematic diagram of a conveyor drive: C01204 = π * d_{Motor} 
C01206/1 (LS_MotionControl Kernel)				Motor mounting direction
	0	**not inverted**		Motor is mounted directly
	1	inverted		Motor is mounted with rotation by 180°
C01202/1..2 (LS_MotionControl Kernel)	1		65535	Gearbox ratio motor - load

	Lenze setting: 1:1			• Set the gearbox ratio with mathematical precision in the two subcodes:
				  • Subcode 1: numerator term (Z2)
				  • Subcode 2: denominator term (Z1)
				$i_{Motor} = \frac{C01202/1}{C01202/2} = \frac{Z2}{Z1}$
				• Schematic diagram of a conveyor drive:
				

Parameter (Block)	Possible settings (Lenze setting printed in bold)			Info
Position encoder				
If the motor encoder is used as position encoder, you do not need to parameterise a separate position encoder. Keep the Lenze setting for the following codes.				
C01206/2 (LS_MotionControl Kernel)				Position encoder mounting direction
	0	**not inverted**		Position encoder is mounted directly
	1	inverted		Position encoder mounted with rotation by 180
C01203/1..2 (LS_MotionControl Kernel)	1		65535	Speed ratio for motor - position encoder

	Lenze setting: 1:1			• Set the "virtual" speed ratio between the motor and the external position encoder with mathematical precision in the two subcodes:
				  • Subcode 1: numerator term (motor speed)
				  • Subcode 2: denominator term (encoder speed)
				• Schematic diagram of a conveyor drive:
				
				• The "virtual" speed ratio can be calculated as follows:
				$i_{virtual} = \frac{C01203/1}{C01203/2} = \frac{i_{Motor}}{i_{load}} \cdot \frac{\pi \cdot d_{load}}{\pi \cdot d_{Motor}}$
				$i_{virtual} = \frac{C01202/1}{C01202/2} \cdot \frac{denominator}{numerator} \cdot \frac{\pi \cdot d_{load}}{C01204}$

Error handling				
C00470/4 (LS_ParFree_b)				Error message to the master For commissioning of the slave drive, the error message to the master via the IO axis bus is deactivated at first.
	0	**deactivated**		
	1	activated		
Selection of the speed setpoint for the application				
C00470/1 (LS_ParFree_b)				Selection of the source for the speed setpoint
	0	**Axis bus**		Master drive
	1	Multi encoder		Multi-Encoder at X8
Stretch factor				
C00471/1..2 (LS_ParFree)	0		65535	Stretch factor • Subcode 1: numerator term • Subcode 2: denominator term
	Lenze setting: 1000:1000			
Gearbox factor				
C00471/3..4 (LS_ParFree)	0		65535	Gearbox factor • Subcode 1: numerator term • Subcode 2: denominator term
	Lenze setting: 1000:1000			
Speed trimming				
C00476/1 (LS_ParFree_a_2)	-199.99	%	199.99	Speed trimming positive • Scaling: 100 % ≡ reference speed (C00011)
	Lenze setting: 3 %			
C00476/2 (LS_ParFree_a_2)	-199.99	%	199.99	Speed trimming negative • Scaling: 100 % ≡ reference speed (C00011)
	Lenze setting: -3 %			

Parameter (Block)	**Possible settings** (Lenze setting printed in bold)			**Info**
Position offset (angular adjustment)				
C00475/1 (LS_ParFreeUnit)	-214748.3647	units	214748.3647	Position offset (X offset) The angular adjustment is reatined as long as the position offset is present and the slave is running in electrical shaft operation.
	Lenze setting: 0.0000 units			
C01060/2 (L_PosCtrlLin_1)	0.010	s	130.000	Acceleration ramp for the higher-level positioning of the X axis via the position offset set in C00475/1.
	Lenze setting: 1.000 s			
C01060/3 (L_PosCtrlLin_1)	0.010	s	130.000	Deceleration ramp for the higher-level positioning of the X axis via the position offset set in C00475/1.
	Lenze setting: 1.000 s			
Mark synchronisation/TP correction				
C00470/3 (LS_ParFree_b)				Mark synchronisation/TP correction
	0	**Off**		
	1	On		Permanent zero pulse/touch probe synchronisation. The correction is carried out by the shortest path. (Other synchronisation modes can be selected in C01075/1)
C01069/2 (L_DFSET_1)	0	Incr./ ms	32767	Correction width (ramp for angle compensation) The increments for the angle compensation in the case of mark synchronisation are specified in the Lenze setting with downstream 3rd-order polynomial.
	Lenze setting: 100 incr./ms			

3.4.5 Step 5: Go online and transfer parameter set to the inverter

In order to set the current parameter settings in the controller to the settings in the project, transmit the parameter set to the controller.

1. Click the icon to go online.
2. Click the icon to transmit the parameter set to the controller.
3. After a successful transmission, click the icon to save the parameter set safe against mains failure in the integrated Memory Module.

3.5 Enabling the interconnection and specifying the speed setpoint

After the parameter set has been transmitted to the inverter, the inverter can now be enabled and the control signals/setpoints can be selected via the corresponding interfaces.

▸ Pre-assignment of the user interface for the master drive (📖 14)
▸ Pre-assignment of the user interface for the slave drive (📖 17)

3.6 (Optionally): Setting the optimisation parameters

The following application-specific parameters are used for optimisation and can also be adapted during operation.

Stop!

If you change parameters in the »Engineer« during an online connection to the device, the changes are directly transferred to the device!

Tip!

Do not forget to save the parameter changes carried out with mains failure protection in the memory module implemented! (C00002/11 = "1: on/start")

3.6.1 Optimisation parameters for the master drive

The following parameters have already been set in step 4. However, depending on the application it may be required to change the settings again during operation:

Parameter (Block)	Possible settings			Info
Settings for main setpoint via MCI interface or analog input 1				
C00012 (L_NSet_1)	0.000	s	999.999	Acceleration time - main setpoint • Not relevant if multi-encoder is set as speed setpoint source in C00470/2.
	Lenze setting: 2.000 s			
C00013 (L_NSet_1)	0.000	s	999.999	Deceleration time - main setpoint • Not relevant if multi-encoder is set as speed setpoint source in C00470/2.
	Lenze setting: 2.000 s			
"Electrical shaft STOP" function				
C01040/1 (L_SRFG_1)	0.001	s	999.999	Linear ramp time for Electrical Shaft STOP • t_{ramp} = C01040/1 * speed [%] + C01041/1
	Lenze setting: 0.5 s			
C01041/1 (L_SRFG_1)	0.001	s	50.000	S-ramp time for Electrical Shaft STOP
	Lenze setting: 0.1 s			

3.6.2 Optimisation parameters for the slave drive

Parameter (Block)	Possible settings			Info
Following error monitoring system				
C00254 (LS_MotorInterface)	0.00	1/s	500.00	Kp position controller • Gain for following error compensation
	Lenze setting: 5.00/s			
C01215/1 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 1 • The setting "0" deactivates following error monitoring 1
	Lenze setting: 5.0000 units			
C01215/2 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 2 • The setting "0" deactivates following error monitoring 2
	Lenze setting: 10.0000 units			

4 Detailed functions of the technology application

4.1 Basic drive functions (MCK)

Danger!

During homing, manual jog, and positioning, specially assigned profile parameters are active. If they have not been set correctly, the drive may carry out an unexpected movement!

Detailed information relating to the basic drive functions and the corresponding parameters can be found in the reference manual/online help of the inverter in the "Basic drive functions (MCK)" chapter.

4.1.1 Homing

In some applications it is required that the master and slave drive are aligned with each other before the actual synchronous operation starts. This can for instance be effected by homing or a reference search. This so-called "homing" is usually carried out after mains connection. Furthermore, homing after the elimination of an encoder error is required.

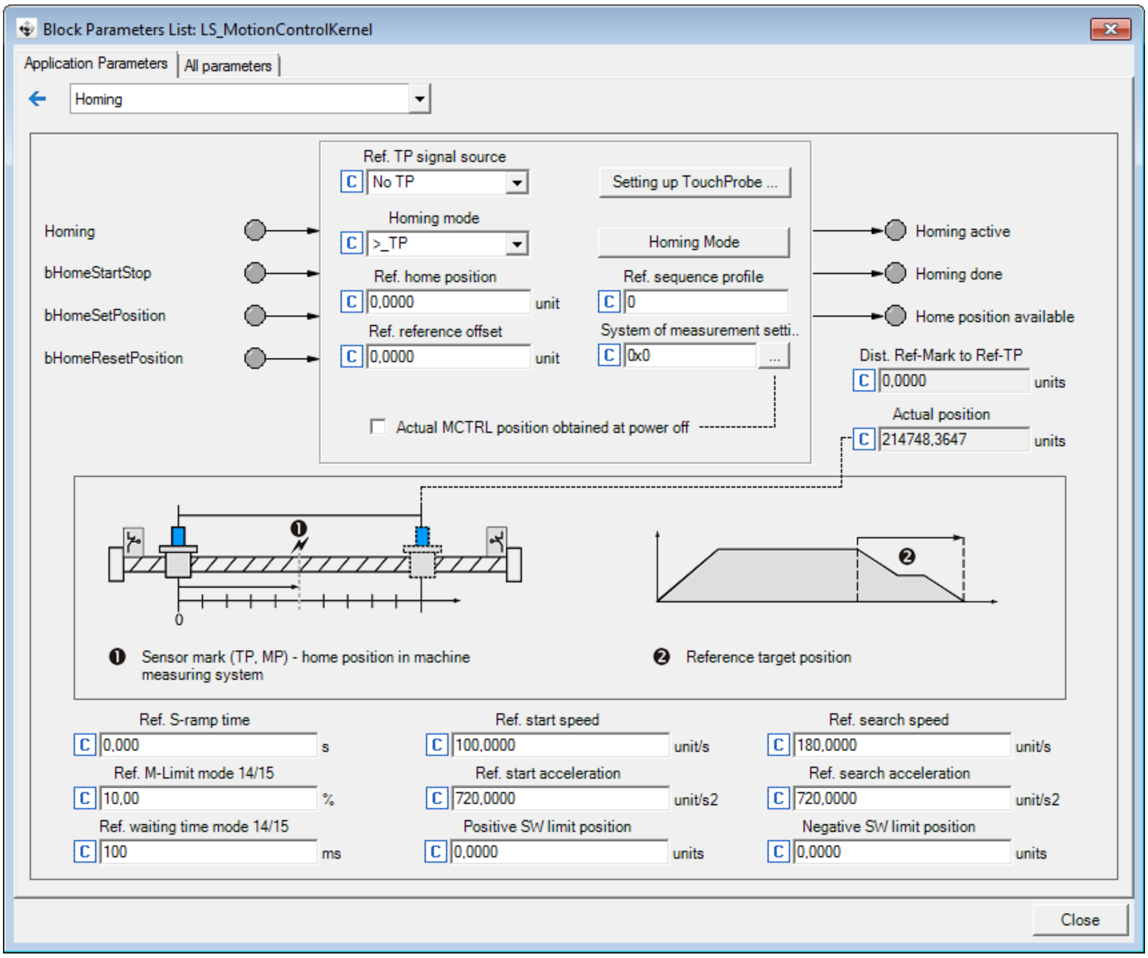
Note!

Depending on the configuration of the master drive, the slave will follow the master during a homing process, or it won't. Therefore it depends on the application whether it is reasonable to, for example, first reference the master and then the slave. If the slave is not to follow the master during the homing process, quick stop (QSP) is to be activated for the slave.

- "Manual" start/stop of the homing process is executed via bit 6 of the application control word (FB **L_ConvBitsToWord_1**).
  - In the case of the master, this bit is linked with digital input DI1 in the default setting.
  - In the case of the slave, this bit is linked with digital input DI6 in the default setting.
- Homing is to be configured according to the requirement of the application, as described in the reference manual/online help of the inverter.
- If the home position is known to the drive, a response is sent via digital output DO3 and bit 15 of the MCI process data output word 1.

How to go to the parameterisation dialog of the basic "Homing" function:

1. Open the parameter list for the **LS_MotionControlKernel** SB.
2. In the *Block Parameters List...* dialog box on the **Application Parameters** tab, select the "Homing" entry in the upper list field.



Note!

For a reference search with touch probe detection:

If the reference signal is to follow a real touch probe, the touch probe interface must be configured accordingly via the **Setting up TouchProbe...** button!

Parameter	Info	Lenze setting	
		Value	Unit
C01221	MCK: Homing mode	8: >_TP	
C01224/1	MCK: Ref. initial speed	100.0000	unit/s
C01225/1	MCK: Ref. initial acceleration	720.0000	unit/s2
C01224/2	MCK: Ref. search speed	180.0000	unit/s
C01225/2	MCK: Ref. search acceleration	720.0000	unit/s2
C01226/1	MCK: Ref. S-ramp time	0.000	s

Parameter	Info	Lenze setting	
		Value	Unit
C01222	MCK: Ref. M limit mode 14/15	10.00	%
C01223	MCK: Ref. waiting time mode 14/15	100	ms
C01227/1	MCK: Ref. offset reference degree	0.0000	unit
C01227/2	MCK: Ref. home position	0.0000	unit
C01228	MCK: Ref. sequence profile	0	
C01229/1	MCK: Positive SW limit position	0.0000	units
C01229/2	MCK: Negative SW limit position	0.0000	units
C01246/1	MCK: Ref. TP signal source	0: No TP	

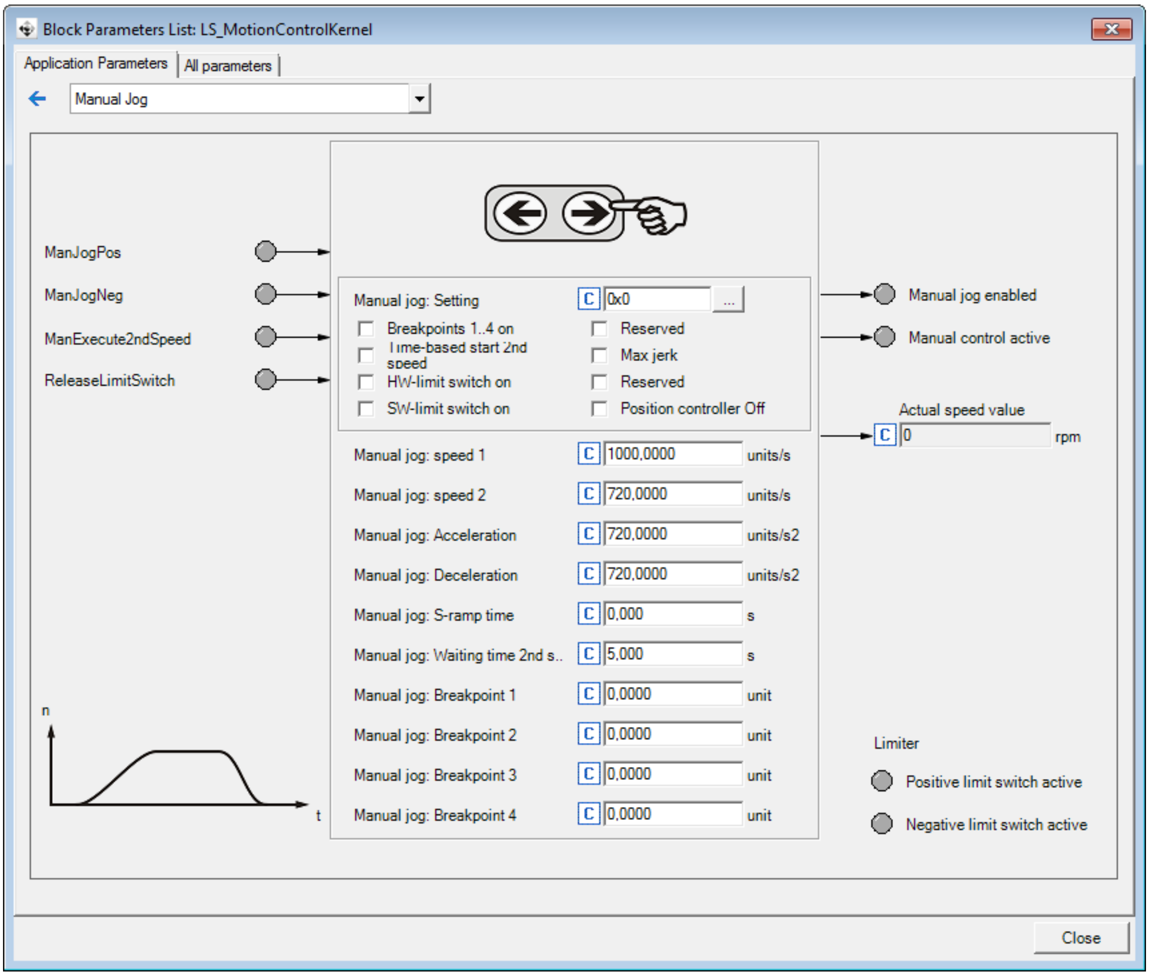
4.1.2 Manual jog

In this operating mode, the drive can be traversed manually in a clockwise or anticlockwise direction ("jogging mode").

- The basic "Manual jog" function is activated via bit 12 and bit 13 of the application control word (FB **L_ConvBitsToWord_1**).
  - In the case of the master, these two bits are linked with digital inputs DI5 and DI6 in the default setting.
  - In the case of the slave, these two bits are linked bit 12 and bit 13 of the MCI process data input word 1 in the default setting.

How to go to the parameterisation dialog of the basic "Manual jog" function:

1. Open the parameter list for the **LS_MotionControlKernel** SB.
2. In the *Block Parameters List...* dialog box on the **Application Parameters** tab, select the "Manual Jog" entry in the upper list field.



Parameter	Info	Lenze setting	
		Value	Unit
C01230	MCK: Manual jog setting	Bit coded	
C01231/1	Manual jog: speed 1	1000.0000	units/s
C01231/2	Manual jog: Speed 2	720.0000	units/s
C01232/1	Manual jog: Acceleration	720.0000	units/s2
C01232/2	Manual jog: Deceleration	720.0000	units/s2
C01233/1	Manual jog: S-ramp time	0.000	s
C01235/1	Waiting time 2nd speed	5.000	s
C01234/1	Manual jog: Breakpoint 1	0.0000	unit
C01234/2	Manual jog: Breakpoint 2	0.0000	unit
C01234/3	Manual jog: Breakpoint 3	0.0000	unit
C01234/4	Manual jog: Breakpoint 4	0.0000	unit

4.1.3 Holding brake control

This basic function is used for low-wear control of a holding brake.

Danger!

Please note that the holding brake is an important element of the safety concept of the entire machine.

Thus, proceed very carefully when commissioning this system part!

In order to ensure that the brake is not released unintentionally while the application is transferred to the inverter, we recommend unplugging the brake connector X107 during the transfer.

Detailed information on how to parameterise the holding brake control can be found in the reference manual/online help of the controller in the chapter "Basic drive functions (MCK)".

The documentation of the holding brake control contains safety instructions which must be observed!

Application-specific notes on the holding brake control:

- In the Lenze setting, the mode 0 (brake control off) is preset in C02580.
- The technology application is prepared for the control of a 24-V holding brake via the high current output.
- In the "Homing" and "Manual jog" operating modes, the holding brake is released automatically by the MCK. In the "Speed follower" operating mode, release of the brake is effected by the application. For the master by the **L_DigitalLogic5_1** FB, for the slave by FBs **L_Or5_2** and **L_Not_2**.
- For the slave drive, the switching threshold for releasing the brake is set in C02581/1.

4.2 Functions in the master drive

4.2.1 Selection of the setpoint signal for the slave drives

For the speed/position output to the slave drives, SB **LS_AxisBusOut** is provided in the master application.

- In the default setting, the speed setpoint integrated to an angle (path) is output via line data words 1 & 2, and the speed setpoint is output via line data word 3.
- A slave drive connected via axis bus with the "Electrical Shaft Slave" technology application is executable immediately after the initialisation of the axis bus if no fault is pending.
  - If a fault is pending in the slave drive, the cause of which, however, has been eliminated, this error can be reset via reset of the master.
- If the drive is to be used as master for a slave drive with the "Position Follower" technology application, set the value "1" in C00470/4. Then, instead of the speed integrated, the current position is output via axis bus.

Stop!

If operation as master position follower is switched on (C00470/4 = "1"), homing of the master causes a step for the slave position follower if the current position is set to the home position!

Remedy: Inhibit the slave position follower when you reference the master!

The signal to be output via axis bus is selected using C00470/4...6:

<table>
<tr><th colspan="2">C00470/4</th><th colspan="2">C00470/6</th><th colspan="2">C00470/5</th><th rowspan="2">Axis bus output</th></tr>
<tr><td colspan="2">Operation as position follower master</td><td colspan="2">Selection of master value/
actual speed</td><td colspan="2">Selection of mere master value</td></tr>
<tr><td rowspan="3">0</td><td rowspan="3">Off</td><td rowspan="2">0</td><td rowspan="2">Master value</td><td>0</td><td>Mere master value</td><td>Line data words 1 & 2 = speed setpoint integrated to an angle (path)
Line data word 3 = speed setpoint</td></tr>
<tr><td>1</td><td>Master value</td><td>Line data words 1 & 2 = speed setpoint integrated to an angle (path) from the MCK
Line data word 3 = speed setpoint from the MCK</td></tr>
<tr><td>1</td><td colspan="3">Actual speed</td><td>Line data words 1 & 2 = current speed integrated to an angle (path)
Line data word 3 = current speed</td></tr>
<tr><td>1</td><td colspan="5">On</td><td>Line data words 1 & 2 = current position
Line data word 3 = current speed</td></tr>
</table>

Details on the signal flow

Switch-over between the different setpoint signals for the slave drives is effected via FBs **L_SignalSwitch_2...4**.

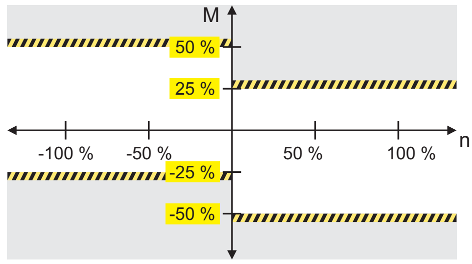
In the default setting C00470/4 = 0, the position value for SB **LS_AxisBusOut** is generated with the **L_PhaseIntK_1** FB. Loading or a reset is not required, since in the slave drive merely consistency with the speed value is compared without attaching importance to the absolute value.

With the setting C00470/4 = 1 (operation as master position follower), as position value for SB **LS_AxisBusOut** the current position *dnMotorPosAct_p* is used by the **LS_MotorInterface** SB.

4.2.2 "Electrical shaft STOP" function

When the "Electrical shaft STOP" function is activated in the master, all slave drives enabled remain synchronous, since this function has a direct impact on the master value path. When the "Quick stop (QSP)" function is activated in the master, however, it depends on the setting in C00104/1 whether the drive is brought to a standstill in an angle-controlled manner or not.

A further characteristic of the "Electrical shaft STOP" function is that a torque limitation which is possibly set via the following parameters is continued to be taken into consideration:

Parameter (Block)	**Possible settings**			**Info**
Torque limitation in motor mode/in generator mode				
The torque limitation set is always active.				Example: Definition of the torque limitations C00472/3 = 25 % C00472/4 = 50 % 
C00472/3 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in motor mode
	Lenze setting: 100 % maximum torque (C00057)			
C00472/4 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in generator mode
	Lenze setting: 100 % maximum torque (C00057)			

Tip!

Depending on the application, the "Electrical Shaft Stop" function can be used for an operation stop, and the "quick stop (QSP)" function for an emergency stop.

4.3 Functions in the slave drive

4.3.1 Mark synchronisation/TP correction

For a mark synchronisation/TP correction, FB **L_DFSET_1** is provided to the slave application. This FB enables the slave drives to synchronise to the master in a "flying" manner.

Preconditions for a mark synchronisation

- The mark synchronisation is switched on (C00470/3 = "1: True").
- The touch probe sensor for the mark synchronisation of the slave drive is connected to digital input DI4.
- The master pulse for the mark synchronisation is connected to digital input DI5.

Operating mode

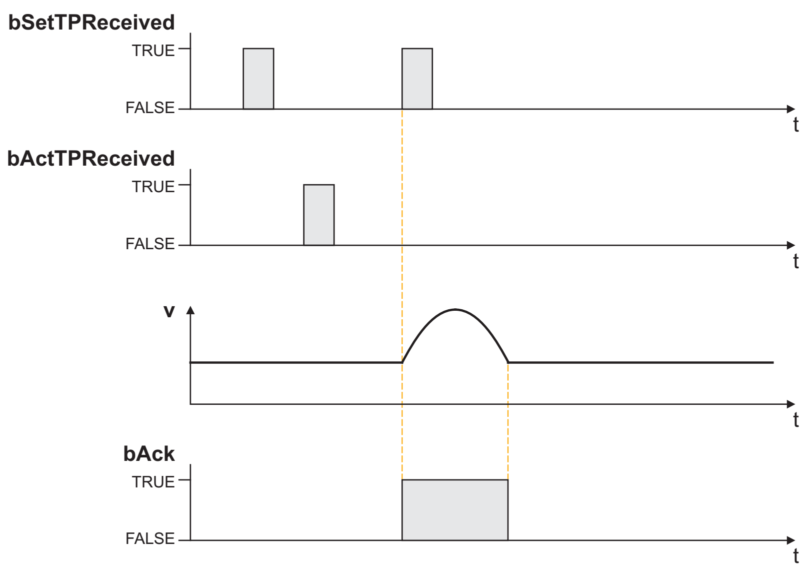
For the synchronisation, the slave drive needs a cyclic or one-time pulse generated by the master drive and provided by digital input DI5. This "setpoint pulse" is either the zero pulse of the master feedback, or the edge of a touch probe sensor. Via the setpoint pulse, the drive position of the master drive is communicated to the slave drive.

In addition, digital input DI4 must provide the slave drive with an "actual value pulse" from a touch probe sensor, recording the marks.

Only if FB **L_DFSET_1** receives both signals, it can execute a synchronisation!

In the synchronisation mode "2" that is preset in C01075/1, a full synchronisation including a correction by the shortest path is executed when the mark synchronisation is switched on (C00470/3 = "1: True"). The synchronisation is only carried out when two setpoint pulses have been received (pulse sequence required: setpoint - actual - setpoint):

Synchronisation procedure



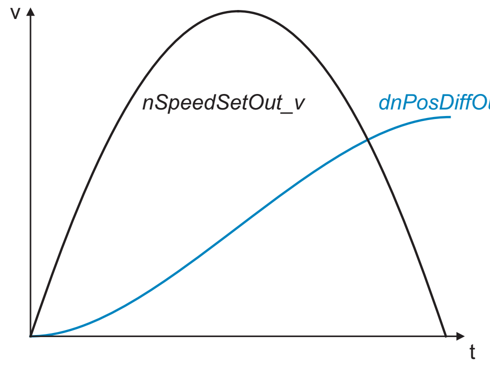
bSetTPReceived	Setpoint pulse
bActTPReceived	Actual value pulse
bAck	Status signal "Synchronising is executed"

Compensation process

At the input of the second setpoint or actual value pulse, the difference between the master and slave psition is determined. This difference is then transferred via the speed output *nSpeedSetOut_v* as an additive offset to the **Motion Control Kernel**. The compensating movement is smoothened by means of a polynomial.

The following illustration shows a compensating process after the marks have been detected. The speed is added here as a parabola to the line speed. (The scaling in this diagram does not correspond to reality)

Example: Compensation process



Detailed information about the **L_DFSET_1** FB can be found in the reference manual/online help of the inverter in the "Function library" chapter.

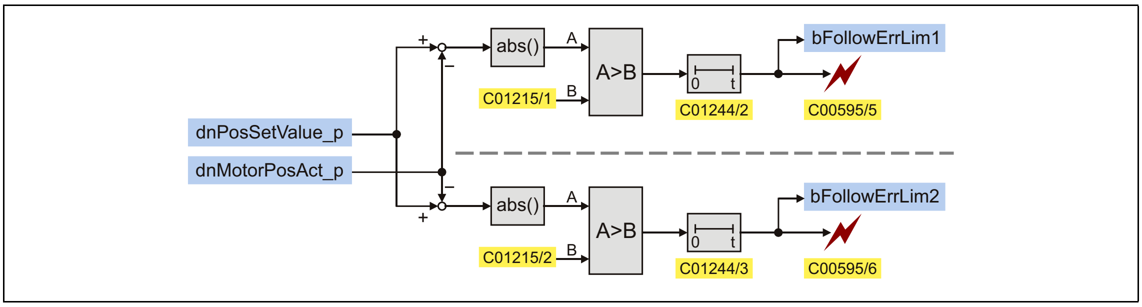
4.4 Monitoring functions

4.4.1 Following error monitoring system

The difference between set position and actual position is called the following error. Ideally, the following error should be "0". The set position is created by the internal definition of the traversing profiles of the **Motion Control Kernel**. The actual position is created by the integration of the speed supplied by the position encoder. If the position control is adjusted optimally, only a minimum following error arises which is always compensated dynamically and not increases continuously.

Certain processes, however, require that a defined limit as a difference between set position and actual position is not exceeded. If it is exceeded, it may have been caused by a mechanical blocking in the machine and the system part is not situated at the position defined at that time. In such a case, it makes sense to activate the "Fault" error response to make the motor torqueless.

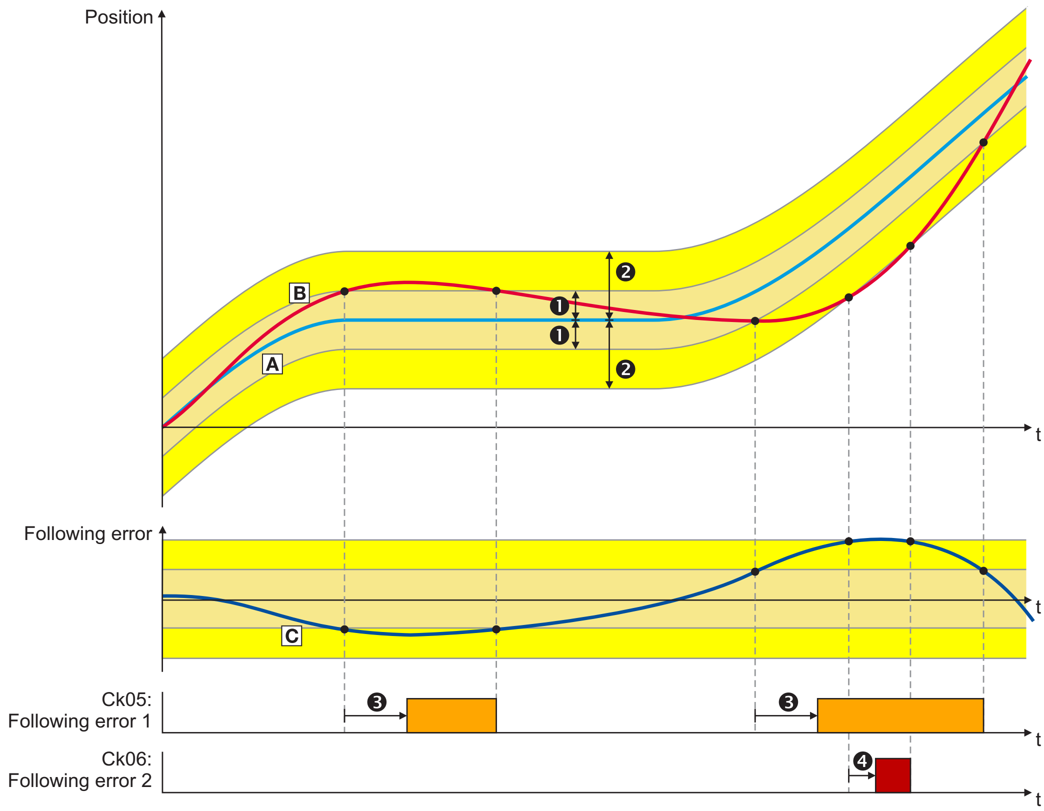
In the 8400 TopLine controller, two independent following error monitoring systems can be parameterised:



[4-1] Two-channel following error monitoring system

Parameter (Block)	Possible settings			Info
C01215/1 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 1 • The setting "0" deactivates following error monitoring 1
	Lenze setting: 5.0000 units			
C01215/2 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 2 • The setting "0" deactivates following error monitoring 2
	Lenze setting: 10.0000 units			
C01244/2...3 (LS_MotionControl Kernel)	0	ms	600000	Waiting time for following error monitoring 1 & 2 • In order to avoid that an error is triggered by acceleration and a narrow tolerance limit can be nevertheless monitored at standstill in the target, the response of the following error monitoring system can be delayed by setting a waiting time.
	Lenze setting: 0 ms			
C00595/5...6 (LS_SetError_1)				Response at the activation of following error monitoring • Subcode 5: response at the activation of following error monitoring 1. • Subcode 6: response at the activation of following error monitoring 2.
	0	No Reaction		
	1	Fault		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	**Warning**		
	6	Information		

Example of the evaluation of the following error



A	Target position	❶	Limit for following error monitoring 1 (C01215/1)
B	Position actual value	❷	Limit for following error monitoring 2 (C01215/2)
C	Current following error	❸	Waiting time for following error monitoring 1 (C01244/2)
		❹	Waiting time for following error monitoring 2 (C01244/3)

The above example clearly shows that there are two monitoring functions acting independently of each other. Normally, one following error monitoring function with a low tolerance is set as warning, and the other with a higher tolerance is set as TroubleQuickStop.

By means of the waiting times (C1244/2 and C1244/3), the error responses can be delayed, so that disconnection does not result immediately when the following error limits are temporarily exceeded. This "switch-on delay" makes it possible to effectively distinguish between dynamic processes and actual error states such as mechanical obstacles or sluggishness.

4.4.2 Limit position monitoring

For safety reasons, drives with a limited traversing range must always be made safe using corresponding safety mechanisms. These are firstly the hardware limit switches, which must effect a standstill of the axes as quickly as possible during the approaching process. In addition, the parameterisable software limit positions are to inhibit all travel commands which would entail an exit of the permissible travel range.

Note!

In the two technology applications "Electrical Shaft Master" and "Electrical Shaft Slave", an evaluation of hardware limit switches is not prepared by default yet, but it can be retrofitted in a few steps if required (see "Hardware limit switch" subchapter).

Take the following items into consideration for placing the limit positions:

- The hardware limit switches must be mounted with a sufficient clearance in front of the mechanical limit positions of the linear axis. The clearance to the mechanical limit position should be calculated for a quick stop (QSP) from maximum speed, so that in the event of an error, standstill is attained safely before the mechanical limit position is reached.
- The software limit positions must be placed with a sufficient distance from the hardware limit switch positions. The distance should be calculated for a quick stop (QSP) from maximum speed without the hardware limit switches being approached.

4.4.2.1 Hardware limit switch

For an evaluation of hardware limit switches, the two digital inputs in the function block editor that are used for the connection of the hardware limit switches have to be connected to the monitoring inputs *bLimitSwitchNeg* and *bLimitSwitchPos* of the **LS_MotionControlKernel** system block.

The following example exemplarily shows the connections required if the negative limit switch is connected to digital input DI1 and the positive limit switch is connected to digital input DI2:

1. **LS_DigitalInput**.*bIn1* → **LS_MotionControlKernel**.*bLimitSwitchNeg*
2. **LS_DigitalInput**.*bIn2* → **LS_MotionControlKernel**.*bLimitSwitchPos*

Stop!

The connection of the limit switch signals must be executed in a fail-safe fashion:

- LOW level = limit switch approached (activated)
- HIGH level = limit switch not approached (not activated)

To obtain this switching logic, the level inversion for the two digital inputs to which the limit switches are connected has to be activated in C00114!

The limit switches are only evaluated if the limit switches for the respective operating mode have been activated (see the following table)!

Operating mode	Hardware limit switch effective
Speed follower	Yes (adjustable in C01219 - bit 2)
Homing	Depending on the homing mode selected (see description of the homing modes in the reference manual/online help of the inverter)
Manual jog	No (adjustable in C01230 - bit 2)

Operating mode	Hardware limit switch effective
Positioning	Yes
Stop	Yes
Position follower	Yes (adjustable in C01218 - bit 2)

Parameter (Block)	Possible settings		Info
C00595/1...2 (LS_MotionControl Kernel)			Response at the activation of limit position monitoring • Subcode 1: response at the approach of the positive limit switch. • Subcode 2: response at the approach of the negative limit switch.
	0	No Reaction	
	1	Fault	
	3	**TroubleQuickStop**	
	4	WarningLocked	
	5	Warning	
	6	Information	

Behaviour when hardware limit switches are active

If one of the two monitoring inputs is set to TRUE, in the Lenze setting the "TroubleQuickStop" error response is triggered: Irrespective of the setpoint selection, the drive is brought to a standstill in the deceleration time set for the quick stop function. Depending on the error response parameterised, the drive can then only be traversed again after the error has been acknowledged.

4.4.2.2 Software limit positions

The parameterisable limit positions are used by the software to limit the traversing range.

Stop!

The software limit positions are only evaluated and monitored if the home position is known to the drive and the software limit positions for the respective operating mode have been activated (see following table)!

Operating mode	Software limit positions active (if home position is known)
Speed follower	Yes (adjustable in C01219 - bit 3)
Homing	Yes
Manual jog	Yes (adjustable in C01230 - bit 3)
Positioning	Yes
Position follower	Yes (adjustable in C01218 - bit 3)

Parameter (Block)	Possible settings			Info
C01229/1...2 (LS_MotionControl Kernel)	-214748.3647	units	214748.3647	Positive and negative software limit position for limiting the valid traversing range • The positive software limit position must be set to a greater value than the negative software limit position!
	Lenze setting: 0.0000 units			

Parameter (Block)	Possible settings		Info
C00595/3...4 (LS_MotionControl Kernel)			Response at the activation of limit position monitoring • Subcode 3: response at overtravelling the positive software limit position (C01229/1). • Subcode 4: response at overtravelling the negative software limit position (C01229/2).
	0	No Reaction	
	1	Fault	
	3	**TroubleQuickStop**	
	4	WarningLocked	
	5	Warning	
	6	Information	

Behaviour in the case of active software limit positions

Note!

The "travel commands" mentioned in the following description are no speed setpoint selections. In the "Speed follower" and "Position follower" operating modes, an acknowledged software limit position error ensures that traversing to the impermissible travel range remains possible afterwards. This is because in these two operating modes, there is no preview of whether a software limit position is approached with a setpoint selection.

If the software limit positions are active, travelling commands that would result in the exit from the permissible travel range can no longer be executed.

If the drive is already outside the permissible travel range and the software limit positions have been activated, only travel commands that result in the drive moving back into the permissible travel range can be executed.

If the software limit positions are active, and one of the software limit positions is overtravelled, in the Lenze setting the "TroubleQuickStop" error response is triggered: Irrespective of the setpoint selection, the drive is brought to a standstill in the deceleration time set for the quick stop function. Depending on the error response parameterised, the drive can then only be traversed again after the error has been acknowledged.

4.4.3 Bus monitoring

When an "electrical shaft" is executed via axis bus, or generally via a bus system, one always has to expect data transmission errors, for example due to EMC interference. In order to prevent a mechanical offset of the drives in the case of even a temporary fault, the **L_Interpolator_1** FB is used in the slave application. This FB is provided with an internal correction mechanism which arranges for an automatic offset correction if a data telegram is missing. Like this, the axis bus monitoring time (C02431/3) can be set to an insensitive value, so that a very robust transmission system is created.

Parameterisation of the error handling:

Parameter	Possible settings		Info
C00591/1			Resp. to axis bus data error • Response in the case of an accumulation of data errors
	0	No Reaction	
	1	**Fault**	
	2	Trouble	
	3	TroubleQuickStop	
	4	WarningLocked	
	5	Warning	
	6	Information	
C00591/2			Resp. to axis bus IO error • Response in the case of an I/O error ("release cord" function)
	0	**No Reaction**	
	1	Fault	
	2	Trouble	
	3	TroubleQuickStop	
	4	WarningLocked	
	5	Warning	
	6	Information	
C02431/3	0 ms 65000		Axis bus monitoring time Tolerance time before the data error is triggered
	Lenze setting: 1000 ms		
C02444/1			Axis bus configuration
	Bit 0	Edge detection SetFail	**1** ≡ input *bSetFail_DigOut* is edge-sensitive. 0 ≡ input *bSetFail_DigOut* is level-sensitive.
	Bit 1	Reserved	Not relevant for bus monitoring
	...	Reserved	
	Bit 13	Reserved	
	Bit 14	same CobId for all Cas	
	Bit 15	500 kbps	

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

Lenze Drives GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
+49 5154 82-0
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Inverter

8400

"Position Follower" technology application
for 8400 TopLine C

Software manual EN

13467301

Lenze

Contents

1 About this documentation

This documentation described the software-based solution of a task. The transferability of the described solution to the respective application case needs to be checked by the user. If required, the user has to adapt the solution accordingly. Thus, physical aspects as e.g. drive dimensioning is not part of this documentation.

Danger!

The controller is a source of danger which may lead to death or severe injury of persons.

To protect yourself and others against these dangers, observe the safety instructions before switching on the controller.

Please read the safety instructions provided in the **8400 mounting instructions** and in the **8400 hardware manual**. Both documents are supplied with the controller.

Target group

This documentation addresses to all persons

- who want to use the "Position Follower" technology application for the 8400 TopLine inverter, and
- who are familiar with handling the device and the »Engineer« software.

Validity

The information in this documentation are valid for the following technology applications:

Technology application	from version
Position Follower	1.0

Screenshots/application examples

All screenshots provided in this documentation are application examples. Depending on the software version of the controller and the version of the installed »Engineer« software, the screenshots in this documentation may differ from the representation in the »Engineer«.

Tip!

Information and tools for Lenze products are provided in the download area at

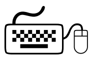
http://www.lenze.com → Download

1.1 Document history

Version			Description
1.0	07/2014	TD05	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is generally used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the prefix "0x" is used. Example: 0x60F4
Binary number	0b	For binary numbers, the prefix "0b" is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that only applies to a certain controller software version or higher is identified accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software ...
Window	*italics*	The *Message* window ... / The *Options* dialog box...
Variable name		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Press **<F1>** to open the online help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...
Hyperlink	underlined	Optically highlighted reference to another topic. In this documentation activated by mouse-click.
Icons		
Page reference	(📖 4)	Optically highlighted reference to another page. In this documentation activated by mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Engineering Tools	Software solutions for simple engineering at all stages
	»EASY Navigator« – Ensures easy operator guidance • All practical Lenze engineering tools at a glance • Tools can be selected quickly • Clearly arranged, simplifying the engineering process from the start
	»EASY Starter« – Simple tool for service technicians • Especially developed for the commissioning and maintenance of Lenze devices • Graphical user interface with few buttons • Simple online diagnostics, parameterisation and commissioning • No risk of accidentally changing the application • Ready applications can be loaded to the device
	»Engineer« – Multi-device engineering • For all products from our L-force portfolio • Practice-oriented user interface • Easy handling due to graphical user interfaces • Suitable for all project stages (configuration, commissioning, production) • Parameter setting and configuration
Code	Parameter used for controller parameterisation or monitoring. The term is usually called "index".
Subcode	If a code contains several parameters, these are stored in "subcodes". This Manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). The term is usually called "subindex".
Lenze setting	This setting is the default factory setting of the device.
FB Editor	Abbreviation for function block editor. Graphic interconnection tool which is available in the »Engineer« for function block interconnections on the **FB Editor**.
Function block	General designation of a function block for free interconnection in the FB Editor. Ein Funktionsbaustein (kurz: "FB") kann mit einer integrierten Schaltung verglichen werden, die eine bestimmte Steuerungslogik enthält und bei der Ausführung einen oder mehrere Werte liefert. Example: "L_Arithmetic_1" (FB for arithmetic operations) Many function blocks are available several times (e.g. L_And_1, L_And_2, and L_And_3).
System block	In the function block editor of the »Engineer«, system blocks provide interfaces to basic functions, "free codes", and to the hardware of the inverter (e.g. to the digital inputs). Each system block is available only once.
Port block	Block for implementing the process data transfer via a fieldbus
LP	Abbreviation for **L**enze **P**ort block Example: "LP_CanIn1" (CAN1 port block)
LS	Abbreviation for **L**enze **S**ystem block Example: "LS_DigitalInput" (system block for digital input signals)
MCI	Abbreviation for **M**otionbus **C**ommunication **I**nterface (fieldbus interface) The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system.
Technology application	A technology application is a drive solution based on the experience and know-how of Lenze in which function blocks interconnected to a signal flow form the basis for implementing typical drive tasks.
USB diagnostic adapter	The USB diagnostic adapter is used for the operation, parameterisation, and diagnostics of the controller. Data are exchanged between the PC (USB connection) and the controller (diagnostic interface on the front) via the diagnostic adapter. Order designation: E94AZCUS

1.4 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

> **Pictograph and signal word!**
>
> (characterise the type and severity of danger)
>
> **Note**
>
> (describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling

2 Features of the technology application

The "Position Follower" technology application represents a configuration for a slave drive and is used in applications in which one or several 8400 TopLine inverters have to be traversed synchronously to each other in terms of their position.

The axis bus (X10) is used as transmission medium. If required, a modification to "CAN on board" system bus (X1) or digital frequency coupling (X8) can be effected without high cost. ▶ Master value selection options for the slave position follower (📖 27)

The core function comprises a slave drive which follows a master drive with accuracy in terms of position. In addition to the "Electrical shaft" function, the application contains a function for positioning to the master position ("Lock to master") as well as all basic drive functions required (homing, manual jog, position follower, holding brake control).

The "Position Follower" technology application is primarily aimed at master/slave systems with limited traversing ranges like for example lifting platforms and Gantry axes, since these systems require absolute position synchronism.

Note!

Systems with modulo measuring systems such as printing units, cross cutters, draw rollers, chain conveyors, etc. are currently not supported by the technology application. Use the "Electrical Shaft Master" and "Electrical Shaft Slave" technology applications for applications of these types.

Tip!

Each master drive is provided with the "Position Sequencer" and "Electrical Shaft Master" technology applications.

2.1 Functional overview

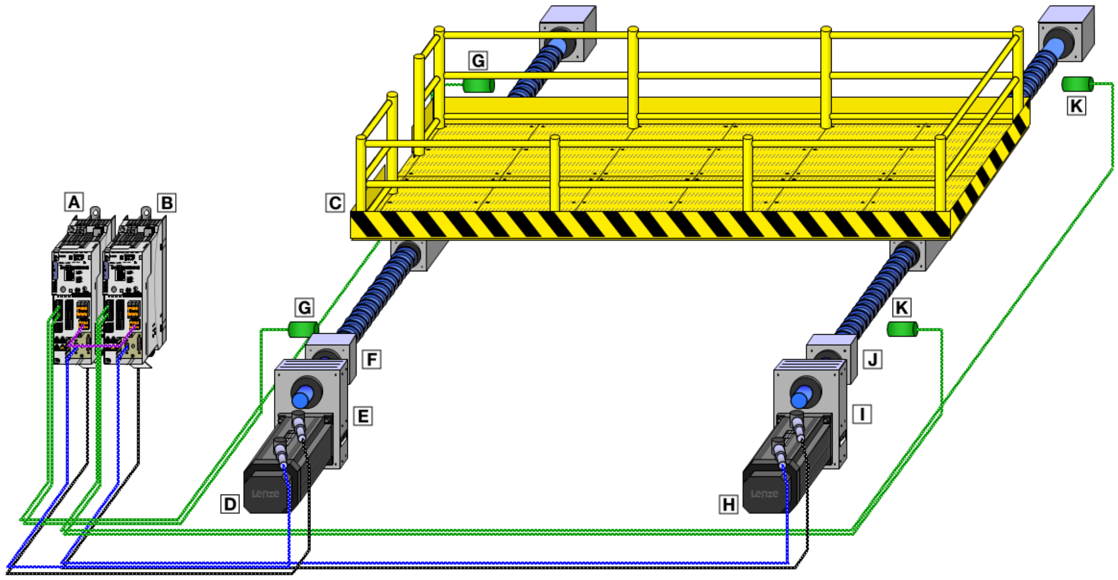
- Electrical shaft in line topology (all slave drives receive the same master position)
- Transfer of the master position via axis bus
- 1 master drive and up to 61 slave drives
- Adjustable synchronism factor
- "Lock to master" function: automatic straightening of the slave axis when position follower mode is selected
- Superimposition of a position offset with continuous coupling via ramp generator (Y offset)
- Use of the state machine of the Motion Control Kernel (MCK) and the following basic drive functions:
  - Homing
  - Manual jog
  - Position follower
  - Holding brake control
- Limit position monitoring (hardware limit switches and software limit positions)
- Following error monitoring system
- Control/status signals optionally via digital terminals and fieldbus interface (MCI)

2.2 Application ranges

- Gantry cranes
- Gantry axes
- Lifting platforms
- ...

Example: transport platform

Exemplary structure of a master/slave interconnection with a limited traversing range:



A	8400 TopLine inverter (master)		
B	8400 TopLine inverter (slave)		
C	Load (platform)		
D	(Servo) motor (master)	H	(Servo) motor (slave)
E	Gearbox (master)	I	Gearbox (slave)
F	Linear unit (spindle) (master)	J	Linear unit (spindle) (slave)
G	Hardware limit switches (master)	K	Hardware limit switches (slave)

2.3 System requirements

The technology application was created with the L-force »Engineer« V2.20 and can only be used with the versions V2.20 or higher.

Software

Product	Order designation	from version
L-force »Engineer« HighLevel	ESPEV-EHNNN	2.20

Hardware

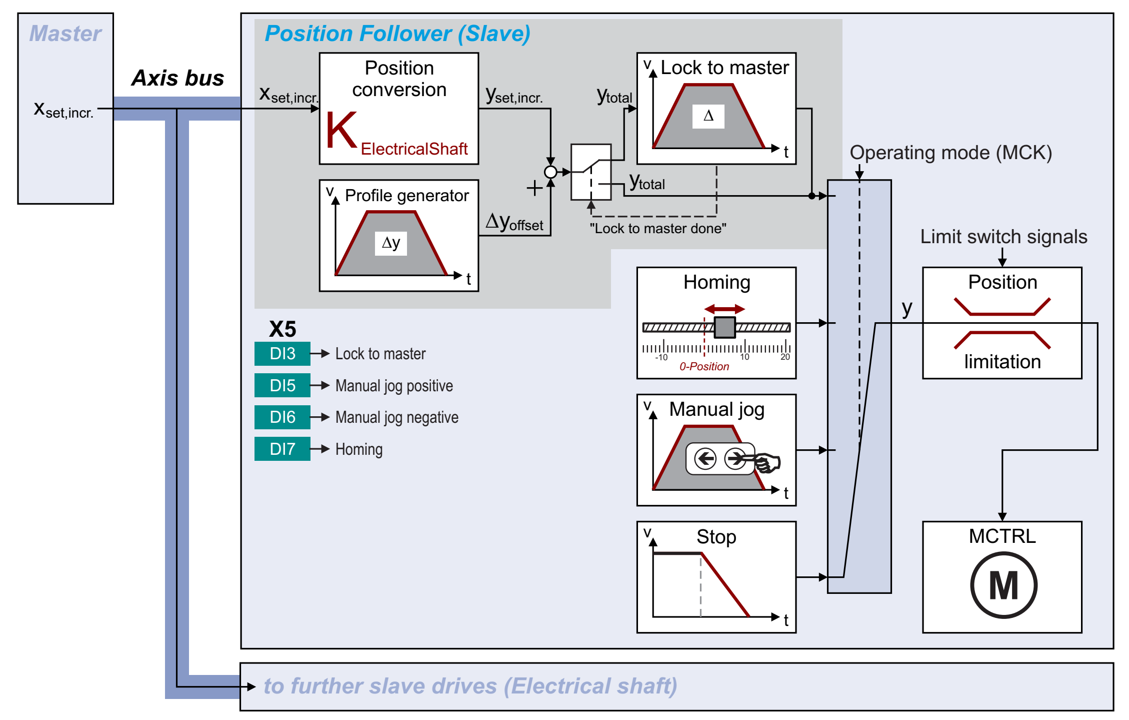
Product	Order designation	from hardware version	from software version
Inverter Drives 8400 TopLine C	E84AVTCxxxxx	VD	13.00

2.4 Basic signal flow

In the technology application, function and system blocks are interconnected so that typical basic functions of the position follower can be implemented from the application ranges mentioned before.

The following describes the principal signal flow with the essential functions.

Basic signal flow of the "Position Follower" technology application



2.5 Parameter setting in the FB Editor view

You can make the settings of the application-specific parameters directly in the FB Editor. This has the advantage that the signal flow can be traced. The interaction of the modules becomes clear. Moreover, you can reconfigure the I/O interconnection using the FB Editor and carry out an online monitoring of the application running in the device (e.g. for diagnostic purposes).

- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

- Colour codes and comments support you in handling the FB Editor.
  - The areas highlighted in turquoise represent the "user interface". If required, the pre-assignment of the I/O terminals can be adapted here and a control via the fieldbus interface (MCI) can be established.
  - In the areas highlighted in yellow, application-specific settings are required.

Detailed information on how to work with the FB Editor can be found in the reference manual/online help of the controller in the chapter "Working with the FB Editor".

2.6 Pre-assignment of the user interface

2.6.1 Pre-assignment of the I/O terminals

<table>
<tr><th>Terminal</th><th colspan="3">Function</th></tr>
<tr><td colspan="4">Digital input terminals</td></tr>
<tr><td rowspan="4">X5/RFR</td><td colspan="3">Controller enable</td></tr>
<tr><td colspan="2">RFR</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Inhibit drive</td></tr>
<tr><td colspan="2">HIGH</td><td>Enable drive</td></tr>
<tr><td>X5/DI1</td><td colspan="3">- (reserved for HTL encoder)</td></tr>
<tr><td>X5/DI2</td><td colspan="3">- (reserved for HTL encoder)</td></tr>
<tr><td rowspan="5">X5/DI3</td><td colspan="3">Switch on position follower operation ("Lock to master")
• Only possible when the controller is enabled and the home position is known.</td></tr>
<tr><td colspan="2">DI3</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>The drive unlocks from the electrical shaft and runs to a standstill ("Stop" state) along the stop ramp C01251/1.</td></tr>
<tr><td colspan="2">HIGH</td><td>The drive positions to the master position (electrical shaft) and then follows it directly.</td></tr>
<tr><td colspan="3">Note!
The electrical shaft interconnection must only be cancelled at standstill and must be re-established before the drive axes are restarted. Otherwise the system runs askew. Provide for corresponding locking in the higher-level control!</td></tr>
<tr><td>X5/DI4</td><td colspan="3">Connection of pre-switch sensor for reference search
• The edge sensitivity of this input and the response to the pre-switch signal depend on the homing mode selected.</td></tr>
<tr><td rowspan="6">X5/DI5
X5/DI6</td><td colspan="3">Manual jog</td></tr>
<tr><td>DI5</td><td>DI6</td><td>Function</td></tr>
<tr><td>LOW</td><td>LOW</td><td>-</td></tr>
<tr><td>HIGH</td><td>LOW</td><td>Manual jog in positive direction</td></tr>
<tr><td>LOW</td><td>HIGH</td><td>Manual jog in negative direction</td></tr>
<tr><td>HIGH</td><td>HIGH</td><td>- / Manual jog in the direction selected first</td></tr>
<tr><td rowspan="4">X5/DI7</td><td colspan="3">Start homing (only possible with controller enable)</td></tr>
<tr><td colspan="2">DI7</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Interrupt/stop homing</td></tr>
<tr><td colspan="2">LOW↗HIGH</td><td>Start homing</td></tr>
<tr><td colspan="4">Analog input terminals</td></tr>
<tr><td>X3/A1U</td><td colspan="3">- (not assigned, can be used freely)</td></tr>
<tr><td>X3/A2U</td><td colspan="3">- (not assigned, can be used freely)</td></tr>
</table>

Terminal	Function
Digital output terminals	
X4/DO1	HIGH ≡ "Drive is ready" state
X4/DO2	HIGH ≡ "Position follower operation active" state ("Lock to master done")
X4/DO3	HIGH ≡ "Home position is known" state
X107/BD1, BD2	Control of a holding brake by the basic function "holding brake control"
X101/COM, NO	Relay contact closed ≡ "An error is pending" state
Analog output terminals	
X3/O1U	Actual speed value • Scaling: 10 V ≡ 100 % reference speed (C00011)
X3/O2U	Actual torque • Scaling: 10 V ≡ 100 % maximum torque (C00057)

2.6.2 Process data input words

Fieldbus interface (MCI); **LP_MciIn** port block

Input words	Assignment
Words 1 ... 16	- (not preconfigured)

2.6.3 Process data output words

Fieldbus interface (MCI); **LP_MciOut** port block

Output words	Assignment
Word 1	Status word (for bit assignment see the following table)
Word 2	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)
Words 3 ... 4	Actual position value • Scaling: 65536 [increments] ≡ 1 encoder revolution (position encoder)
Word 5 ... 16	- (not preconfigured)

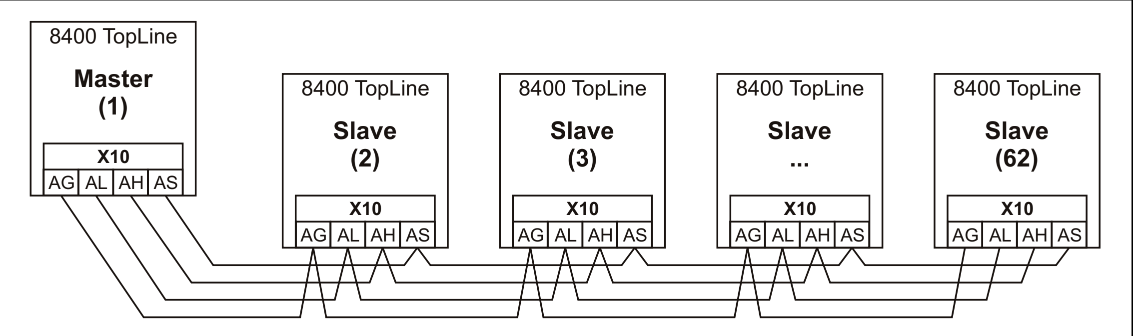
<table>
<tr><th>Status word</th><th colspan="6">Status</th></tr>
<tr><td>Bit 0</td><td colspan="6">1 ≡ Group error active (configurable in C00148)</td></tr>
<tr><td>Bit 1</td><td colspan="6">1 ≡ Inverter control inhibited (pulse inhibit is active)</td></tr>
<tr><td>Bit 2</td><td colspan="6">1 ≡ Drive controller is ready for operation</td></tr>
<tr><td>Bit 3</td><td colspan="6">1 ≡ Quick stop is active</td></tr>
<tr><td>Bit 4</td><td colspan="6">1 ≡ Setpoint torque is in the limitation</td></tr>
<tr><td>Bit 5</td><td colspan="6">- (not preconfigured)</td></tr>
<tr><td rowspan="2">Bit 6</td><td colspan="6">During open-loop operation:
1 ≡ Speed setpoint < Comparison value (C00024)</td></tr>
<tr><td colspan="6">During closed-loop operation:
1 ≡ Actual speed value < Comparison value (C00024)</td></tr>
<tr><td>Bit 7</td><td colspan="6">1 ≡ Controller inhibited (controller inhibit is active)</td></tr>
<tr><td rowspan="14">Bit 8 ... 11</td><td colspan="6">Bit coded display of the active device status</td></tr>
<tr><td>Bit 11</td><td>Bit 10</td><td>Bit 9</td><td>Bit 8</td><td>Device status</td><td>Meaning</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>FirmwareUpdate</td><td>Firmware update function is active</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Init</td><td>Initialisation active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Ident</td><td>Identification active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>ReadyToSwitchOn</td><td>Device is ready to start</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>SwitchedOn</td><td>Device is switched on</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>OperationEnabled</td><td>Operation</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>Trouble</td><td>Trouble active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>Fault</td><td>Fault active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>TroubleQSP</td><td>TroubleQSP is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>SafeTorqueOff</td><td>Safe torque off is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>SystemFault</td><td>System fault active</td></tr>
<tr><td>Bit 12</td><td colspan="6">1 ≡ a warning is indicated</td></tr>
<tr><td>Bit 13</td><td colspan="6">1 ≡ a fault is active. The inverter is in the "Trouble" device state.
• The motor has no torque (is coasting) due to the inhibit of the inverter.
• The "Trouble" device status is automatically abandoned if the error cause has been removed.</td></tr>
<tr><td>Bit 14</td><td colspan="6">1 ≡ position follower operation switched on</td></tr>
<tr><td>Bit 15</td><td colspan="6">1 ≡ Home position is known</td></tr>
</table>

3 Short setup of the technology application

3.1 Hardware structure required

The "Position Follower" technology application requires at least two 8400 TopLine inverters that are connected to each other via axis bus. A maximum of 62 nodes on the axis bus can be actuated:

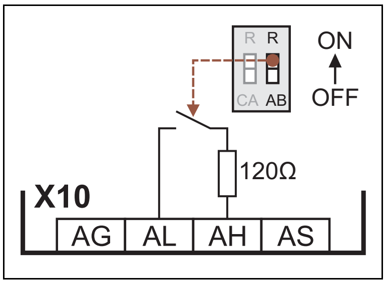
Axis bus topology



X10/AL, AH	Data transfer axis bus
X10/AS	IO axis bus
X10/AG	GND, reference potential of axis bus

The axis bus must be terminated between axis bus low (AL) and axis bus high (AH) at the first and last physical node each by a resistor (120 Ω).

The 8400 controller is provided with an integrated bus terminating resistor, which can be activated via the DIP switch labelled with "AB":



- OFF = bus terminating resistor is inactive
- ON = bus terminating resistor is active

3.2 Preconditions

For the execution of the short setup described in the following, the setting of the most important parameters (motor, feedback system, etc.) for each node on the axis bus is assumed.

The "commissioning wizard 8400" serves to carry out a guided commissioning of the controller based on the Lenze setting of the parameters.

How to proceed:

1. Before switching on: Make sure that the inverter is inhibited (digitale input terminal X5/RFR open).
2. Switch on voltage supply of the controller.
   For parameter setting and diagnostics of the controller without motor operation, an external 24-V supply through a safely separated power supply unit (SELV/PELV) is sufficient.
3. Establish a communication link between controller and Engineering PC, e.g. via USB diagnostic adapter (E94AZCUS):
   - connect the USB diagnostic adapter to the X6 diagnostic interface.
   - establish a connection between the USB diagnostic adapter and the PC via a free USB port.
4. Start »Engineer« on the Engineering PC, e.g. via the Windows® start menu:
   Start → All programs → Lenze → Engineering → L-force Engineer...
   After the program start, no project has been loaded first and the *start-up wizard* is displayed.

> You can find detailed information on the options of the start-up wizard and on the general use of the »Engineer« in the online help for the program which you can call with **[F1]**.

5. Create a new project or open a project already available.
6. Go to *Project View* and select the 8400 controller.
7. Click the icon to go online.
   After a connection to the controller has been established, the following status is displayed in the *Status line*:

8. Click the icon to start the *commissioning wizard 8400*.
   - Now the commissioning wizard guides you step by step through the setting of the important parameters for a quick commissioning.
   - The **Next** button can only be activated again after all parameter settings in the device have been reset via the **Load Lenze setting** button.
   - Execute the commissioning wizard right to the end.
   - You can skip the "Control mode" step by clicking **Next** (only relevant for "Speed actuating drive" technology application).

3.3 Short setup of the master drive

Each master drive is provided with the "Position Sequencer" and "Electrical Shaft Master" technology applications.

- With the "Position Sequencer" technology application, the drive can execute parameterisable travel profiles. The program flow is defined on the basis of a sequence table.
- The "Electrical Shaft" technology application serves to easily establish a so-called "electrical shaft" between several 8400 TopLine inverters, so that two or more devices can be used in angular synchronism.

Note!

The commissioning process for the master drive is described in the documentation for the corresponding technology application and is not part of this documentation.

The following subchapters only contain notes regarding the commissioning of the master drive.

3.3.1 Configuration of the speed/position output in the master drive

For the speed/position output to the slave drives, the **LS_AxisBusOut** SB is provided in the two "technology applications Position Sequencer" and "Electrical Shaft Master".

- In the default setting, the speed setpoint integrated to an angle (path) is output via line data words 1 & 2, and the speed setpoint is output via line data word 3.
- For optimum interaction with a slave position follower, set the value "1" for the master in C00470/4. Then the current position instead of the integrated speed is output via the axis bus.

Stop!

If operation as master position follower is switched on in the master (C00470/4 = "1"), homing of the master causes a step for the slave position follower if the current position is set to the home position!

Remedy: Inhibit the slave position follower when you are referencing the master!

If you are using an individual application in the master drive, provide for a suitable signal interconnection of the **LS_AxisBusOut** SB!

Configuration options of the speed/position output in the master drive

"Position Sequencer" technology application:

C00470/4 Operation as position follower master		C00470/2 Selection setpoint/actual value		-	Axis bus output
0	**Off**	**0**	**Setpoint**	-	Line data words 1 & 2 = speed setpoint integrated to an angle (path) Line data word 3 = speed setpoint
		1	Actual value	-	Line data words 1 & 2 = current speed integrated to an angle (path) Line data word 3 = current speed
1	On				Line data words 1 & 2 = current position Line data word 3 = current speed

"Electrical Shaft Master" technology application:

C00470/4 Operation as position follower master		C00470/6 Selection of master value/ actual speed		C00470/5 Selection of mere master value		Axis bus output
0	**Off**	**0**	**Master value**	**0**	**Mere master value**	Line data words 1 & 2 = speed setpoint integrated to an angle (path) Line data word 3 = speed setpoint
				1	Master value	Line data words 1 & 2 = speed setpoint integrated to an angle (path) from the MCK Line data word 3 = speed setpoint from the MCK
		1	Actual speed			Line data words 1 & 2 = current speed integrated to an angle (path) Line data word 3 = current speed
1	On					Line data words 1 & 2 = current position Line data word 3 = current speed

3.3.2 Setpoint retention in the master drive

In order to eliminate signal propagation delays of the electrical shaft via axis bus, a setpoint retention for the master drive can be set in C01239. The setpoint retention provides for a delay of the setpoints in the master drive.

If the setting in C01239 corresponds to the actual runtime of the signal from the master drive to the slave position follower, the setpoints in the master and slave are executed simultaneously, and no speed-dependent position offset will occur.

Parameter	Possible settings			Info
C01239	0	ms	32	Setpoint holding
	Lenze setting: 0 ms			

3.4 Short setup of the slave drive

The following steps are to be carried out for all slave position followers of the interconnection.

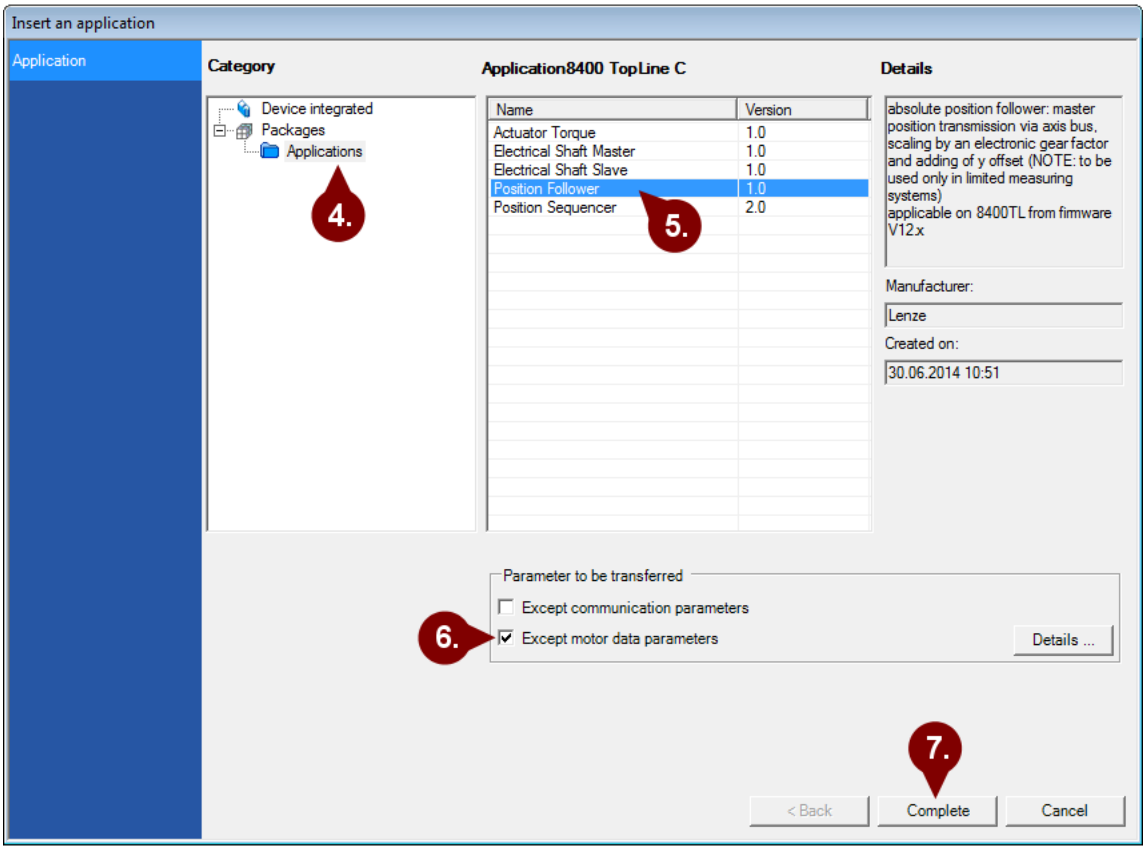
3.4.1 Step 1: Load "Position Follower" technology application

In the Lenze setting, the inverter uses the "Speed actuating drive" technology application integrated in the device. Execute the following steps to use the "Position Follower" technology application instead:

1. Select the controller in the *Project view*.
2. If there is still an online connection to the controller:

   Click the icon to go offline again.
   (the application can only be selected offline.)

3. Click the icon to select another application.
   The *Insert application* dialog box appears:



4. In the left field, select the "Packages" ➜ "Applications" category.
5. In the right field, select the "Position Follower" application.
6. Activate the **Except motor data parameters** option in order that the settings of the motor data parameters made before will not be overwritten.
7. Press **Complete** to close the dialog box again and load the selected application into the »Engineer« project.
8. Confirm the prompt on whether the current application is to be replaced by the "Position Follower" application with **Yes**.

3.4.2 Step 2: Axis bus settings

Make the following settings for every slave controller in the network:

- Axis bus address (C02430/1) = "2" ... "62"
  - Make sure that all controllers connected to the axis bus have different axis bus addresses.
- Sync signal source (C01120) = "2: AxisBusIO"
  - With this setting, the synchronisation cycle output by the master is used as synchronisation source. The parameters C01121 ... C01123 for the synchronisation of the internal time base are automatically set to fixed reasonable values to provide for a technically perfect operation of the axis bus!
  - Basically, only one source is allowed to synchronise the internal time base.
- Axis bus IO function (C02440/1) = "2: Slave"
  - This setting serves to define the slave.

Note!

For synchronisation of the master and slave, the axis bus IO line is used.
The **LS_AxisBusIO** SB must therefore not be utilised by the user in the application!

3.4.3 Step 3 (optional): Set up control system via the fieldbus interface (MCI)

In the default setting, the application is controlled via the digital input terminals. If functions are to be controlled via the fieldbus interface (MCI), the user interface (area highlighted in turquoise) has to be adapted accordingly in the function block editor.

- The assignment of the outputs on the left to the inputs on the right can be changed at will.
- The inputs on the right are permanently linked to functions of the application.

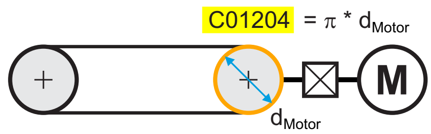
Tip!

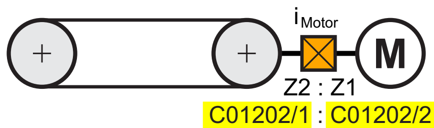
Control via the integrated CANopen interface ("CAN on board") can also be attained by means of simple interconnection changes. For this purpose, the **LP_CanIn** port block has to be inserted in the interconnection and connected to the corresponding inputs of the user interface.

3.4.4 Step 4: Set commissioning parameters

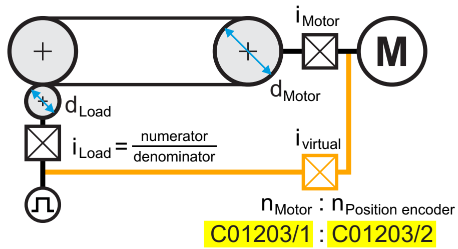
For a quick commissioning, only the following application-specific parameters have to be set or their default setting has to be checked!

- In order that you quickly find the respective parameterisation dialog in the FB Editor, the following table lists the block related to each parameter.
- The ▤ icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

Parameter (Block)	**Possible settings** (Lenze setting printed in bold)			**Info**
Machine parameters/axis settings				
C01204 (LS_MotionControl Kernel)	0.0001	units/rev.	214748.3647	Feed constant • The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft. • The value is entered in application units relating to one gearbox revolution. • Schematic diagram of a conveyor drive: 
	Lenze setting: 360.0000 units/U			
C01206/1 (LS_MotionControl Kernel)				Motor mounting direction
	0	**not inverted**		Motor is mounted directly
	1	inverted		Motor is mounted with rotation by 180°

C01202/1..2
(LS_MotionControl Kernel)
	1		65535	Gearbox ratio motor - load
				• Set the gearbox ratio with mathematical precision in the two subcodes:
				  • Subcode 1: numerator term (Z2)
				  • Subcode 2: denominator term (Z1)
				$i_{Motor} = \frac{C01202/1}{C01202/2} = \frac{Z2}{Z1}$
				• Schematic diagram of a conveyor drive:
				

	Lenze setting: 1:1			

Parameter (Block)	Possible settings (Lenze setting printed in bold)		Info
Position encoder			
If the motor encoder is used as position encoder, you do not need to parameterise a separate position encoder. Keep the Lenze setting for the following codes.			
C01206/2 (LS_MotionControl Kernel)			Position encoder mounting direction
	0	**not inverted**	Position encoder is mounted directly
	1	inverted	Position encoder mounted with rotation by 180
C01203/1..2 (LS_MotionControl Kernel)	1	65535	Speed ratio for motor - position encoder • Set the "virtual" speed ratio between the motor and the external position encoder with mathematical precision in the two subcodes: • Subcode 1: numerator term (motor speed) • Subcode 2: denominator term (encoder speed) • Schematic diagram of a conveyor drive:  • The "virtual" speed ratio can be calculated as follows: $i_{virtual} = \frac{C01203/1}{C01203/2} = \frac{i_{Motor}}{i_{load}} \cdot \frac{\pi \cdot d_{load}}{\pi \cdot d_{Motor}}$ $i_{virtual} = \frac{C01202/1}{C01202/2} \cdot \frac{denominator}{numerator} \cdot \frac{\pi \cdot d_{load}}{C01204}$
	Lenze setting: 1:1		
Electrical shaft function (measuring system conversion)			
Use the following parameters to specify the electrical shaft factor $K_{ElectricalShaft}$ for the conversion of the incremental master position to an incremental setpoint position for the slave. Basics with regard to the calculation can be found in the "Electrical shaft function (measuring system conversion)" chapter. (📖 38)			
C00471/3..4 (LS_ParFree)	0	65535	Electrical shaft factor $K_{ElectricalShaft}$ • Set the factor with mathematical precision in the two subcodes: • Subcode 3: numerator term • Subcode 4: denominator term (is only evaluated in terms of value) **Note:** Values greater than 32767 are interpreted as negative values according to the so-called two's complement. The effective value is calculated on the basis of the formula C0471/x - 2^{16}. Example: The setting "65535" is internally considered as the value 65535 - 2^{16} = -1.
	Lenze setting: 1:1		

Parameter (Block)	Possible settings (Lenze setting printed in bold)			Info
Y offset (shift of the slave position follower setpoint position with regard to the master position)				
C00475/1 (LS_ParFreeUnit_1)	-214748.3647	units	214748.3647	Y offset
	Lenze setting: 0.0000 units			
When the y offset is changed during position follower operation, the following parameters are to be set:				
C01060/2 (L_PosCtrlLin_1)	0.010	s	130.000	Acceleration ramp
	Lenze setting: 10 s			
C01060/3 (L_PosCtrlLin_1)	0.010	s	130.000	Deceleration ramp
	Lenze setting: 10 s			
C01061/1 (L_PosCtrlLin_1)	-15000	rpm	15000	Forward motion (positive speed) • The value is only evaluated with regard to its amount.
	Lenze setting: 50 min-1			
C01061/2 (L_PosCtrlLin_1)	-15000	rpm	15000	Return motion (negative speed) • The value is only evaluated with regard to its amount.
	Lenze setting: 50 min-1			
Following error monitoring system				
C01215/1 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 1 • The setting "0" deactivates following error monitoring 1
	Lenze setting: 500 units			
C01215/2 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 2 • The setting "0" deactivates following error monitoring 2
	Lenze setting: 1000 units			
C00595/5...6 (LS_SetError_1)				Response at the activation of following error monitoring • Subcode 5: response at the activation of following error monitoring 1. • Subcode 6: response at the activation of following error monitoring 2.
	0	No Reaction		
	1	Fault		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	**Warning**		
	6	Information		
Limit position monitoring				
C01229/1...2 (LS_MotionControl Kernel)	-214748.3647	units	214748.3647	Positive and negative software limit position for limiting the valid traversing range • Subcode 1: Positive software limit position • Subcode 2: Negative software limit position **Note:** The software limit positions are only evaluated if • the home position is known to the drive, **and** • the software limit positions for the respective operating mode have been activated, **and** • the positive software limit position is set to a greater value than the negative software limit position!
	Lenze setting: ±100000.0000 units			

3.4.5 Step 5 (optional): Connection of limit switch signals

Note!

In der "Position Follower" technology application, no evaluation of hardware limit switches is prepared by default yet. However, if required, an evaluation can be retrofitted by means of only a few steps.

▸ Limit position monitoring (📖 48)

3.4.6 Step 6: Go online and transfer parameter set to the inverter

In order to set the current parameter settings in the controller to the settings in the project, transmit the parameter set to the controller.

1. Click the icon to go online.
2. Click the icon to transmit the parameter set to the controller.
3. After a successful transmission, click the icon to save the parameter set safe against mains failure in the integrated memory module.

3.5 Approaching the drive system (terminal control)

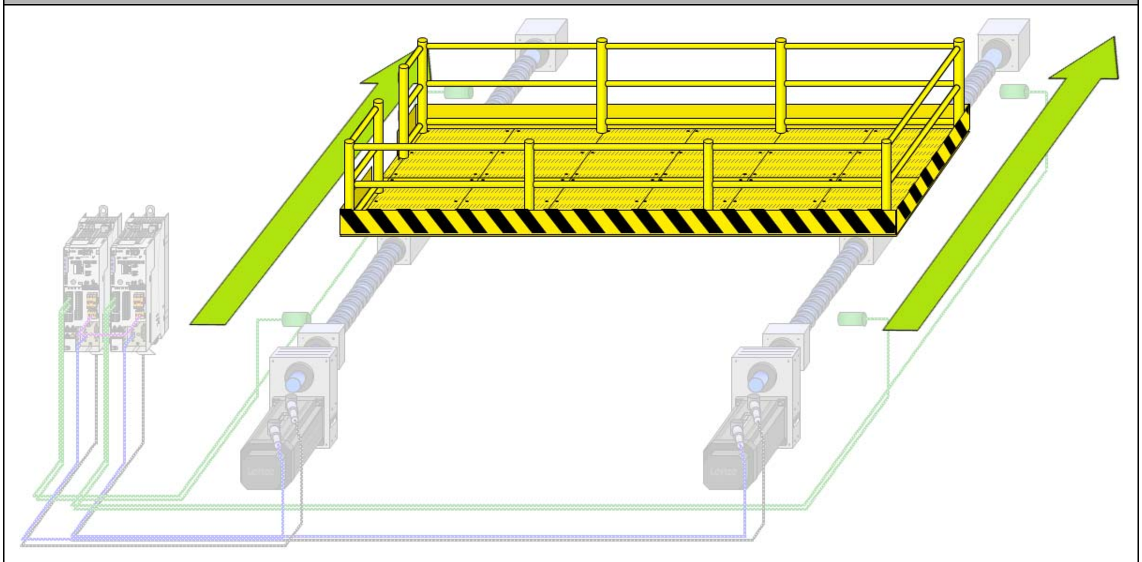
After the parameter set has been transmitted to the inverter, the inverter can now be enabled and the control signals can be selected via the corresponding interfaces.

▸ Pre-assignment of the user interface (📖 11)

How to proceed:

1. Enable position follower slave (set X5/RFR to HIGH level).
2. If the home position in the position follower slave is not known:
   reference position follower slave (set X5/DI7 to HIGH level).
3. After having completed the homing process (output X4/DO3 is set to HIGH level):
   reset X5/DI7 to LOW level.
4. "Lock to master": position the position follower slave to the position of the master drive to correct an inclination (set X5/DI3 to HIGH level).
   Note: During the correction movement of the position follower slave, the master drive must be at standstill.
5. After having completed the "Lock to master" function (output X4/DO2 is set to HIGH level):
   reset X5/DI3 to LOW level.
6. Enable master drive and traverse as desired.
   The position follower slave traverses accordingly.

Example: Synchronous movement of the master drive and position follower slave



3.6 (Optionally): Setting the optimisation parameters in the slave drive

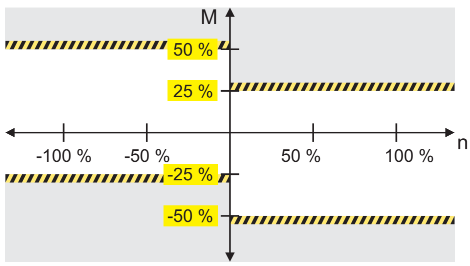
The following application-specific parameters are used for optimisation and can also be adapted during operation.

Stop!

If you change parameters in the »Engineer« during an online connection to the device, the changes are directly transferred to the device!

Tip!

Do not forget to save the parameter changes carried out with mains failure protection in the memory module implemented! (C00002/11 = "1: on/start")

Parameter (Block)	**Possible settings**			**Info**
Position controller				
C00472/1 (LS_ParFree_a)	-199.99	%	199.99	Limitation of the position controller output
	Lenze setting: 100 % reference speed (C00011)			
C00472/2 (LS_ParFree_a)	-199.99	%	199.99	Adaptation of the position controller gain
	Lenze setting: 100 % Vp (C00254)			
Torque limitation in motor mode/in generator mode				
The torque limitation set is always active.				Example: Definition of the torque limitations C00472/3 = 25 % C00472/4 = 50 % 
C00472/3 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in motor mode
	Lenze setting: 100 % maximum torque (C00057)			
C00472/4 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in generator mode
	Lenze setting: 100 % maximum torque (C00057)			

Parameter (Block)	Possible settings			Info
Profile parameters for positioning to master position ("Lock to master")				
C01236/1 (LS_MotionControl Kernel)	-214748.3647	units/s	214748.3647	PosFollower: Sync. speed Synchronisation speed for approaching the setpoint position of the master.
	Lenze setting: 360.0000 units/s			
C01237/1 (LS_MotionControl Kernel)	-214748.3647	units/s^2	214748.3647	Pos follower: Sync. accel. Acceleration for ramp-up to synchronisation speed
	Lenze setting: 3600.0000 units/s^2			
C01237/2 (LS_MotionControl Kernel)	-214748.3647	units/s^2	214748.3647	Pos follower: Sync. decel. Deceleration for synchronisation speed ramp-down to standstill (to the setpoint position of the master).
	Lenze setting: 3600.0000 units/s^2			
C01238/1 (LS_MotionControl Kernel)	0.000	s	10.000	Pos follower: Sync. S-ramp time S-ramp time for the synchronisation speed ramps.
	Lenze setting: 0.000 s			
Following error monitoring system				
C01244/2...3 (LS_MotionControl Kernel)	0	ms	600000	Waiting time for following error monitoring 1 & 2 • In order to avoid that an error is triggered by acceleration and a narrow tolerance limit can be nevertheless monitored at standstill in the target, the response of the following error monitoring system can be delayed by setting a waiting time.
	Lenze setting: 0 ms			
Limit position monitoring				
C01218 - Bit 3 (LS_MotionControl Kernel)				Consideration of the software limit positions in MCK "Position follower" operating mode
	0	Software limit positions not active		
	1	**Software limit positions active** (if the home position is known)		
C01230 - Bit 3 (LS_MotionControl Kernel)				Consideration of the software limit positions in MCK "Manual jog" operating mode
	0	Software limit positions not active		
	1	**Software limit positions active** (if the home position is known)		
C00595/1...4 (LS_MotionControl Kernel)				Response at the activation of limit position monitoring • Subcode 1: response at the approach of the positive limit switch. • Subcode 2: response at the approach of the negative limit switch. • Subcode 3: response at overtravelling the positive software limit position (C01229/1). • Subcode 4: response at overtravelling the negative software limit position (C01229/2).
	0	No Reaction		
	1	Fault		
	3	**TroubleQuickStop**		
	4	WarningLocked		
	5	Warning		
	6	Information		

4 Detailed functions of the technology application

This chapter describes the functions implemented in the "Position Follower" technology application with the possible settings relevant for the application.

Detailed information on the function and parameterisation of the functions described in the following can be found in the reference manual/online help of the controller.

4.1 Master value selection options for the slave position follower

The master value for position follower operation can be obtained from different sources:

- Axis bus (X10)
- Digital frequency coupling (X8)
- "CAN on board" system bus (X1)

In the following subchapters, these sources are specified in detail.

4.1.1 Axis bus (X10)

In the default setting, the "Position Follower" technology application uses the master value selection via the axis bus. In this configuration only an 8400 TopLine can be used as master drive.

▸ Hardware structure required (📖 14)

▸ Step 2: Axis bus settings (📖 19)

Note!

Activate the time-out monitoring functions of the axis bus in both the master and the slave drive to prevent the master and slave from moving apart in the case of incorrect master value transmission! ▸ Bus monitoring (📖 52)

For synchronisation of the master and slave, the axis bus IO line is used.
The **LS_AxisBusIO** SB must therefore not be utilised by the user in the application!

4.1.2 Digital frequency coupling (X8)

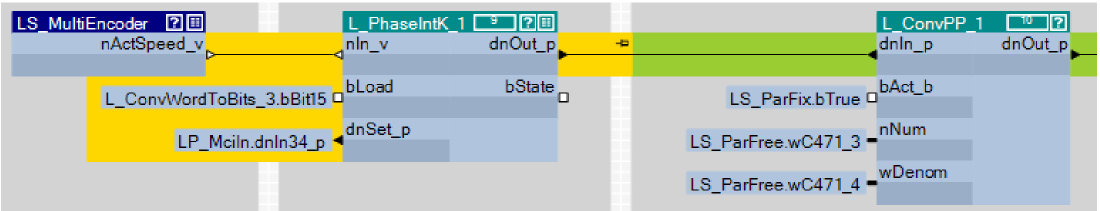
The "Position Follower" technology application can be refitted to a master value selection via digital frequency at low cost and effort. Therefore an 8400 TopLine is not absolutely required as master drive for the master value selection.

Note!

For a master value selection via digital frequency, the multi-encoder interface (X8) must not be occupied by a feedback system!

Since the digital frequency signal solely reflects the relative movement of the master (speed signal $V_{ElShaft}$), an integrator (**L_PhaseIntK_1**) is to be added to the main signal flow, integrating the master speed (in the example: *nActSpeed_v*) to a master position:

Example: Retrofitting of the function block interconnection for digital frequency coupling



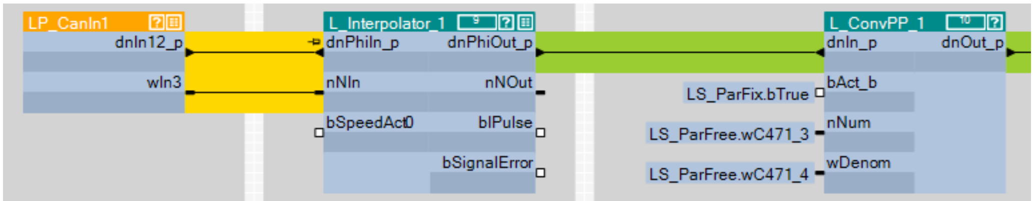
- The **L_PhaseIntK_1** integrator must be called in the processing order before the **L_ConvPP_1** FB.
- The integrator must be pre-loaded once with a suitable starting position before position follower operation is started.
  In the example,
  - the incremental starting position is preselected via the MCI process data input words 3 + 4.
  - the starting position is accepted via bit 15 of the application control word (**L_ConvBitsToWord_3** FB). The user must then interconnect this bit accordingly.

4.1.3 "CAN on board" system bus (X1)

The "Position Follower" technology application can be refitted to a master value selection via the "CAN on board" system bus at low cost and effort. Therefore an 8400 TopLine is not absolutely required as master drive for the master value selection.

Compared with the standard signal flow, the signal flow hardly changes at all. Instead of via the **LS_AxisBusIn** SB, the master value selection is effected via one of the CAN input ports (in the example **LP_CanIn1**):

Example: Retrofitting of the function block interconnection for master value selection via CAN



- The telegram structure must be identical to that for the axis bus: The master position is transferred via CAN process data input words 1 + 2, and the master speed via CAN process data input word 3.
- If transmission cycles > 1 ms are used, the interpolation function (**L_Interpolator_1** FB) must be activated and parameterised.

Note!

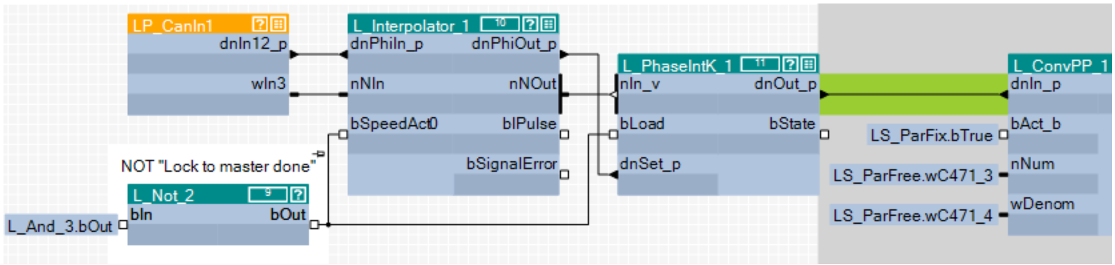
Compared with the real setpoint, the interpolation produces a dead time which complies with the duration of the interpolation cycles (plus possible transmission times on the data bus). In order to eliminate this dead time, a setpoint retention for the master drive can be set in C01239. ▶ Setpoint retention in the master drive (📖 17)

Tip!

In the case of lower to medium demands to the dynamics, even a non-synchronised master value selection can be applied.

- For this purpose, the follow-up control contained in the **L_Interpolator_1** FB can be used, which outputs a smoothed speed setpoint at output *nNOut*.
- A downstream integrator (**L_PhaseIntK_1**) eventually converts the *nNOut* speed setpoint to a position setpoint.
- By means of the additional logic (**L_Not_2**), the set position with inactive position follower operation (**L_And_3**.*bOut* = FALSE) is directly loaded to the integrator (**L_PhaseIntK_1**).

Example: Use of the internal follow-up control in the L_Interpolator_1 with the L_PhaseIntK_1 auxiliary integrator



4.2 Basic drive functions (MCK)

Danger!

During homing, manual jog, and positioning, specially assigned profile parameters are active. If they have not been set correctly, the drive may carry out an unexpected movement!

Detailed information relating to the basic drive functions and the corresponding parameters can be found in the reference manual/online help of the inverter in the "Basic drive functions (MCK)" chapter.

4.2.1 Operating mode selection

The operating mode selection by the user is carried out in the default setting via the digital inputs.
▸ Pre-assignment of the I/O terminals (📖 11)

For internal mapping of the individual basic drive functions, the "Position Follower" technology application uses the state machine of the **Motion Control Kernel**. The desired operating mode is selected via the *wOperationMode* input of the **L_MckCtrlInterface_1** FB.

For a control bit-dependent selection, a corresponding interconnection logic is configured upstream to the *wOperationMode* input in the default setting (consisting of the **L_DigitalLogic5_1**, **L_And_1**, and **L_FixSet_w_1** FBs).

Note!

If more than one operating mode is selected at the same time, a change-over to the "Stop (STOP)" operating mode is carried out. The drive is braked to a standstill from its movement.

Tip!

- C00137 displays the current device state.
- C01243 shows the active operating mode of the **Motion Control Kernel**.

4.2.2 Stop (STOP)

Similar to the 9400 device series, the so-called "Stop" mode is also provided for 8400 HighLine/TopLine inverters, braking the drive to a standstill from its movement when a basic drive function is deselected.

The "Stop" mode differs from quick stop as follows:

Stop (STOP)	Quick stop (QSP)
Is triggered via **Motion Control Kernel**.	Is triggered directly in the motor control. The **Motion Control Kernel** changes to the "Standby" state immediately.
Deceleration ramp adjustable in C01251/1 in application units [units/s²].	Deceleration ramp adjustable in C00105 in [s] with reference to the reference speed (C00011).
S-ramp time adjustable in C01252/1 in [s].	Only linear ramp.
Braking process is executed in a position-controlled manner.	Braking process is executed in a speed- or position-controlled manner (can be selected via bit 1 in C00104/1).
Standstill is executed in position-controlled manner.	Standstill is executed in speed- or position-control (can be selected via bit 0 in C00104/1).

"Stop" is defined as an individual operating mode in the Motion Control Kernel. When another operating mode is deselected, the drive is braked to a standstill from the current movement with the profile parameters of "Stop". The following setting parameters are provided:

Parameter (Block)	Possible settings			Info
C01251/1 (LS_MotionControl Kernel)	-214748.3647	units/s^2	214748.3647	MCK: Stop: Decel. Deceleration for setpoint speed ramp-down to standstill.
	Lenze setting: 3600.0000 units/s^2			
C01251/2 (LS_MotionControl Kernel)	0.000	s	10.000	MCK: Stop: S-ramp time S-ramp time for setpoint speed ramp-down to standstill.
	Lenze setting: 0.000 s			

STOP **Stop!**

In the case of the 8400 device series, both "Stop" and "Quick stop" (QSP) are executed with the torque that is specified at the **LS_MotorInterface** SB via the *nTorqueMotLimit_a* or *nTorqueGenLimit_a* input (difference to the 9300, ECS, and 9400 device series).

4.2.3 Homing

In order to be able to actuate the drive in a reasonable fashion, the position of the zero position within the physically possible travel range must be known. The zero position, also called the "reference", can be defined by homing or reference setting. This so-called "homing" is usually carried out after the mains is switched on. Furthermore homing is required after an encoder error has been eliminated.

- "Manually" starting/stopping the homing process is executed via bit 6 of the application control word (**L_ConvBitsToWord_3** FB). In the default setting, bit 6 is linked with digital input DI7.
- Homing is to be configured according to the requirement of the application, as described in the reference manual/online help of the inverter.
- If the home position is known to the drive, a response is sent via digital output DO3 and bit 15 of the MCI process data output word 1.

Note!

Positioning operation and position follower operation can only be executed if the home position is known to the drive (**LS_MotionControlKernel**.*bHomePosAvailable* = TRUE). The monitoring of software limit positions can also only be carried out with a known reference!

Note!

The "Position Follower" technology application is typically used on following axes that are coupled to a master axis and further following axes via the mechanics and which can therefore be traversed against each other in any way. Thus the homing processes cannot be executed independently on each axis.

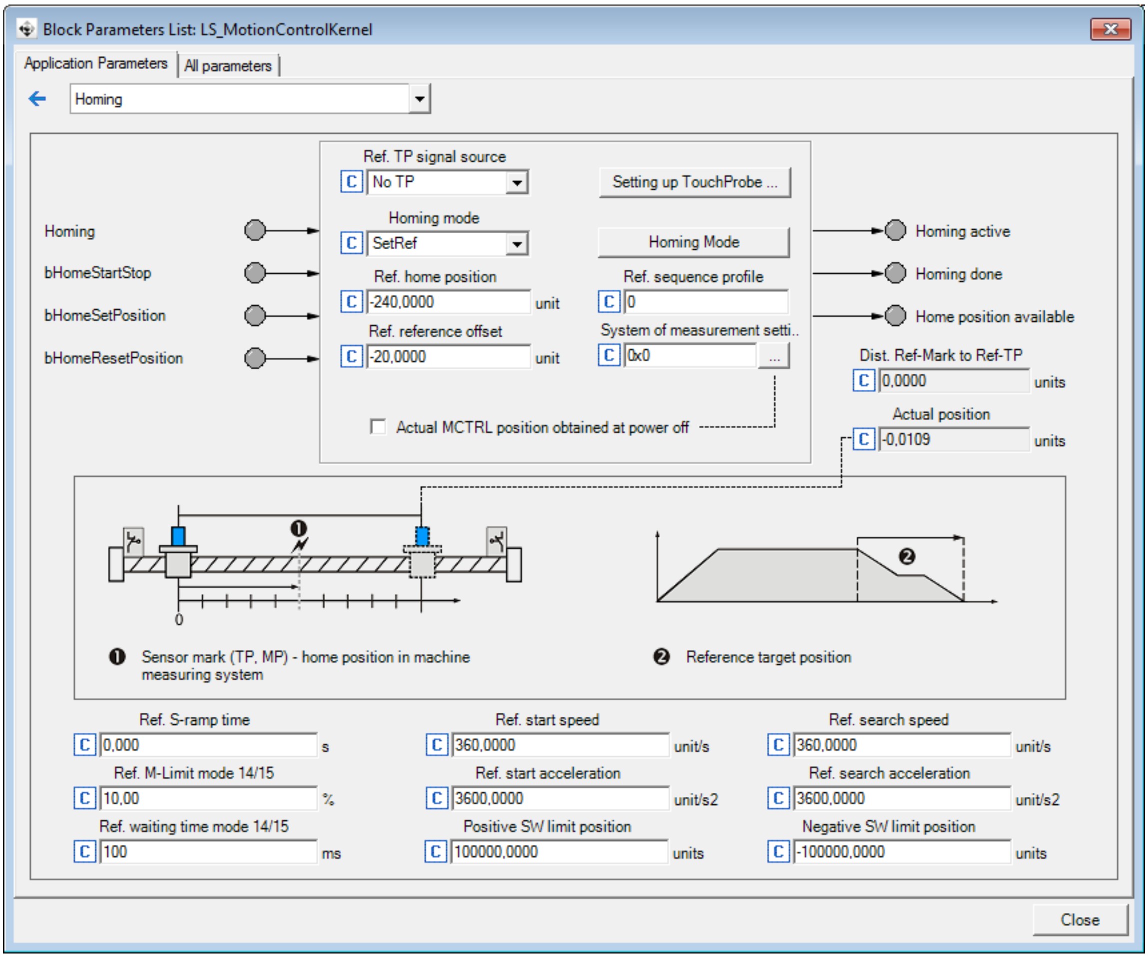
The following procedure has proved of value:

- Mount the reference-determining sensors (e.g. touch probe sensor) to the same position for all axes.
- Set the feed constant (C01204) and motor mounting position (C01206/1) on all axes of the interconnection so that an identical length scaling and counting direction results on all axes.
- Be sure to parameterise all homing profile and position parameters in all axes in an identical manner (see the following parameter list).
- Start the homing process simultaneously on all axes involved and only reset the homing command when all axes in the interconnection have completed the homing process successfully!

Like this, the homing process produces identical traversing profiles for all axes and the mechanics are not pulled askew.

How to go to the parameterisation dialog of the basic "Homing" function:

1. Open the parameter list for the **LS_MotionControlKernel** SB.
2. In the *Block Parameters List...* dialog box on the **Application parameters** tab, select the "Homing" entry in the upper list field.



Note!

For a reference search with touch probe detection:

If the reference signal is to follow a real touch probe, the touch probe interface must be configured accordingly via the **Setting up TouchProbe...** button!

Parameter	Info	Lenze setting	
		Value	Unit
C01221	MCK: Homing mode	100: SetRef	
C01224/1	MCK: Ref. initial speed	360.0000	unit/s
C01225/1	MCK: Ref. initial acceleration	3600.0000	unit/s2
C01224/2	MCK: Ref. search speed	360.0000	unit/s
C01225/2	MCK: Ref. search acceleration	3600.0000	unit/s2
C01226/1	MCK: Ref. S-ramp time	0.000	s

Parameter	Info	Lenze setting	
		Value	Unit
C01222	MCK: Ref. M limit mode 14/15	10.00	%
C01223	MCK: Ref. waiting time mode 14/15	100	ms
C01227/1	MCK: Ref. offset reference degree	-20.0000	unit
C01227/2	MCK: Ref. home position	-240.0000	unit
C01228	MCK: Ref. sequence profile	0	
C01229/1	MCK: Positive SW limit position	1000000	units
C01229/2	MCK: Negative SW limit position	1000000	units
C01246/1	MCK: Ref. TP signal source	0: No TP	

4.2.4 Manual jog

In this operating mode, the drive can be traversed manually in a clockwise or anticlockwise direction ("jogging mode").

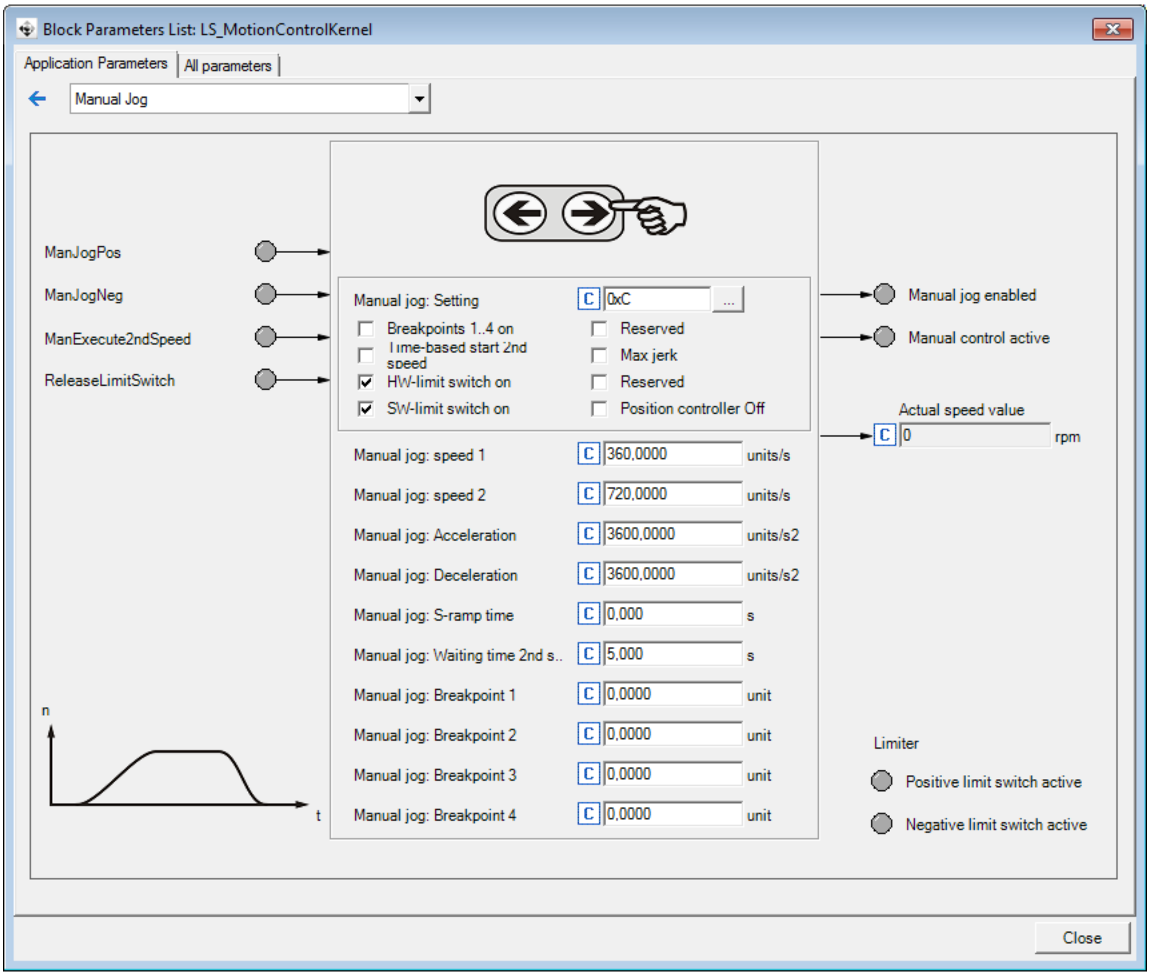
- The basic function "Manual jog" is activated via bit 12 and bit 13 of the application control word (**L_ConvBitsToWord_3** FB). In the default setting, bit 12 and bit 13 are linked to digital inputs DI5 and DI6.
- An activation via the fieldbus interface (MCI) is possible after a corresponding adaptation of the user interface. See commissioning; Step 3 (optional): Set up control system via the fieldbus interface (MCI). (📖 19)

Note!

In the Lenze setting, the software limit positions parameterised are active for manual jog if the home position is known. ▸ Limit position monitoring (📖 48)

How to go to the parameterisation dialog of the basic "Manual jog" function:

1. Open the parameter list for the **LS_MotionControlKernel** SB.
2. In the *Block Parameters List...* dialog box on the **Application parameters** tab, select the "Manual Jog" entry in the upper list field.

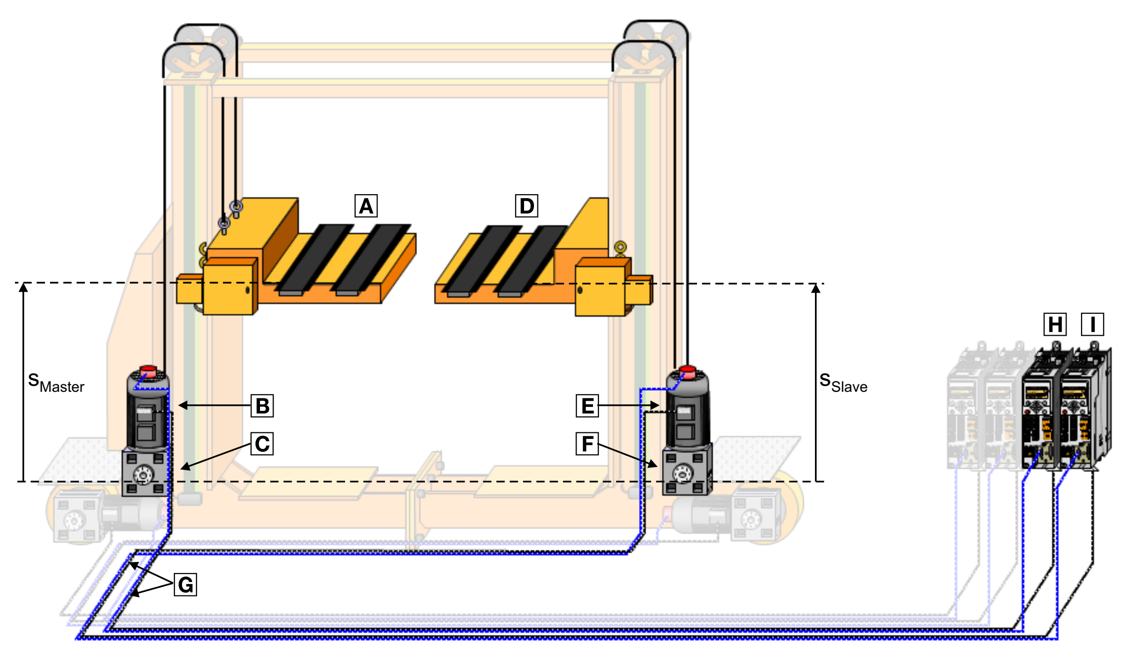


Parameter	Info	Lenze setting Value	Unit
C01230	MCK: Manual jog setting	Bit coded	
C01231/1	Manual jog: speed 1	360.0000	units/s
C01231/2	Manual jog: Speed 2	720.0000	units/s
C01232/1	Manual jog: Acceleration	3600.0000	units/s2
C01232/2	Manual jog: Deceleration	3600.0000	units/s2
C01233/1	Manual jog: S-ramp time	0.000	s
C01235/1	Waiting time 2nd speed	5.000	s
C01234/1	Manual jog: Breakpoint 1	0.0000	unit
C01234/2	Manual jog: Breakpoint 2	0.0000	unit
C01234/3	Manual jog: Breakpoint 3	0.0000	unit
C01234/4	Manual jog: Breakpoint 4	0.0000	unit

4.2.5 Holding brake control

For hoisting axes with floating loads, drives are often equipped with holding brakes making it possible to inhibit the drives without the load sagging.

Example: Use of holding brakes with a storage and retrieval unit



A	Lifting device (master)	D	Lifting device (slave)
B	Drive motor (master) with holding brake and feedback system	E	Drive motor (slave) with holding brake and feedback system
C	Gearbox (master)	F	Gearbox (slave)
G	Control of the holding brake via two additional cores in the motor cable		
H	Inverter (master)	I	Inverter (slave)

The 8400 TopLine inverter is already equipped with an automatic holding brake control.

Danger!

Please note that the holding brake is an important element of the safety concept of the entire machine.

Thus, proceed very carefully when commissioning this system part!

Detailed information on how to parameterise the holding brake control can be found in the reference manual/online help of the controller in the chapter "Basic drive functions (MCK)".

The documentation of the holding brake control contains safety instructions which must be observed!

Application-specific notes on the holding brake control:

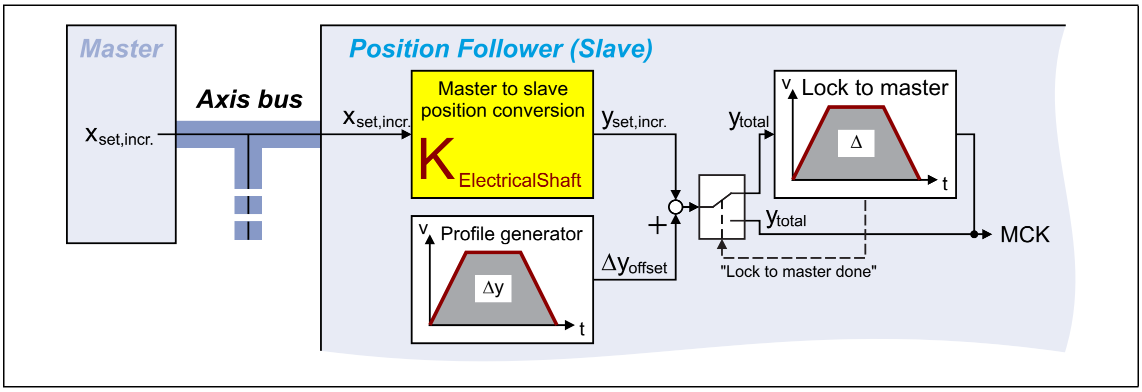
- In the Lenze setting, the mode 0 (brake control off) is preset in C02580.
- The technology application is prepared for the control of a 24 V holding brake via the high current output (terminal strip X107).
- The application of the holding brake effects a controller inhibit, and with an inhibited inverter, the following error is reset. Starting from version 14.00.00, a following error value can be set in C01215/3, which also remains stored when the controller is inhibited.
- In mode 12 (controlled automatically), the forced release is executed on the holding brake when position follower operation is active ("Lock to master done"), so that the slave drive follows each of the master drive's movements without the holding brake being applied whilst the master value is fixed. For the implementation of this function, the "Lock to master done" signal in the application (**L_And_3**.*bOut*) is linked with the *bMBrakeRelease* input of the **LS_MotionControlKernel** SB.

4.3 Position follower functions

4.3.1 Electrical shaft function (measuring system conversion)

In order to enable the slave position follower to follow the master in a position-synchronous fashion, the position of the master must be provided in the slave. For this purpose, various transmission paths from the master to the slave drive are provided, as described in the "Master value selection options for the slave position follower" chapter.

Then a measuring system conversion is carried out in the slave position follower: The incremental master position $x_{set,incr.}$ is converted to an incremental setpoint position $y_{set,incr.}$ for the slave.



[4-1] Measuring system conversion (excerpt from the basic signal flow)

Basics for the calculation	
Formula for the conversion of the master position to a set position for the slave position follower	
$$y_{set,incr.} = K_{ElectricalShaft} \cdot x_{set,incr.} = \frac{i_{Slave}}{i_{Master}} \cdot \frac{FC_{Master}}{FC_{Slave}} \cdot K_{User} \cdot x_{set,incr.}$$	
Formula for the calculation of the electrical shaft factor $K_{ElectricalShaft}$	
Solving for the electrical shaft factor $K_{ElectricalShaft}$ from the first formula yields the following: $$K_{ElectricalShaft} = \frac{y_{set,incr.}}{x_{set,incr.}} = \frac{i_{Slave}}{i_{Master}} \cdot \frac{FC_{Master}}{FC_{Slave}} \cdot K_{User} = \frac{C00471/3}{C00471/4}$$	
$K_{ElectricalShaft}$	Electrical shaft factor
$y_{set,incr.}$	Incremental set position (y position)
$x_{set,incr.}$	Incremental master position (x position)
i_{Slave}	Gearbox ratio of slave position follower
i_{Master}	Gearbox ratio of master drive
FC_{Master}	Feed constant of master drive
FC_{Slave}	Feed constant of slave position follower
K_{User}	User proportionality factor (in the case of position synchronism, set to "1"!)

The electrical shaft factor $K_{ElectricalShaft}$ is preselected via the following parameters in the "Position Follower" technology application:

Parameter (Block)	Possible settings (Lenze setting printed in bold)			Info
C00471/3..4 (LS_ParFree)	0		65535	Electrical shaft factor $K_{ElectricalShaft}$ • Set the factor with mathematical precision in the two subcodes: • Subcode 3: numerator term • Subcode 4: denominator term (is only evaluated in terms of value) **Note:** Values greater than 32767 are interpreted as negative values according to the so-called two's complement. The effective value is calculated on the basis of the formula C0471/x - 2^{16}. Example: The setting "65535" is internally considered as the value 65535 - 2^{16} = -1.
	Lenze setting: 1:1			

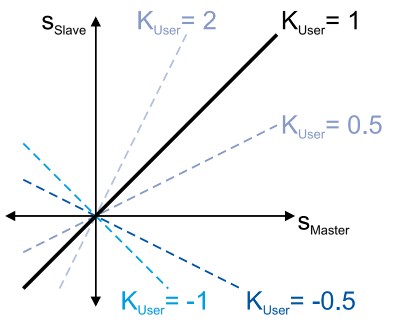
For the conversion of the position, the **L_ConvPP_1** FB is used in the application.

Tip!

Most applications require an exact position synchronism. The user proportionality factor K_{User} in these cases must always be set to "1".

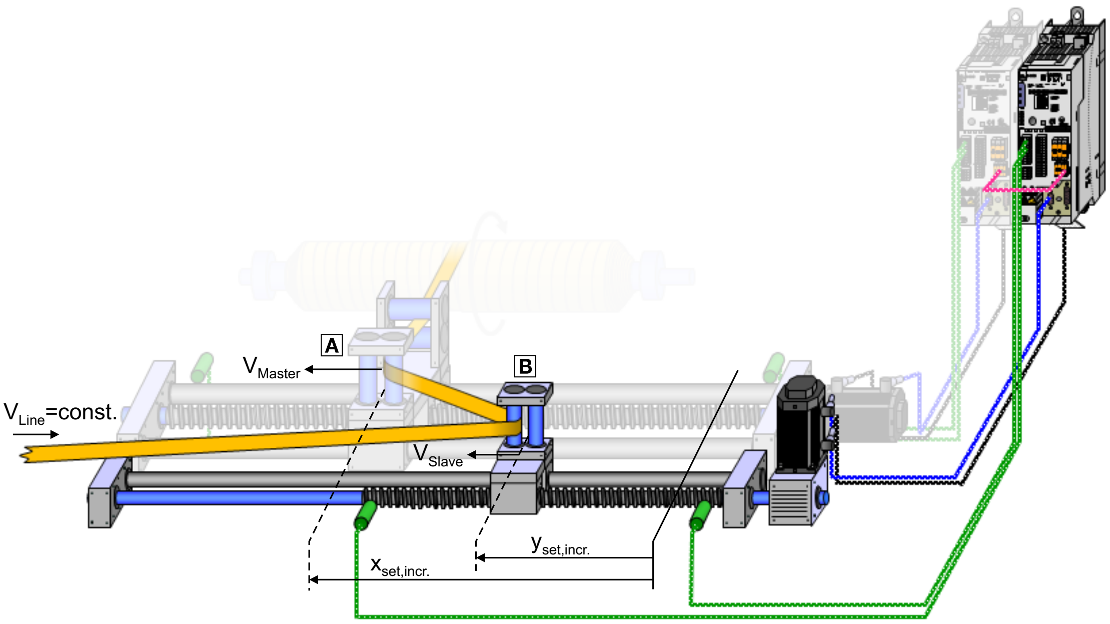
- With $K_{User} \neq 1$, absolute stretching/compression of the set position, similar to that in the electronic cam technology, can be obtained.
- K_{User} < 0 produces a movement in the opposite direction with regard to the master.

Example: Impact of the user proportionality factor K_{User}



K_{User}	Impact
1	Position synchronism
>1	Slave travels faster than master
<1	Slave travels slower than master
-1	Slave travels in the opposite direction with regard to the master
0> K_{User} >1	Slave travels slower than master and in the opposite direction with regard to master

Example: Speed compensation for traversing control
In order to keep the machine material speed V_{Line} exactly constant, a compensation process is to be executed via an additional control system (slave), which is to move in the ratio $K_{User} = s_{Slave} : s_{Master} = 1 : 2$ to the traversing support (master).



A	Traversing support (master):		B	Additional control system (slave):	
	i_{Master}	= 1 : 1		i_{Slave}	= 39 : 17
	FC_{Master}	= 10.0000 [mm/rev.]		FC_{Slave}	= 7.5000 [mm/rev.]
				K_{User}	= 1 : 2

Calculation of the electrical shaft factor $K_{ElectricalShaft}$

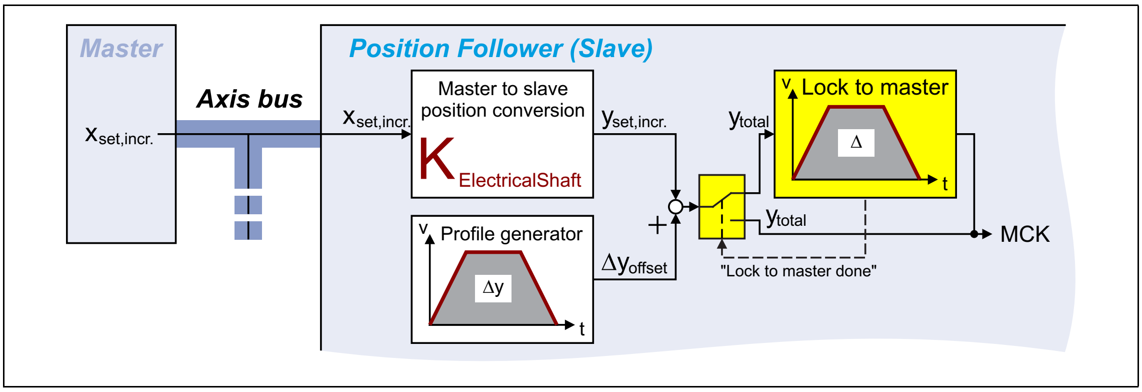
$$K_{ElectricalShaft} = \frac{i_{Slave}}{i_{Master}} \cdot \frac{FC_{Master}}{FC_{Slave}} \cdot K_{User} = \frac{\frac{39}{17}}{\frac{1}{1}} \cdot \frac{10.0000[mm/rev.]}{7.5000[mm/rev.]} \cdot \frac{1}{2} = \frac{390.0000}{255.0000} = \frac{26}{17}$$

Setting of the electrical shaft factor $K_{ElectricalShaft}$

C00471/3	= 26	Numerator term
C00471/4	= 17	Denominator term (is only evaluated in terms of value)

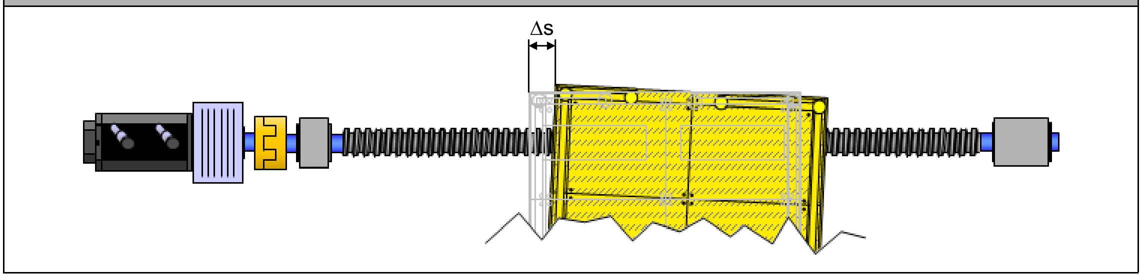
4.3.2 Positioning to master position ("Lock to master")

Before the master/slave interconnection can be reasonably actuated in electrical shaft mode, a potential inclination of the slave drive with regard to the master drive must be corrected. This correction can be mapped by means of a positioning process. The functionality that is referred to as "Lock to master" here is known by the designation "Lock to cam" from the field of electronic cam technology.

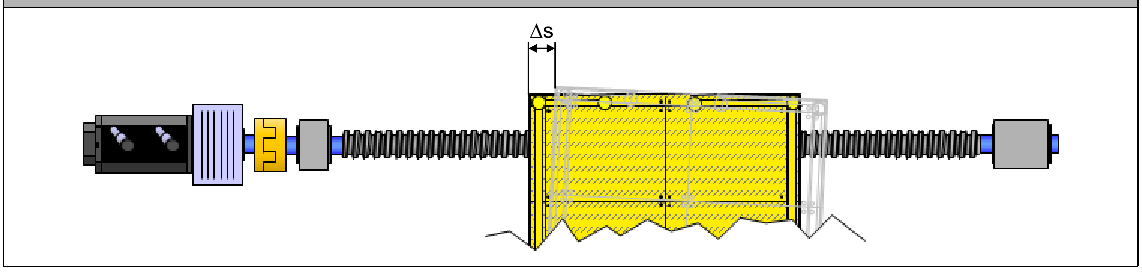


[4-2] "Lock to master" function (excerpt from the basic signal flow)

Inclination of the slave drive by the difference Δs before switching on the "Position follower" operating mode



Correct position of the slave drive after correction of the inclination Δs ("Lock to master done")



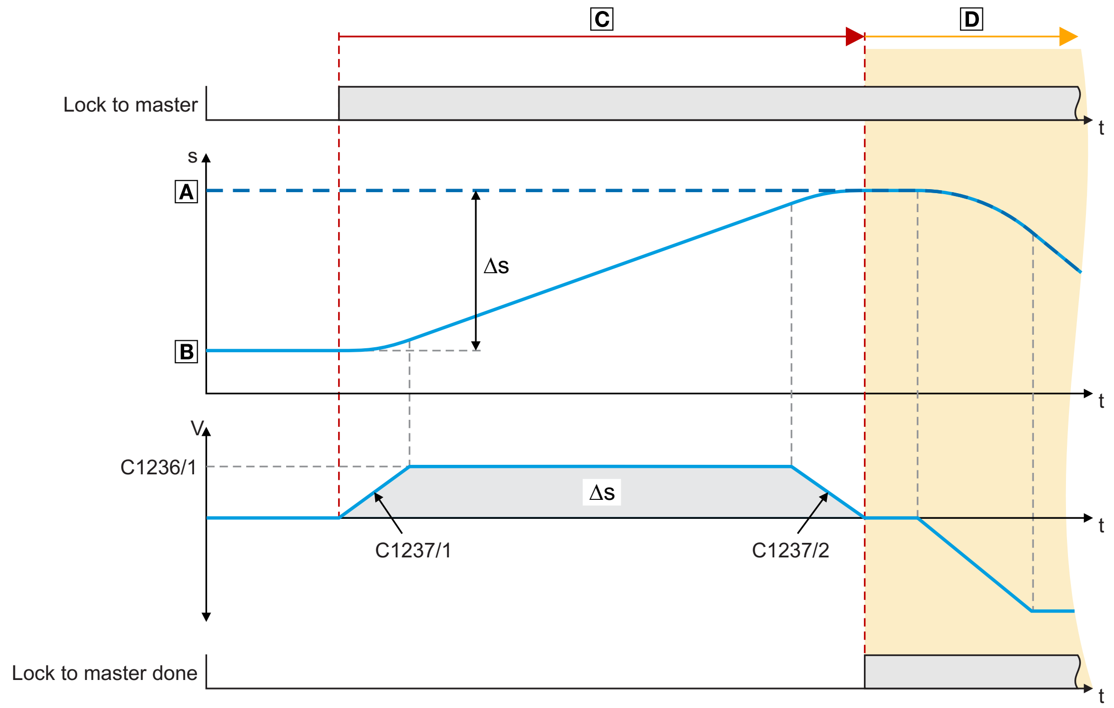
In the "Position Follower" technology application, the basic function "position follower" of the **Motion Control Kernel** (MCK) is used for the correction of the inclination. The correction is automatically executed if position follower operation is switched on.

- Position follower operation is switched on via bit 9 of the application control word (**L_ConvBitsToWord_3** FB). In the default setting, bit 9 is linked with digital input DI3.
- Positioning to the master position can only be executed when the home position is known.
- When positioning to the master position (correction of the inclination) has been completed ("Lock to master done"), feedback is given via digital output DO2 and bit 14 of the MCI process data output word 1.

Stop!

Only start the master drive when all slave drives have carried out the correction of the inclination successfully and reported the "Lock to master done" status! Otherwise a following error message will be output in position follower operation.

Example: Traversing profile for positioning to the master position (electrical shaft position)

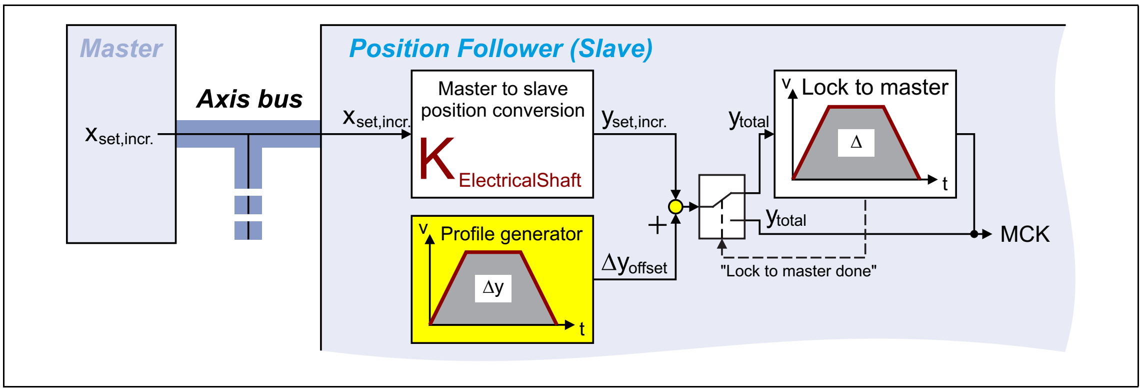


A	Master position (target position for "Lock to master" positioning)
B	Starting position of the slave drive
C	Slave drive corrects inclination Δs. **The master drive must not be traversed yet in this situation!**
D	Master drive traverses, slave drive follows (position follower operation active).

Parameter (Block)	**Possible settings**			**Info**
C01236/1 (LS_MotionControl Kernel)	-214748.3647	units/s	214748.3647	PosFollower: Sync. speed Synchronisation speed for approaching the setpoint position of the master.
	Lenze setting: 360.0000 units/s			
C01237/1 (LS_MotionControl Kernel)	-214748.3647	units/s^2	214748.3647	Pos follower: Sync. accel. Acceleration for ramp-up to synchronisation speed
	Lenze setting: 3600.0000 units/s^2			
C01237/2 (LS_MotionControl Kernel)	-214748.3647	units/s^2	214748.3647	Pos follower: Sync. decel. Deceleration for synchronisation speed ramp-down to standstill (to the setpoint position of the master).
	Lenze setting: 3600.0000 units/s^2			
C01238/1 (LS_MotionControl Kernel)	0.000	s	10.000	Pos follower: Sync. S-ramp time S-ramp time for the synchronisation speed ramps.
	Lenze setting: 0.000 s			

4.3.3 Shifting the set position (y offset)

In addition to stretching/compression the master position, the set position of the slave can be shifted by an offset distance with regard to the master position. In the signal flow, this offset couples in dowstream of the electrical shaft function (measuring system conversion), so that this is also referred to as an y offset.



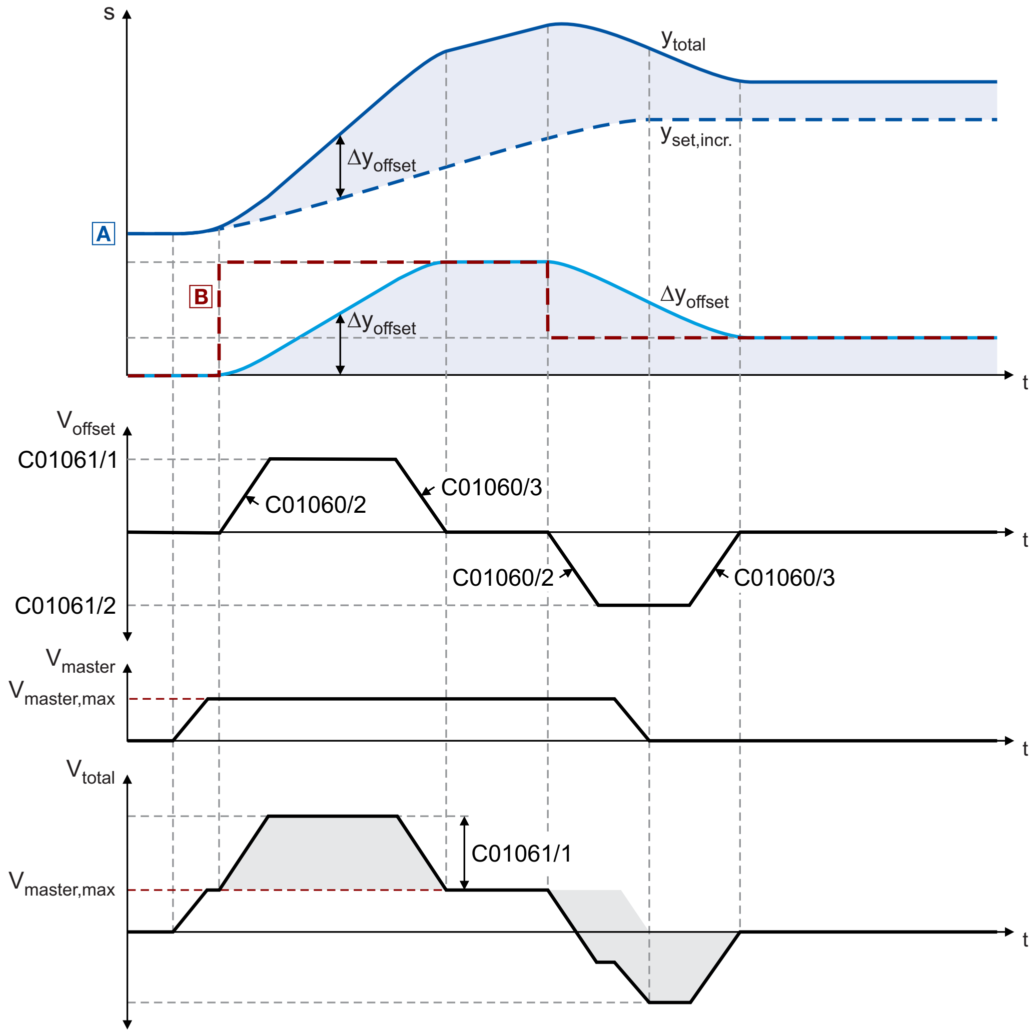
[4-3] y offset (excerpt from the basic signal flow)

- The y offset (Δy) is selected via C00475/1 in application units.
- The y offset (Δy) is controlled via a profile generator (**L_PosCtrlLin_1** FB), in order to provide for constant setpoint characteristics even if the offset is changed in a discontinuous manner. Like this, the y offset can also be changed during position follower operation.
- The overall setpoint y_{total} is composed of the electrical shaft setpoint position $y_{set,incr.}$ and the offset value Δy_{offset}.
- In terms of function, the y offset causes a shift of the slave setpoint position compared with the master position (also see the following illustration).

STOP

Stop!

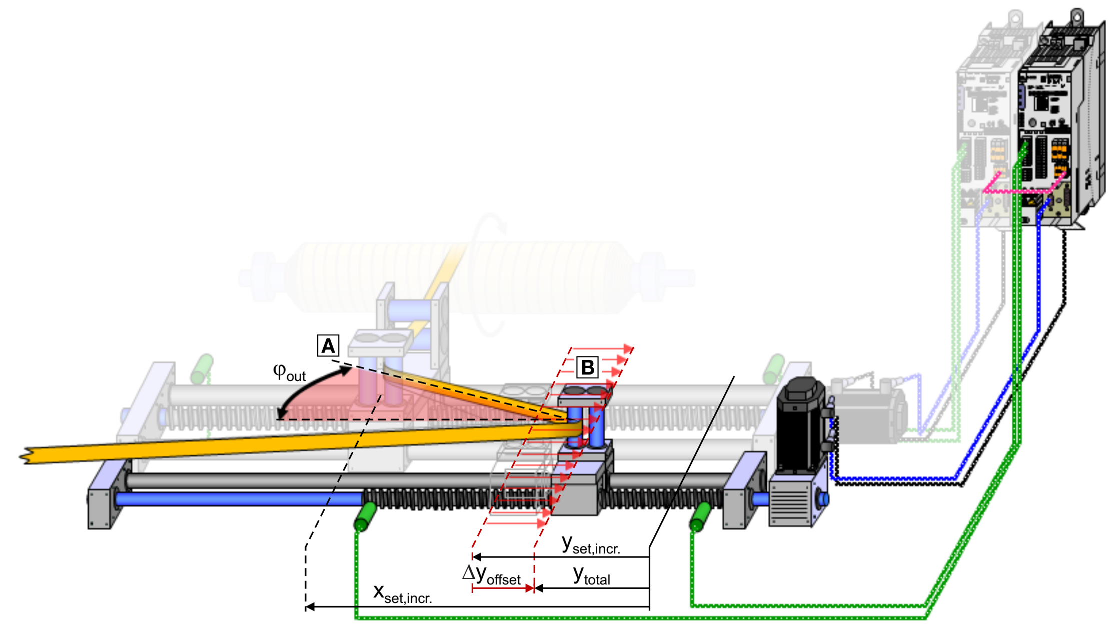
When the master and slave axis are mechanically coupled to each other (like for example the traversing platform shown in the "Application ranges" chapter), the y offset function in the slave drive must not be used, because otherwise an inclination in the machine mechanics will occur!

Example: Superimposition of the electrical shaft setpoint position with a y offset	
 	
A	Starting position of the master drive
B	Y offset selection via C00475/1 (possibly abrupt changes in values)

Parameter (Block)	Possible settings (Lenze setting printed in bold)			Info
Y offset (shift of the slave position follower setpoint position with regard to the master position)				
C00475/1 (LS_ParFreeUnit_1)	-214748.3647	units	214748.3647	Y offset
	Lenze setting: 0.0000 units			
When the y offset is changed during position follower operation, the following parameters are to be set:				
C01060/2 (L_PosCtrlLin_1)	0.010	s	130.000	Acceleration ramp
	Lenze setting: 10 s			
C01060/3 (L_PosCtrlLin_1)	0.010	s	130.000	Deceleration ramp
	Lenze setting: 10 s			
C01061/1 (L_PosCtrlLin_1)	-15000	rpm	15000	Forward motion (positive speed) • The value is only evaluated with regard to its amount.
	Lenze setting: 50 min-1			
C01061/2 (L_PosCtrlLin_1)	-15000	rpm	15000	Return motion (negative speed) • The value is only evaluated with regard to its amount.
	Lenze setting: 50 min-1			

Example: Shift of the additional guide (slave) for a traversing control

In order to keep the discharge angle of the material φ_{out} low at the additional guide, the additional guide is to act as far back as possible in the traversing range. This state can be achieved by a correspondingly negative Δy_{offset}:



A	Traversing support (master)	B	Additional guide (slave)

Calculation of the overall setpoint for the slave drive

$$y_{total} = x_{set,incr.} \cdot K_{ElectricalShaft} + \Delta y_{offset} = y_{set,incr.} + \Delta y_{offset}$$

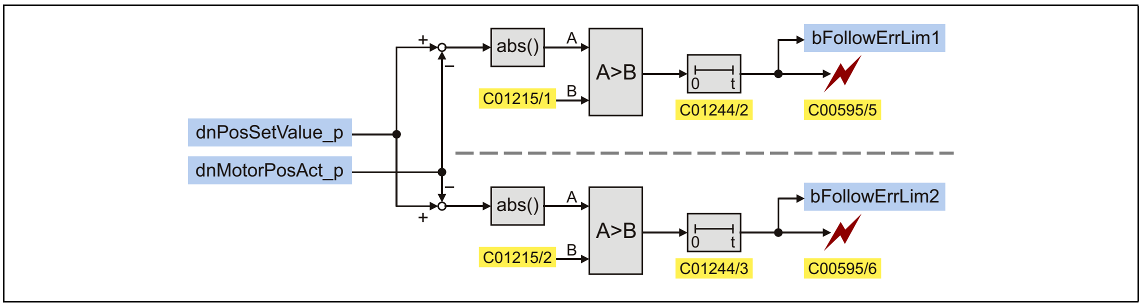
4.4 Monitoring functions

4.4.1 Following error monitoring system

The difference between set position and actual position is called the following error. Ideally, the following error should be "0". The set position is created by the internal definition of the traversing profiles of the **Motion Control Kernel**. The actual position is created by the integration of the speed supplied by the position encoder. The following error is always compensated dynamically. If the position control is adjusted optimally, only a minimum following error occurs which does not increase continuously.

Certain processes, however, require that a defined limit as a difference between set position and actual position is not exceeded. If it is exceeded, it may have been caused by a mechanical blocking in the machine and the system part is not situated at the position defined at that time. In such a case, it makes sense to activate the "Fault" error response to make the motor torqueless.

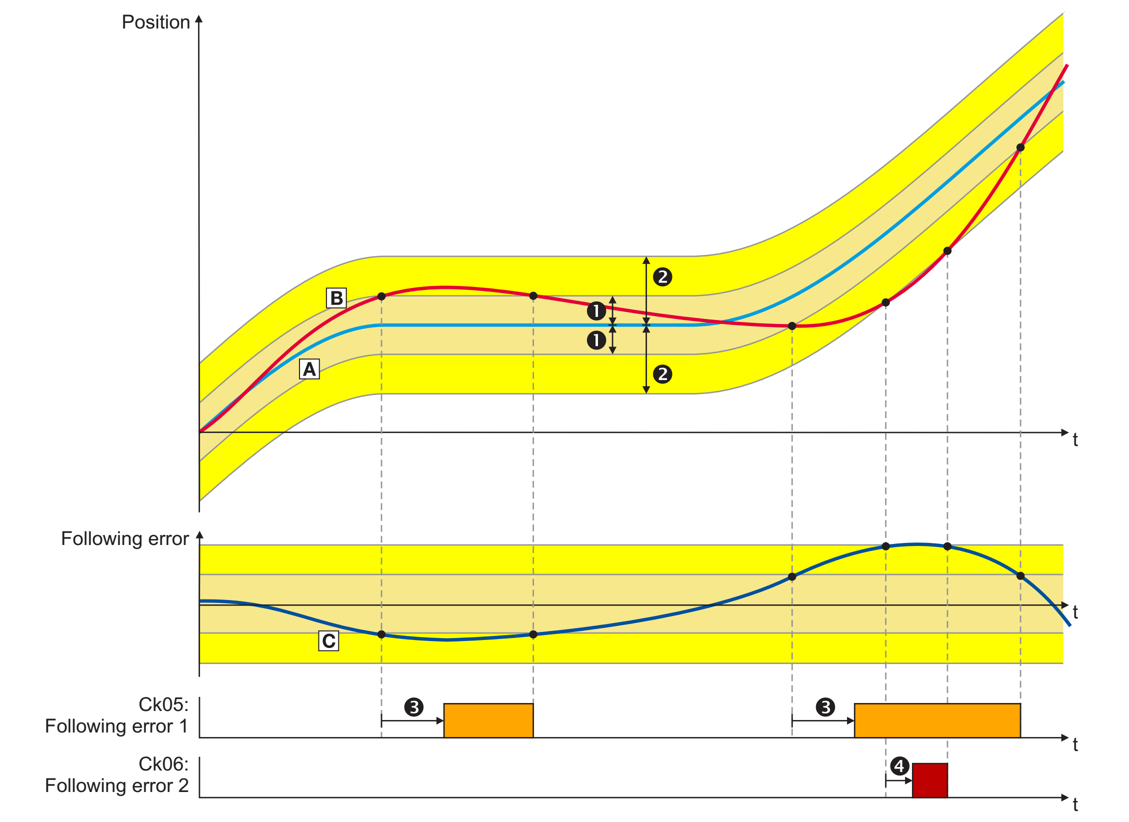
In the 8400 TopLine controller, two independent following error monitoring systems can be parameterised:



[4-4] Two-channel following error monitoring system

Parameter (Block)	Possible settings			Info
C01215/1 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 1 • The setting "0" deactivates following error monitoring 1
	Lenze setting: 5.0000 units			
C01215/2 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 2 • The setting "0" deactivates following error monitoring 2
	Lenze setting: 10.0000 units			
C01244/2...3 (LS_MotionControl Kernel)	0	ms	60000	Waiting time for following error monitoring 1 & 2 • In order to avoid that an error is triggered by acceleration and a narrow tolerance limit can be nevertheless monitored at standstill in the target, the response of the following error monitoring system can be delayed by setting a waiting time.
	Lenze setting: 0 ms			
C00595/5...6 (LS_SetError_1)				Response at the activation of following error monitoring • Subcode 5: response at the activation of following error monitoring 1. • Subcode 6: response at the activation of following error monitoring 2.
	0	No Reaction		
	1	Fault		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	**Warning**		
	6	Information		

Example of the evaluation of the following error



A	Target position	❶	Limit for following error monitoring 1 (C01215/1)
B	Position actual value	❷	Limit for following error monitoring 2 (C01215/2)
C	Current following error	❸	Waiting time for following error monitoring 1 (C01244/2)
		❹	Waiting time for following error monitoring 2 (C01244/3)

The above example clearly shows that there are two monitoring functions acting independently of each other. Normally, one following error monitoring function with a low tolerance is set as warning, and the other with a higher tolerance is set as TroubleQuickStop.

By means of the waiting times (C1244/2 and C1244/3), the error responses can be delayed, so that disconnection does not result immediately when the following error limits are temporarily exceeded. This "switch-on delay" makes it possible to effectively distinguish between dynamic processes and actual error states such as mechanical obstacles or sluggishness.

4.4.2 Limit position monitoring

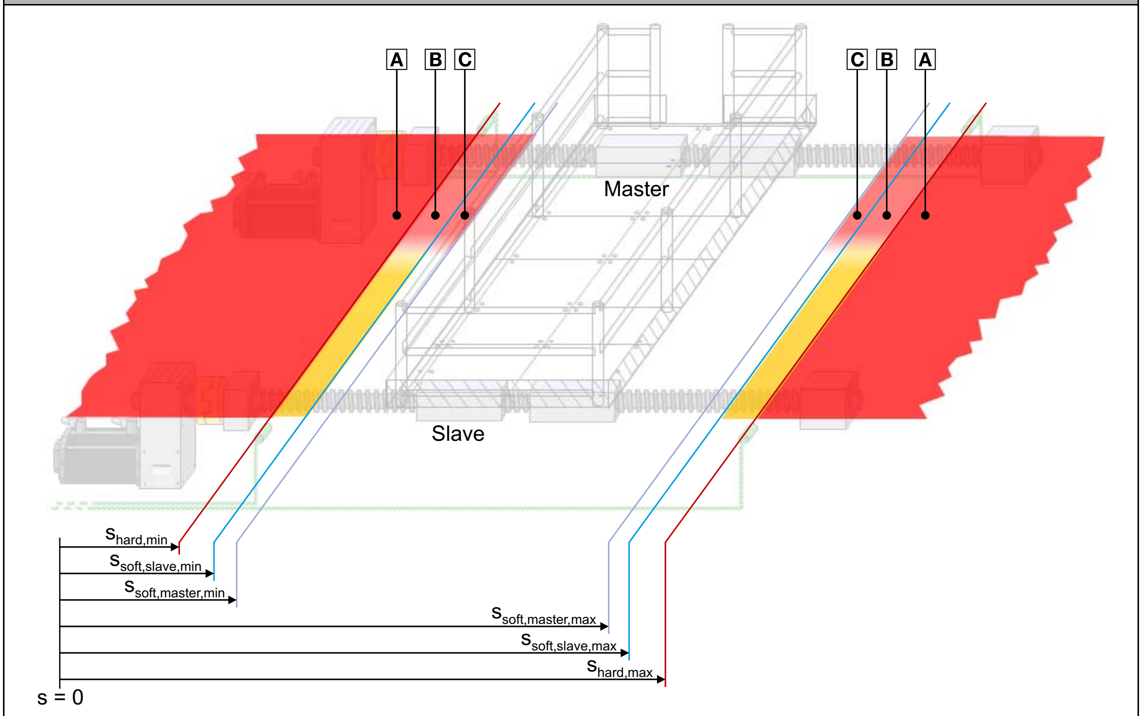
For safety reasons, drives with a limited traversing range must always be made safe using corresponding safety mechanisms. These are firstly the hardware limit switches, which must effect a standstill of the axes as quickly as possible during the approaching process. In addition, the parameterisable software limit positions are to inhibit all travel commands which would entail an exit of the permissible travel range.

Note!

In der "Position Follower" technology application, no evaluation of hardware limit switches is prepared by default. However, if required, an evaluation can be retrofitted by means of only a few steps (see "Hardware limit switch" subchapter).

In practice, the software limit positions are always placed within a safety distance from the limit switches (see the following example).

Example: Areas protected by hardware limit switches and software limit positions



A	Area protected by hardware limit switches
B	Area in the traversing range of the master drive, protected by software limit positions
C	Area in the traversing range of the slave drive, protected by software limit positions

Take the following items into consideration for placing the limit positions:

- The hardware limit switches must be mounted with a sufficient clearance in front of the mechanical limit positions of the linear axis. The clearance to the mechanical limit position should be calculated for a quick stop (QSP) from maximum speed, so that in the event of an error, standstill is attained safely before the mechanical limit position is reached.
- The software limit positions of the slave position follower must be placed with a sufficient distance from the hardware limit switch positions. The distance should be calculated for a quick stop (QSP) from maximum speed without the hardware limit switches being approached.
- The software limit positions of the master must take effect earlier than the software limit positions of the slaves. Like this, error activation by approaching the software limit positions of a slave is prevented.

4.4.2.1 Hardware limit switch

For an evaluation of hardware limit switches, the two digital inputs in the function block editor that are used for connection of the hardware limit switches have to be connected to the *bLimitSwitchNeg* and *bLimitSwitchPos* monitoring inputs of the **LS_MotionControlKernel** system block.

The following example exemplarily shows the connections required if the negative limit switch is connected to digital input DI1 and the positive limit switch is connected to digital input DI2:

1. **LS_DigitalInput**.*bIn1* **→ LS_MotionControlKernel**.*bLimitSwitchNeg*
2. **LS_DigitalInput**.*bIn2* **→ LS_MotionControlKernel**.*bLimitSwitchPos*

Stop!

The connection of the limit switch signals must be executed in a fail-safe fashion:

- LOW level = limit switch approached (activated)
- HIGH level = limit switch not approached (not activated)

For this switching logic, the level inversion for the two digital inputs to which the limit switches are connected is to be activated in C00114.
(If DI1 and DI2 are used, bit 1 and bit 2 are to be set to "1" in C00114.)

The limit switches are only evaluated if the limit switches for the respective operating mode have been activated (see the following table)!

Operating mode	Hardware limit switch effective
Homing	Depending on the homing mode selected (see description of the homing modes in the reference manual/online help of the inverter)
Manual jog	Yes (adjustable in C01230 - bit 2)
Position follower	Yes (adjustable in C01218 - bit 2)
Positioning	Yes

Parameter (Block)	Possible settings		Info
C00595/1...2 (LS_MotionControl Kernel)			Response at the activation of limit position monitoring • Subcode 1: response at the approach of the positive limit switch. • Subcode 2: response at the approach of the negative limit switch.
	0	No Reaction	
	1	Fault	
	3	**TroubleQuickStop**	
	4	WarningLocked	
	5	Warning	
	6	Information	

Behaviour when hardware limit switches are active

If one of the two monitoring inputs is set to TRUE, in the Lenze setting the "TroubleQuickStop" error response is triggered: Irrespective of the setpoint selection, the drive is brought to a standstill in the deceleration time set for the quick stop function. Depending on the error response parameterised, the drive can then only be traversed again after the error has been acknowledged.

Note!

Hardware limit switches approached in the case of the 8400 TopLine inverter can only be retracted via the "Manual jog" operating mode.

4.4.2.2 Software limit positions

The parameterisable limit positions are used by the software to limit the traversing range.

Stop!

The software limit positions are only evaluated and monitored if the home position is known to the drive and the software limit positions for the respective operating mode have been activated (see following table)!

If the drive is stopped at a high deceleration, depending on the mass inertia and friction, an overvoltage in the DC bus may occur, and pulse inhibit is set as error response. The use of a brake resistor can prevent this response.

Operating mode	Software limit positions active (if home position is known)
Homing	Yes
Manual jog	Yes (adjustable in C01230 - bit 3)
Position follower	Yes (adjustable in C01218 - bit 3)
Positioning	Yes

Parameter (Block)	Possible settings			Info
C01229/1...2 (LS_MotionControl Kernel)	-214748.3647	units	214748.3647	Positive and negative software limit position for limiting the valid traversing range • Subcode 1: Positive software limit position • Subcode 2: Negative software limit position **Note:** The positive software limit position must be set to a greater value than the negative software limit position!
	Lenze setting: ±100000.0000 units			
C00595/3...4 (LS_MotionControl Kernel)				Response at the activation of limit position monitoring • Subcode 3: response at overtravelling the positive software limit position (C01229/1). • Subcode 4: response at overtravelling the negative software limit position (C01229/2).
	0	No Reaction		
	1	Fault		
	3	**TroubleQuickStop**		
	4	WarningLocked		
	5	Warning		
	6	Information		

Behaviour in the case of active software limit positions

Note!

The "travel commands" mentioned in the following description are no speed setpoint selections. In the "Speed follower" and "Position follower" operating modes, an acknowledged software limit position error ensures that traversing to the impermissible travel range remains possible afterwards. This is because in these two operating modes, there is no preview of whether a software limit position is approached with a setpoint selection.

If the software limit positions are active, travelling commands that would result in the exit from the permissible travel range can no longer be executed.

If the drive is already outside the permissible travel range and the software limit positions have been activated, only travel commands that result in the drive moving back into the permissible travel range can be executed.

If the software limit positions are active, and one of the software limit positions is overtravelled, in the Lenze setting the "TroubleQuickStop" error response is triggered: Irrespective of the setpoint selection, the drive is brought to a standstill in the deceleration time set for the quick stop function. Depending on the error response parameterised, the drive can then only be traversed again after the error has been acknowledged.

4.4.3 Bus monitoring

If an "Electrical shaft" is operated via axis bus, or generally via a bus system, data transmission errors, for example due to EMC interference, must always be expected. In order to prevent a mechanical offset of the drives in the event of a fault that is only temporary, the **L_Interpolator_1** FB is used in the slave application. This FB is equipped with an internal correction mechanism, which, in the case of a missing data telegram, provides for an automatic offset correction. If transmission cycles > 1 ms are used, the interpolation function must be activated and parameterised.

Parameterisation of the error handling:

Parameter	Possible settings			Info
C00591/1				Resp. to axis bus data error • Response in the case of an accumulation of data errors
	0	No Reaction		
	1	**Fault**		
	2	Trouble		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	Warning		
	6	Information		
C02431/3	0	ms	65000	Axis bus monitoring time • Tolerance time before the data error is triggered • For systems that are coupled in a mechanically rigid fashion, enter a value that is as small as possible.
	Lenze setting: 100 ms			

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

Lenze Drives GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
+49 5154 82-0
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Inverter

8400

"Position Sequencer" technology application
for 8400 TopLine C

Software manual EN

13467299

Lenze

Contents

1 About this documentation

This documentation described the software-based solution of a task. The transferability of the described solution to the respective application case needs to be checked by the user. If required, the user has to adapt the solution accordingly. Thus, physical aspects as e.g. drive dimensioning is not part of this documentation.

Danger!

The controller is a source of danger which may lead to death or severe injury of persons.

To protect yourself and others against these dangers, observe the safety instructions before switching on the controller.

Please read the safety instructions provided in the **8400 mounting instructions** and in the **8400 hardware manual**. Both documents are supplied with the controller.

Target group

This documentation addresses to all persons

- who want to use the "Position Sequencer" technology application for the 8400 TopLine inverter, and
- who are familiar with handling the device and the »Engineer« software.

Validity

The information in this documentation are valid for the following technology applications:

Technology application	from version
Position Sequencer	2.0

Screenshots/application examples

All screenshots provided in this documentation are application examples. Depending on the software version of the controller and the version of the installed »Engineer« software, the screenshots in this documentation may differ from the representation in the »Engineer«.

Tip!

Information and tools for Lenze products are provided in the download area at

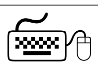
http://www.lenze.com → Download

1.1 Document history

Version			Description
1.0	07/2014	TD05	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is generally used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the prefix "0x" is used. Example: 0x60F4
Binary number	0b	For binary numbers, the prefix "0b" is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that only applies to a certain controller software version or higher is identified accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software ...
Window	*italics*	The *Message* window ... / The *Options* dialog box...
Variable name		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Press **<F1>** to open the online help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...
Hyperlink	underlined	Optically highlighted reference to another topic. In this documentation activated by mouse-click.
Icons		
Page reference	(📖 4)	Optically highlighted reference to another page. In this documentation activated by mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning	
Engineering Tools	Software solutions for simple engineering at all stages	
		»EASY Navigator« – Ensures easy operator guidance • All practical Lenze engineering tools at a glance • Tools can be selected quickly • Clearly arranged, simplifying the engineering process from the start
		»EASY Starter« – Simple tool for service technicians • Especially developed for the commissioning and maintenance of Lenze devices • Graphical user interface with few buttons • Simple online diagnostics, parameterisation and commissioning • No risk of accidentally changing the application • Ready applications can be loaded to the device
		»Engineer« – Multi-device engineering • For all products from our L-force portfolio • Practice-oriented user interface • Easy handling due to graphical user interfaces • Suitable for all project stages (configuration, commissioning, production) • Parameter setting and configuration
Code	Parameter used for controller parameterisation or monitoring. The term is usually called "index".	
Subcode	If a code contains several parameters, these are stored in "subcodes". This Manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). The term is usually called "subindex".	
Lenze setting	This setting is the default factory setting of the device.	
FB Editor	Abbreviation for function block editor. Graphic interconnection tool which is available in the »Engineer« for function block interconnections on the **FB Editor**.	
Function block	General designation of a function block for free interconnection in the FB Editor. Ein Funktionsbaustein (kurz: "FB") kann mit einer integrierten Schaltung verglichen werden, die eine bestimmte Steuerungslogik enthält und bei der Ausführung einen oder mehrere Werte liefert. Example: "L_Arithmetic_1" (FB for arithmetic operations) Many function blocks are available several times (e.g. L_And_1, L_And_2, and L_And_3).	
System block	In the function block editor of the »Engineer«, system blocks provide interfaces to basic functions, "free codes", and to the hardware of the inverter (e.g. to the digital inputs). Each system block is available only once.	
Port block	Block for implementing the process data transfer via a fieldbus	
LP	Abbreviation for **L**enze **P**ort block Example: "LP_CanIn1" (CAN1 port block)	
LS	Abbreviation for **L**enze **S**ystem block Example: "LS_DigitalInput" (system block for digital input signals)	
MCI	Abbreviation for **M**otionbus **C**ommunication **I**nterface (fieldbus interface) The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system.	
Technology application	A technology application is a drive solution based on the experience and know-how of Lenze in which function blocks interconnected to a signal flow form the basis for implementing typical drive tasks.	
USB diagnostic adapter	The USB diagnostic adapter is used for the operation, parameterisation, and diagnostics of the controller. Data are exchanged between the PC (USB connection) and the controller (diagnostic interface on the front) via the diagnostic adapter. Order designation: E94AZCUS	

1.4 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling

2 Features of the technology application

With the "Position Sequencer" technology application, the drive can execute parameterisable travel profiles. The program flow is defined on the basis of a sequence table.

2.1 Functional overview

- Sequence control for several successive positioning steps with pause and abort functions and different auxiliary functions (e.g. branching, counting, waiting)
- Positioning in different positioning modes
  - Point-to-point positioning
  - Touch probe positioning (residual path positioning)
  - Profile linkage with velocity changeover (overchange)
- Homing (homing/reference setting)
  - For a homing process, different modes are provided.
- Profile data management
  - Support of S-profiles (jerk limitation)
  - Separate setting for acceleration and deceleration
- Use of the state machine of the Motion Control Kernel (MCK) and the following basic drive functions:
  - Homing
  - Manual jog
  - Positioning
  - Holding brake control
- Limit position monitoring (hardware limit switches and software limit positions)
- Following error monitoring system
- Control/status signals optionally via digital terminals and fieldbus interface (MCI)

Tip!

By means of the "Position Sequencer" technology application, the 8400 TopLine can also be used as master for slave drives with the "Electrical Shaft Slave" or "Position Follower" technology application.

2.2 Application ranges

- Transport devices
- Rotary tables
- Feed drives
- Dosing units
- Hoists
- ...

2.3 System requirements

The technology application was created with the L-force »Engineer« V2.20 and can only be used with the versions V2.20 or higher.

Software

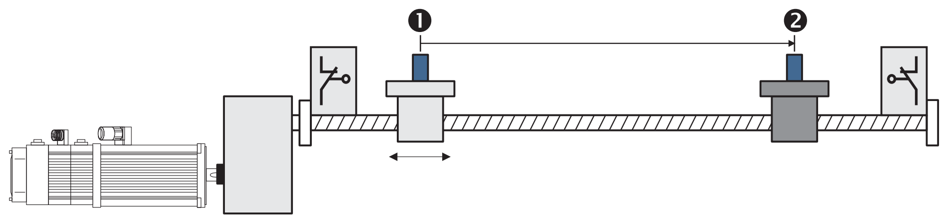
Product	Order designation	from version
L-force »Engineer« HighLevel	ESPEV-EHNNN	2.20

Hardware

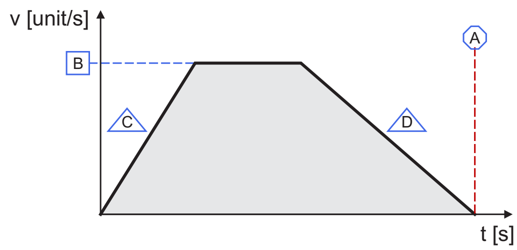
Product	Order designation	from hardware version	from software version
Inverter Drives 8400 TopLine C	E84AVTCxxxxx	VD	13.00

2.4 Basics of positioning

Positioning means that a workpiece/tool or material is moved from a starting position ❶ to a defined destination ❷:



To carry out positioning, a travel profile has to be stored in the drive controller for at least the following profile parameter:



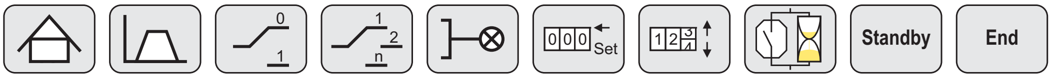
Icon	Profile parameter
A	**Position** Target position or distance to be traversed.
B	**Velocity** Maximum speed during the positioning process.
C	**Acceleration** Maximum acceleration during the positioning process.
D	**Delay** Maximum deceleration during the positioning process.

- A profile describes a motion task that can be converted into a rotary motion of the motor shaft by the **Motion Control Kernel** in the "Positioning" operating mode.
- A positioning process can be composed of a large number of profiles that are executed in a fixed manner.
- For the 8400 TopLine inverter, 15 different profiles can be parameterised.

A detailed explanation of all profile parameters can be found in the appendix in the description for the "Positioning" action type. (📖 44)

2.5 Positioning sequence control

For the positioning sequence control, the **L_Sequencer_1** FB is used. This FB processes a positioning program on the basis of a sequence table (also referred to as "sequencer"), which can contain up to 100 references to so-called "Actions".



- An action comprises a clear functionality which is described with a few parameters.
- Different action types are available which serve to realise, for instance, program branching, switching operation, waiting times and counters.
- A certain number of actions are available from every action type which can be parameterised individually.
- An action can be called from several positions in the sequence table.
- After an action has been processed, the action entered in the sequence step of the sequence table is automatically processed unless a branching causes a jump to another step in the sequence table.
- Maximally one action can be processed per calculation cycle.
- The sequence table and the actions themselves are represented by parameters (codes with subcodes).

A detailed description of the action types can be found in the appendix. (📖 42)

The **L_Sequencer_1** FB is described in detail in the reference manual/online help for the inverter in the "Function library »function blocks" chapter.

Note!

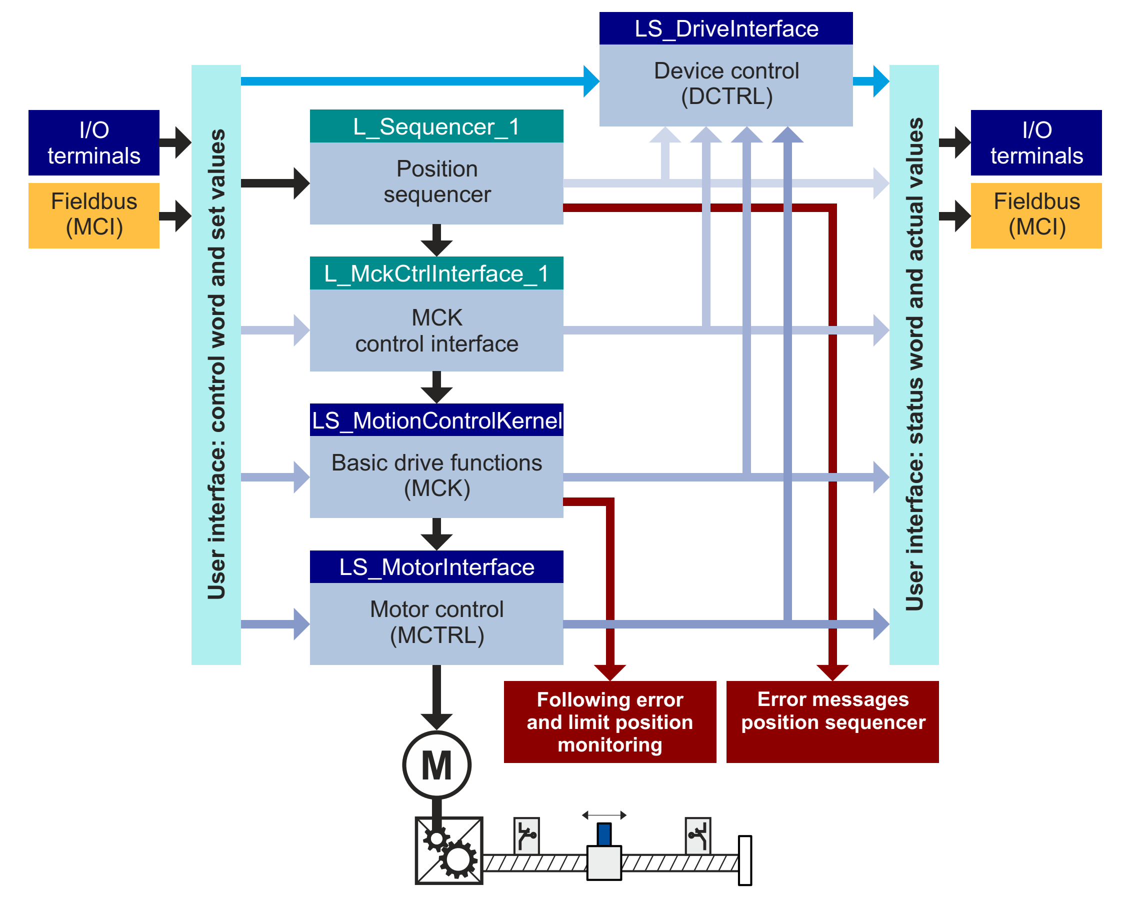
Additional logic operations in the application prevent the positioning program from being started if the basic function "Manual jog" or "Homing" is activated. On the other hand, it is avoided that these two basic functions can be activated if the positioning program is being processed.

2.6 Basic signal flow

In the technology application, function blocks and system blocks are interconnected so that a positioning sequence control can be implemented for the application ranges mentioned before.

The following describes the principal signal flow with the essential functions.

Basic signal flow of the "Position Sequencer" technology application



2.7 Parameter setting in the FB Editor view

You can make the settings of the application-specific parameters directly in the FB Editor. This has the advantage that the signal flow can be traced. The interaction of the modules becomes clear. Moreover, you can reconfigure the I/O interconnection using the FB Editor and carry out an online monitoring of the application running in the device (e.g. for diagnostic purposes).

- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.
- Colour codes and comments support you in handling the FB Editor.
  - The areas highlighted in turquoise represent the "user interface". If required, the pre-assignment of the I/O terminals can be adapted here and a control via the fieldbus interface (MCI) can be established.
  - In the areas highlighted in yellow, application-specific settings are required.

Detailed information on how to work with the FB Editor can be found in the reference manual/online help of the controller in the chapter "Working with the FB Editor".

2.8 Pre-assignment of the I/O terminals

<table>
<tr><th>Terminal</th><th colspan="3">Function</th></tr>
<tr><td colspan="4">Digital input terminals</td></tr>
<tr><td>X5/RFR</td><td colspan="3">Controller enable</td></tr>
<tr><td rowspan="8">X5/DI1
X5/DI2</td><td colspan="3">Connection of positive/negative limit switches for monitoring of the travel range limits
• The connection is configured in a fail-safe fashion (LOW = limit switch activated).</td></tr>
<tr><td colspan="2">DI1</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Positive limit switch approached (activated)</td></tr>
<tr><td colspan="2">HIGH</td><td>Positive limit switch not approached (not activated)</td></tr>
<tr><td colspan="2">DI2</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Negative limit switch approached (activated)</td></tr>
<tr><td colspan="2">HIGH</td><td>Negative limit switch not approached (not activated)</td></tr>
<tr><td colspan="3">Note!
If no limit switches are available:
1. Keep terminals DI1 and DI2 unconfigured.
2. Deactivate the inversion of DI1 and DI2: set bit 0 and bit 1 to "0" in C00114.</td></tr>
<tr><td rowspan="4">X5/DI3</td><td colspan="3">Start positioning program</td></tr>
<tr><td colspan="2">DI3</td><td>Function</td></tr>
<tr><td colspan="2">LOW↗HIGH</td><td>Start positioning program</td></tr>
<tr><td colspan="3">Tip!
The "USER" LED status display on the front of the inverter is lit when the positioning program is running.</td></tr>
<tr><td rowspan="4">X5/DI4</td><td colspan="3">Reference switch connection</td></tr>
<tr><td colspan="2">DI4</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Reference switch activated</td></tr>
<tr><td colspan="2">HIGH</td><td>Reference switch not activated</td></tr>
<tr><td rowspan="6">X5/DI5
X5/DI6</td><td colspan="3">Manual jog</td></tr>
<tr><td>DI5</td><td>DI6</td><td>Function</td></tr>
<tr><td>LOW</td><td>LOW</td><td>-</td></tr>
<tr><td>HIGH</td><td>LOW</td><td>Manual jog in positive direction</td></tr>
<tr><td>LOW</td><td>HIGH</td><td>Manual jog in negative direction</td></tr>
<tr><td>HIGH</td><td>HIGH</td><td>- / Manual jog in the direction selected first</td></tr>
<tr><td rowspan="5">X5/DI7</td><td colspan="3">Reset error message and positioning program</td></tr>
<tr><td colspan="2">DI7</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>No reset</td></tr>
<tr><td colspan="2">LOW↗HIGH</td><td>Reset error message</td></tr>
<tr><td colspan="2">HIGH</td><td>Reset positioning program</td></tr>
<tr><td colspan="4">Analog input terminals</td></tr>
<tr><td>X3/A1U</td><td colspan="3">- (not assigned, can be used freely)</td></tr>
<tr><td>X3/A2U</td><td colspan="3">- (not assigned, can be used freely)</td></tr>
</table>

Terminal	Function
Digital output terminals	
X4/DO1	HIGH ≡ "Drive is ready" state
X4/DO2	HIGH ≡ "Actual position is in target position window" state
X4/DO3	HIGH ≡ "Home position is known" state
X107/BD1, BD2	Control of a holding brake by the basic function "holding brake control"
X101/COM, NO	Relay contact closed ≡ "An error is pending" state
Analog output terminals	
X3/O1U	Actual speed value • Scaling: 10 V ≡ 100 % reference speed (C00011)
X3/O2U	Actual torque • Scaling: 10 V ≡ 100 % maximum torque (C00057)

3 Short setup of the technology application

3.1 Preconditions

For the execution of the short setup described in the following, the setting of the most important parameters (motor, feedback system, etc.) is assumed.

The "commissioning wizard 8400" serves to carry out a guided commissioning of the controller based on the Lenze setting of the parameters.

How to proceed:

1. Before switching on: Make sure that the inverter is inhibited (digital input terminal X5/RFR open).
2. Switch on voltage supply of the controller.
   For parameter setting and diagnostics of the controller without motor operation, an external 24-V supply through a safely separated power supply unit (SELV/PELV) is sufficient.
3. Establish a communication link between controller and Engineering PC, e.g. via USB diagnostic adapter (E94AZCUS):
   - connect the USB diagnostic adapter to the X6 diagnostic interface.
   - establish a connection between the USB diagnostic adapter and the PC via a free USB port.
4. Start »Engineer« on the Engineering PC, e.g. via the Windows® start menu:
   Start → All programs → Lenze → Engineering → L-force Engineer...
   After the program start, no project has been loaded first and the *start-up wizard* is displayed.

> You can find detailed information on the options of the start-up wizard and on the general use of the »Engineer« in the online help for the program which you can call with **[F1]**.

5. Create a new project or open a project already available.
6. Go to *Project View* and select the 8400 controller.
7. Click the icon to go online.
   After a connection to the controller has been established, the following status is displayed in the *Status line*:

8. Click the icon to start the *commissioning wizard 8400*.
   - Now the commissioning wizard guides you step by step through the setting of the important parameters for a quick commissioning.
   - The **Next** button can only be activated again after all parameter settings in the device have been reset via the **Load Lenze setting** button.
   - Execute the commissioning wizard right to the end.
   - You can skip the "Control mode" step by clicking **Next** (only relevant for "Speed actuating drive" technology application).

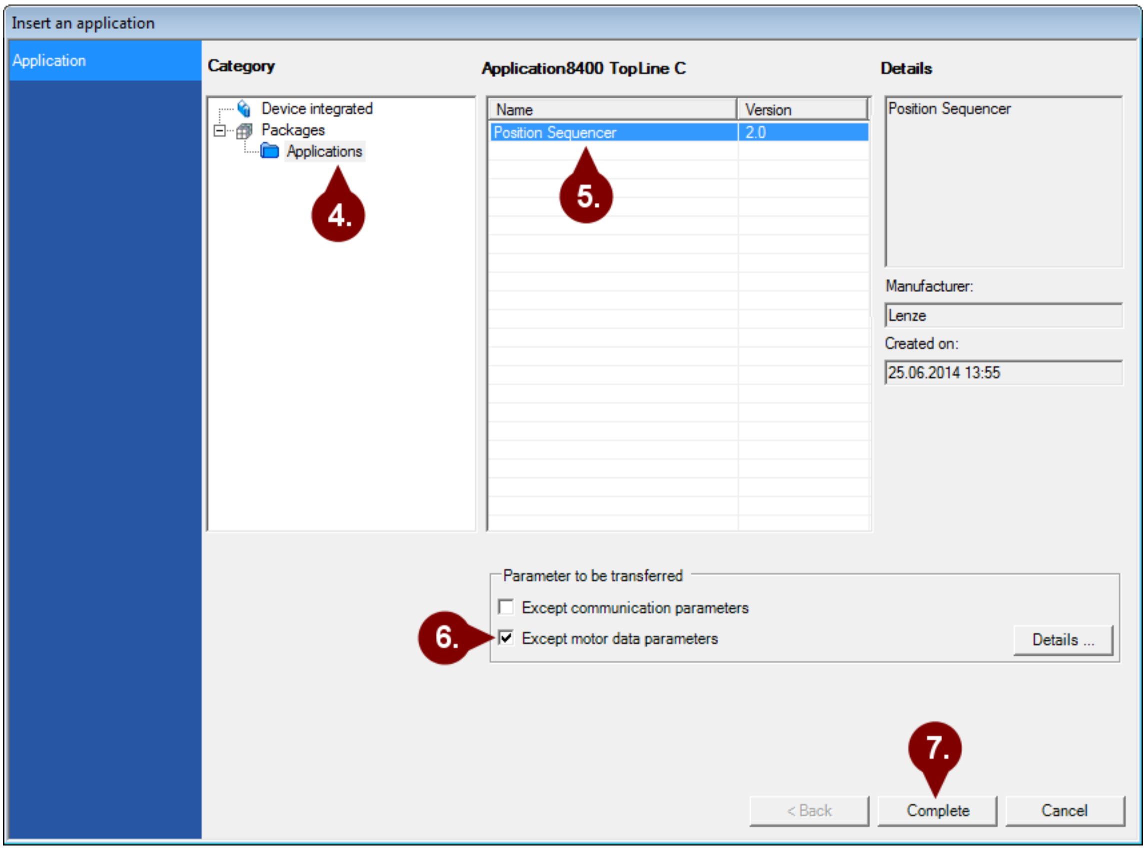
3.2 Step 1: Load "Position Sequencer" technology application

In the Lenze setting, the inverter uses the "Speed actuating drive" technology application integrated in the device. Execute the following steps to use the "Position Sequencer" technology application instead:

1. Select the controller in the *Project view*.
2. If there is still an online connection to the controller:

   Click the icon to go offline again.
   (the application can only be selected offline.)

3. Click the icon to select another application.
   The *Insert application* dialog box appears:



4. In the left field, select the "Packages" ➜ "Applications" category.
5. In the right field, select the "Position Sequencer" application.
6. Activate the **Except motor data parameters** option in order that the settings of the motor data parameters made before will not be overwritten.
7. Press **Complete** to close the dialog box again and load the selected application into the »Engineer« project.
8. Confirm the prompt on whether the current application is to be replaced by the "Position Sequencer" application with **Yes**.

3.3 Step 2 (optional): Establish control via the fieldbus interface (MCI)

In the default setting, the application is controlled via the digital input terminals. For higher automated systems, mostly data bus systems are used for controlling the drives.

For control via the fieldbus interface (MCI), the user interface (area highlighted in turquoise) is to be adapted accordingly in the function block editor.

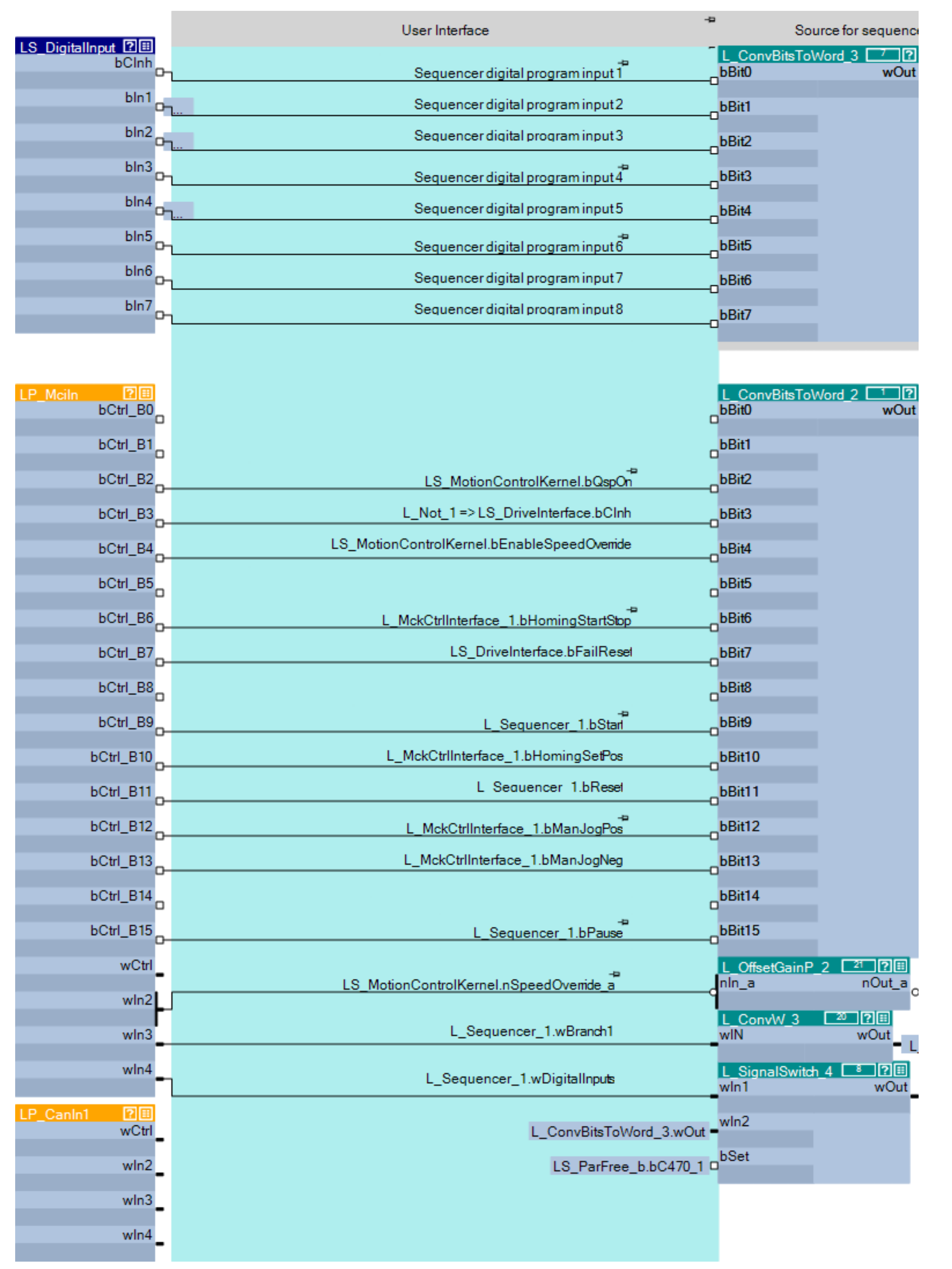
- The assignment of the outputs on the left to the inputs on the right can be changed at will.
- The inputs on the right are permanently linked to functions of the application.

Tip!

By means of some simple configuration changes, control via the integrated CANopen interface ("CAN on board") can also be implemented. The port blocks **LP_CanIn** and **LP_CanOut** required for this are already provided in the interconnection.

The following sample illustration shows the interconnection required for achieving the Pre-assignment of the process data input words described in the following subchapter:

Adaptation of the user interface for a control via the fieldbus interface (MCI)



[3-1] User interface for sequencer inputs, the application control word, and the speed override setpoint in the function block editor

3.3.1 Pre-assignment of the process data input words

After adapting the interconnection according to the [3-1] illustration, the process data input words for a control via the fieldbus interface (MCI) are assigned as follows:

<table>
<tr><th>Input words</th><th>Assignment</th></tr>
<tr><td>Word 1</td><td>Control word (for bit assignment see the following table)</td></tr>
<tr><td>Word 2</td><td>Setpoint for speed override</td></tr>
<tr><td>Word 3</td><td>Target for a variable branch in the positioning program
(input signal for action 1 of the "Variable branch" type)
• When the positioning program contains an action 1 of "Variable branching" type, branching is carried out in the corresponding step depending on the value available at this input.</td></tr>
<tr><td>Word 4</td><td>Sequencer inputs 1 ... 16 (bit coded)
• The "Positioning", "Branch", "Wait", and "Standby" action types are provided with the "Input for..." parameter. If it is non-zero, it designates the number of the sequencer input at which the positioning program expects the level defined for this purpose before it executes the action.
Note!
If the sequencer inputs are to be triggered via the digital inputs instead, C00470/1 has to be set to the value "1: True".</td></tr>
<tr><td>Word 5 ... 16</td><td>- (not preconfigured)</td></tr>
</table>

<table>
<tr><th>Control word</th><th colspan="3">Function</th></tr>
<tr><td>Bit 0</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 1</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 2</td><td colspan="3">1 ≡ activate quick stop (QSP)</td></tr>
<tr><td>Bit 3</td><td colspan="3">1 ≡ enable controller (RFR)</td></tr>
<tr><td>Bit 4</td><td colspan="3">1 ≡ activate speed override</td></tr>
<tr><td>Bit 5</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 6</td><td colspan="3">0 ≡ stop homing
1 ≡ start homing</td></tr>
<tr><td>Bit 7</td><td colspan="3">1 ≡ Reset fault (trip reset)</td></tr>
<tr><td>Bit 8</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 9</td><td colspan="3">1 ≡ start positioning sequence control</td></tr>
<tr><td>Bit 10</td><td colspan="3">1 ≡ set reference</td></tr>
<tr><td>Bit 11</td><td colspan="3">1 ≡ reset positioning sequence control</td></tr>
<tr><td rowspan="6">Bit 12 ... 13</td><td colspan="3">Manual jog</td></tr>
<tr><th>Bit 12</th><th>Bit 13</th><th>Function</th></tr>
<tr><td>0</td><td>0</td><td>-</td></tr>
<tr><td>1</td><td>0</td><td>Manual jog in positive direction</td></tr>
<tr><td>0</td><td>1</td><td>Manual jog in negative direction</td></tr>
<tr><td>1</td><td>1</td><td>- / Manual jog in the direction selected first</td></tr>
<tr><td>Bit 14</td><td colspan="3">- (not preconfigured)</td></tr>
<tr><td>Bit 15</td><td colspan="3">1 ≡ stop positioning sequence control (pause)</td></tr>
</table>

3.3.2 Pre-assignment of the process data output words

The **LP_MciOut** port block is already implemented in the technology application. The process data output words are assigned as follows:

Output words	Assignment
Word 1	Status word (for bit assignment see the following table)
Word 2	Actual speed value • Scaling: 16384 ≡ 100 % reference speed (C00011)
Word 3	Actual torque • Scaling: 16384 ≡ 100 % maximum torque (C00057)
Word 4 ... 16	- (not preconfigured)

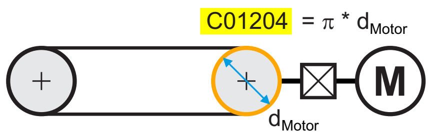
<table>
<tr><th>Status word</th><th colspan="6">Status</th></tr>
<tr><td>Bit 0</td><td colspan="6">1 ≡ Group error active (configurable in C00148)</td></tr>
<tr><td>Bit 1</td><td colspan="6">1 ≡ Inverter control inhibited (pulse inhibit is active)</td></tr>
<tr><td>Bit 2</td><td colspan="6">1 ≡ Drive controller is ready for operation</td></tr>
<tr><td>Bit 3</td><td colspan="6">1 ≡ Quick stop is active</td></tr>
<tr><td>Bit 4</td><td colspan="6">1 ≡ Setpoint torque is in the limitation</td></tr>
<tr><td>Bit 5</td><td colspan="6">1 ≡ actual position is inside the target position window</td></tr>
<tr><td rowspan="2">Bit 6</td><td colspan="6">During open-loop operation:
1 ≡ speed setpoint < comparison value (C00024)</td></tr>
<tr><td colspan="6">During closed-loop operation:
1 ≡ actual speed value < comparison value (C00024)</td></tr>
<tr><td>Bit 7</td><td colspan="6">1 ≡ Controller inhibited (controller inhibit is active)</td></tr>
<tr><td rowspan="14">Bit 8 ... 11</td><td colspan="6">Bit coded display of the active device status</td></tr>
<tr><td>Bit 11</td><td>Bit 10</td><td>Bit 9</td><td>Bit 8</td><td>Device status</td><td>Meaning</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>FirmwareUpdate</td><td>Firmware update function is active</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Init</td><td>Initialisation active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Ident</td><td>Identification active</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>ReadyToSwitchOn</td><td>Device is ready to start</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>SwitchedOn</td><td>Device is switched on</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>OperationEnabled</td><td>Operation</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>Trouble</td><td>Trouble active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>Fault</td><td>Fault active</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>TroubleQSP</td><td>TroubleQSP is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>SafeTorqueOff</td><td>Safe torque off is active</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>SystemFault</td><td>System fault active</td></tr>
<tr><td>Bit 12</td><td colspan="6">1 ≡ a warning is indicated</td></tr>
<tr><td>Bit 13</td><td colspan="6">1 ≡ a fault is active. The inverter is in the "Trouble" device state.
• The motor has no torque (is coasting) due to the inhibit of the inverter.
• The "Trouble" device status is automatically abandoned if the error cause has been removed.</td></tr>
<tr><td>Bit 14</td><td colspan="6">1 ≡ positioning program running</td></tr>
<tr><td>Bit 15</td><td colspan="6">1 ≡ Home position is known</td></tr>
</table>

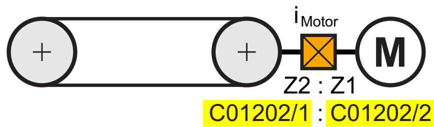
3.4 Step 3: Set commissioning parameters

For a quick commissioning, only the following application-specific parameters have to be set or their default setting has to be checked!

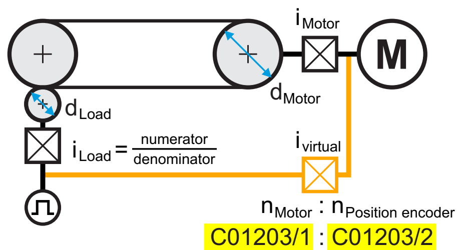
- In order that you quickly find the respective parameterisation dialog in the FB Editor, the following table lists the block related to each parameter.
- The icon in the head of the module, a double-click on the module, or the **Parameter...** command in the *Context menu* of the module serve to open the parameterisation dialog or the parameter list for the module.

<table>
<tr><th>Parameter
(Block)</th><th colspan="3">Possible settings
(Lenze setting printed in bold)</th><th>Info</th></tr>
<tr><td rowspan="3">C00470/1
(LS_ParFree_b)</td><td colspan="3"></td><td>Selection of the source for the sequencer inputs
• The sequencer inputs can be used for the "Positioning", "Branch", "Wait", and "Standby" actions to control the program flow by this.</td></tr>
<tr><td>0</td><td colspan="2">MCI interface</td><td>Sequencer inputs = process data input word 3</td></tr>
<tr><td>1</td><td colspan="2">Digital inputs</td><td>For this setting, cancel all double assignments of the digital inputs, so that they only work as sequencer inputs.</td></tr>
<tr><td rowspan="3">C00470/2
(LS_ParFree_b)</td><td colspan="3"></td><td>Selection of the signal to be output via axis bus</td></tr>
<tr><td>0</td><td colspan="2">Setpoint</td><td>Line data words 1 & 2 = speed setpoint integrated to an angle (path)
Line data word 3 = speed setpoint</td></tr>
<tr><td>1</td><td colspan="2">Actual value</td><td>Line data words 1 & 2 = actual speed value integrated to an angle (path)
Line data word 3 = actual speed value</td></tr>
<tr><td rowspan="3">C00470/4
(LS_ParFree_b)</td><td colspan="3"></td><td>Operation as position follower master</td></tr>
<tr><td>0</td><td colspan="2">Off</td><td>The signals are output via axis bus according to the setting in C00470/2.</td></tr>
<tr><td>1</td><td colspan="2">On</td><td>This setting is only required if the drive is to be used as master for a slave drive with the "Position Follower" technology application. The following signals are then output via axis bus:
Line data words 1 & 2 = current position
Line data word 3 = current speed
Note:
With this setting, homing of the master causes a step in the slave position follower if the current position is set to the home position!
<u>Remedy:</u> Inhibit the slave position follower when you are referencing the master!</td></tr>
<tr><td colspan="5">Machine parameters/axis settings</td></tr>
<tr><td rowspan="2">C01201/1
(LS_MotionControl Kernel)</td><td>0.0000</td><td>units</td><td>214748.3647</td><td rowspan="2">Cycle length
• For a modulo measuring system, the length of one cycle to the overflow is to be set.
• For a limited measuring system, the Lenze setting "0.0000" is to be retained.</td></tr>
<tr><td colspan="3">Lenze setting: 0.0000 units</td></tr>
</table>

Parameter (Block)	**Possible settings** (Lenze setting printed in bold)			**Info**
C01204 (LS_MotionControl Kernel)	0.0001	units/rev.	214748.3647	Feed constant • The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft. • The value is entered in application units relating to one gearbox revolution. • Schematic diagram of a conveyor drive: 
	Lenze setting: 360.0000 units/U			
C01206/1 (LS_MotionControl Kernel)				Motor mounting direction
	0	**not inverted**		Motor is mounted directly
	1	inverted		Motor is mounted with rotation by 180°

C01202/1..2
(LS_MotionControl Kernel)
	1		65535	Gearbox ratio motor - load
				• Set the gearbox ratio with mathematical precision in the two subcodes:
				  • Subcode 1: numerator term (Z2)
				  • Subcode 2: denominator term (Z1)
				$i_{Motor} = \frac{C01202/1}{C01202/2} = \frac{Z2}{Z1}$
				• Schematic diagram of a conveyor drive:
				

	Lenze setting: 1:1			
Position encoder				
If the motor encoder is used as position encoder, you do not need to parameterise a separate position encoder. Keep the Lenze setting for the following codes.				
C01206/2 (LS_MotionControl Kernel)				Position encoder mounting direction
	0	**not inverted**		Position encoder is mounted directly
	1	inverted		Position encoder mounted with rotation by 180

C01203/1..2
(LS_MotionControl Kernel)
	1		65535	Speed ratio for motor - position encoder
				• Set the "virtual" speed ratio between the motor and the external position encoder with mathematical precision in the two subcodes:
				  • Subcode 1: numerator term (motor speed)
				  • Subcode 2: denominator term (encoder speed)
				• Schematic diagram of a conveyor drive:
				
				• The "virtual" speed ratio can be calculated as follows:
				$i_{virtual} = \frac{C01203/1}{C01203/2} = \frac{i_{Motor}}{i_{load}} \cdot \frac{\pi \cdot d_{load}}{\pi \cdot d_{Motor}}$
				$i_{virtual} = \frac{C01202/1}{C01202/2} \cdot \frac{denominator}{numerator} \cdot \frac{\pi \cdot d_{load}}{C01204}$

	Lenze setting: 1:1			

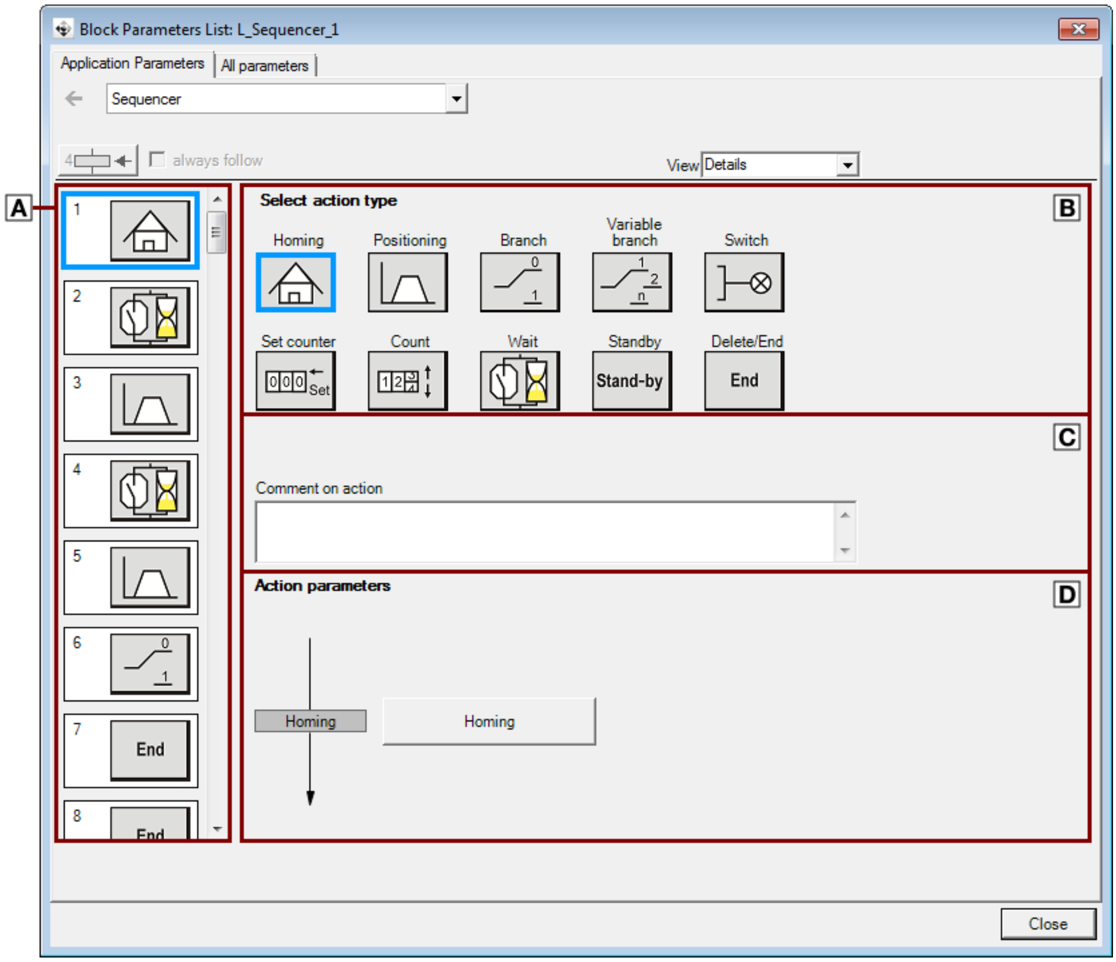
Parameter (Block)	Possible settings (Lenze setting printed in bold)			Info
Following error monitoring system				
C01215/1 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 1 • The setting "0" deactivates following error monitoring 1
	Lenze setting: 500 units			
C01215/2 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 2 • The setting "0" deactivates following error monitoring 2
	Lenze setting: 1000 units			
C00595/5...6 (LS_SetError_1)				Response at the activation of following error monitoring • Subcode 5: response at the activation of following error monitoring 1. • Subcode 6: response at the activation of following error monitoring 2.
	0	No Reaction		
	1	Fault		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	**Warning**		
	6	Information		
Limit position monitoring				
C01229/1...2 (LS_MotionControl Kernel)	-214748.3647	units	214748.3647	Positive and negative software limit position for limiting the valid traversing range **Note:** The software limit positions are only evaluated if • the home position is known to the drive, **and** • the software limit positions for the respective operating mode have been activated, **and** • the positive software limit position is set to a greater value than the negative software limit position!
	Lenze setting: 0.0000 units			

3.5 Step 4: create the positioning program (sequence table)

The sequence table is parameterised in the parameterisation dialog for the **L_Sequencer_1** FB.

Note!

In the default setting, the sequence table already contains a small "positioning program", which rotates the axis first by 360° in clockwise direction and then by 360° in counter-clockwise direction.



The parameterisation dialog is divided into four main areas:

Main area		Info
A	Sequence table (sequencer)	The sequence table consists of 100 fields which can be filled with actions.
B	Selection of action type	In this area, the different action types are provided for selection, by means of which the sequence table can be filled.
C	Comment on the action	Optionally, a comment on the action selected can be entered here.
D	Action parameters	In this area, the parameters for the action selected are set.

General procedure

Proceed as follows to define the desired program flow:

1. Select the program step ((1 ... 100) that is to be edited in the **Sequence table** on the left.
2. Select the action type for the program step selected by clicking it in the **Selection of action type** area.
   If more than one action is provided for the action type selected, the next free action in the **Selection of action number** list field is automatically proposed.
3. Optionally enter a comment on the action.
4. Set the action parameters.
   Depending on the action type, further parameterisation dialogs can be called via buttons.
5. Repeat steps 1 ... 4 until all actions required for the program flow have been parameterised.
6. Click **Close** to change back to the function block editor.

A detailed description of the action types can be found in the Appendix. (📖 42)

The **L_Sequencer_1** FB is described in detail in the reference manual/online help for the inverter in the "Function library »function blocks" chapter.

3.6 Step 5: Go online and transfer parameter set to the inverter

In order to set the current parameter settings in the controller to the settings in the project, transmit the parameter set to the controller.

1. Click the icon to go online.
2. Click the icon to transmit the parameter set to the controller.
3. After a successful transmission, click the icon to save the parameter set safe against mains failure in the integrated Memory Module.

3.7 Step 6: Enable inverter and start positioning program

After the parameter set has been transmitted to the inverter, the inverter can now be enabled and the control signals/setpoints can be selected via the corresponding interfaces.

▸ Pre-assignment of the I/O terminals (📖 12)
▸ Pre-assignment of the process data input words (📖 18)

Display parameters for diagnostic purposes

Parameter (Block)	Display area			Info
C01210/2 (LS_MotionControl Kernel)	-214748.3647	units	214748.3647	MCK: Set position • Display of the current setpoint position calculated by the MCK.
C01210/3 (LS_MotionControl Kernel)	-214748.3647	units	214748.3647	MCK: Actual position • Display of the current actual position calculated by an optional encoder system.
C01210/4 (LS_MotionControl Kernel)	-214748.3647	units	214748.3647	MCK: Following error • Display of the current following error as a difference between setpoint position and actual position.

Start positioning program

- The positioning program is started in a positive edge-controlled fashion via bit 4 of the application control word (**L_ConvBitsToWord_2** FB). In the default setting, bit 4 is linked with digital input DI3.
  - Counters and outputs are not reset automatically through this.
  - The positioning program started is processed up to its program end even if bit 4 is reset to FALSE again.
- A start via the fieldbus interface (MCI) is possible after the user interface has been adapted correspondingly. See commissioning; Step 2 (optional): Establish control via the fieldbus interface (MCI). (📖 16)

Resetting the positioning program (reset)

- Bit 11 of the application control word (**L_ConvBitsToWord_2** FB) can be used to reset the positioning program. In the default setting, bit 11 is linked with digital input DI7.
  - A reset can also be executed if the positioning program is interrupted.
  - If a positioning is active, the drive is brought to a standstill with the delay time for stop (C01251/1) without considering an acceleration override.
  - The program flow is cancelled ("program end").
  - The digital output signals, counters and timing elements are reset.
  - A possibly active action of the "Standby" type is aborted.
- A reset via the fieldbus interface (MCI) can be executed after the user interface has been adapted correspondingly. See commissioning; Step 2 (optional): Establish control via the fieldbus interface (MCI). (📖 16)

Interrupting the positioning program (pause)

- Bit 15 of the application control word (**L_ConvBitsToWord_2** FB) can be used to interrupt the current step of the positioning program (pause). In the default setting, bit 15 is linked with bit 15 of MCI process data input word 1. ▶ Pre-assignment of the process data input words (📖 18)

3.8 Step 7 (optional): Set optimisation parameters

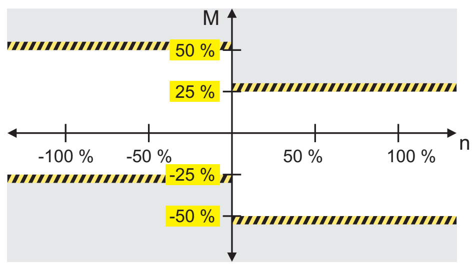
The following application-specific parameters are used for optimisation and can also be adapted during operation.

Stop!

If you change parameters in the »Engineer« during an online connection to the device, the changes are directly transferred to the device!

Tip!

Do not forget to save the parameter changes carried out with mains failure protection in the memory module implemented! (C00002/11 = "1: on/start")

Parameter (Block)	**Possible settings**			**Info**
Position controller				
C00472/1 (LS_ParFree_a)	-199.99	%	199.99	Limitation of the position controller output
	Lenze setting: 100 % reference speed (C00011)			
C00472/2 (LS_ParFree_a)	-199.99	%	199.99	Adaptation of the position controller gain
	Lenze setting: 100 % Vp (C00254)			
Torque limitation in motor mode/in generator mode				
The torque limitation set is always active.				Example: Definition of the torque limitations C00472/3 = 25 % C00472/4 = 50 % 
C00472/3 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in motor mode
	Lenze setting: 100 % maximum torque (C00057)			
C00472/4 (LS_ParFree_a)	-199.99	%	199.99	Torque limitation in generator mode
	Lenze setting: 100 % maximum torque (C00057)			
Following error monitoring system				
C01244/2...3 (LS_MotionControl Kernel)	0	ms	600000	Waiting time for following error monitoring 1 & 2 • In order to avoid that an error is triggered by acceleration and a narrow tolerance limit can be nevertheless monitored at standstill in the target, the response of the following error monitoring system can be delayed by setting a waiting time.
	Lenze setting: 0 ms			

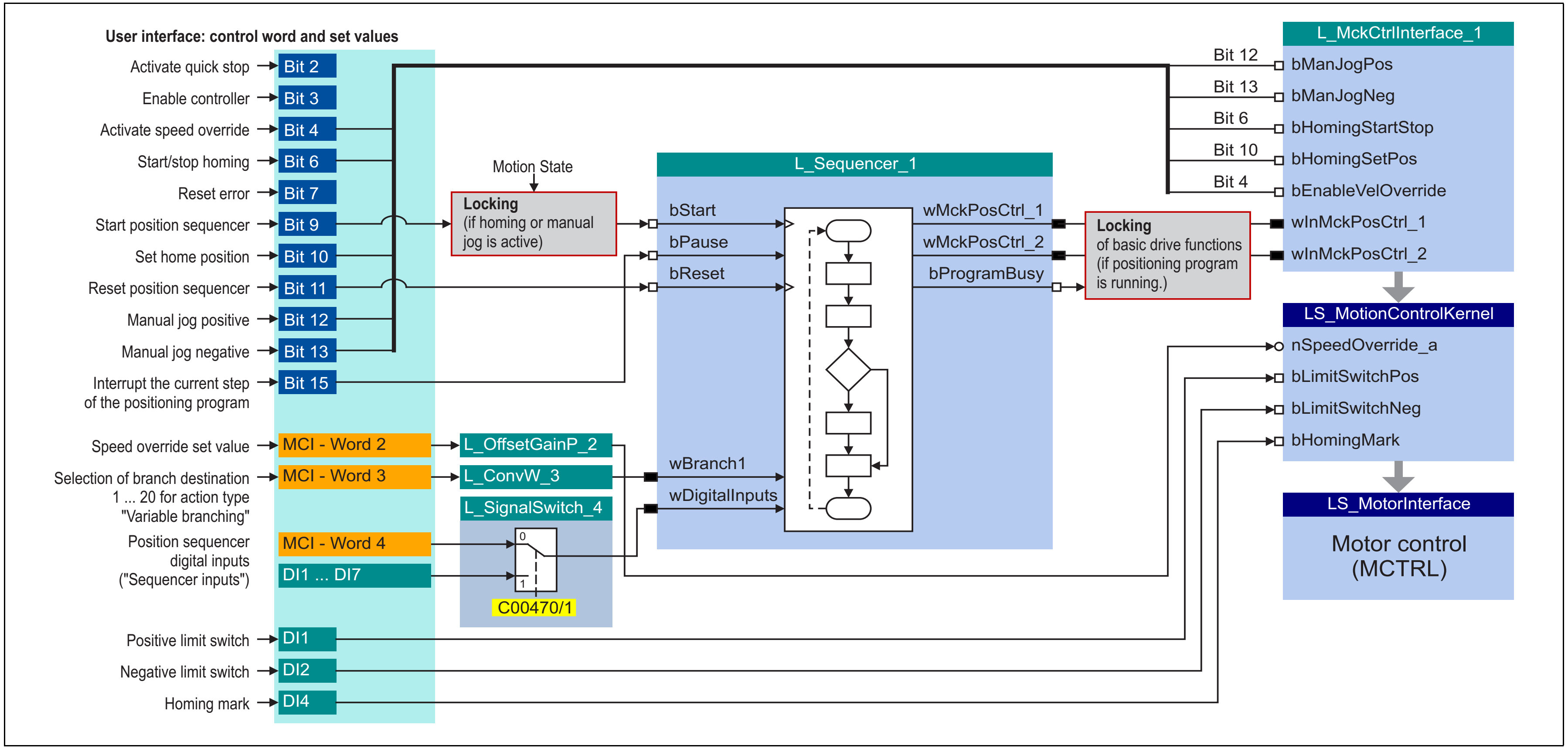
Parameter (Block)	**Possible settings**		**Info**
Limit position monitoring			
C01230 - Bit 3 (LS_MotionControl Kernel)			Consideration of the software limit positions in MCK "Manual jog" operating mode
	0	Software limit positions not active	
	1	**Software limit positions active** (if the home position is known)	
C00595/1...4 (LS_MotionControl Kernel)			Response at the activation of limit position monitoring • Subcode 1: response at the approach of the positive limit switch. • Subcode 2: response at the approach of the negative limit switch. • Subcode 3: response at overtravelling the positive software limit position (C01229/1). • Subcode 4: response at overtravelling the negative software limit position (C01229/2).
	0	No Reaction	
	1	Fault	
	3	**TroubleQuickStop**	
	4	WarningLocked	
	5	Warning	
	6	Information	
Error/status messages of the positioning sequence control			
C00581/1...2 (LS_SetError_1)			Response to errors of the positioning sequence control • Subcode 1: response if the time monitoring function for the "Positioning" action has been activated. • Subcode 2: response if the positioning sequence control reports an error.
	0	No Reaction	
	1	Fault	
	2	Trouble	
	3	**TroubleQuickStop**	
	4	WarningLocked	
	5	Warning	
	6	Information	
C00581/3 (LS_SetError_1)			Response if the positioning program has been started and is pausing.
	0	No Reaction	
	1	Fault	
	2	Trouble	
	3	TroubleQuickStop	
	4	WarningLocked	
	5	Warning	
	6	**Information**	

4 Detailed functions of the technology application

This chapter describes the functions implemented in the "Position Sequencer" technology application with the possible settings relevant for the application.

Detailed information on the function and parameterisation of the functions described in the following can be found in the reference manual/online help of the controller.

4.1 Signal flow of the technology application



4.2 Basic drive functions (MCK)

Danger!

During homing, manual jog, and positioning, specially assigned profile parameters are active. If they have not been set correctly, the drive may carry out an unexpected movement!

Detailed information relating to the basic drive functions and the corresponding parameters can be found in the reference manual/online help of the inverter in the "Basic drive functions (MCK)" chapter.

4.2.1 Homing

In order to be able to actuate the drive in a reasonable fashion, the position of the zero position within the physically possible travel range must be known. The zero position, also called the "reference", can be defined by homing or reference setting. This so-called "homing" is usually carried out after the mains is switched on. Furthermore homing is required after an encoder error has been eliminated.

- The homing is started/stopped manually via bit 6 of the application control word (**L_ConvBitsToWord_2** FB). In the default setting, bit 6 is linked with bit 6 of MCI process data input word 1. ▶ Pre-assignment of the process data input words (📖 18)
- The "100: SetRef" homing mode is preset in C01221. Homing is not executed; in this mode the current position is just accepted as zero point of the measuring system ("reference setting").
- Homing is to be configured according to the requirement of the application, as described in the reference manual/online help of the inverter.
- If the home position is known to the drive, a response is sent via digital output DO3 and bit 15 of the MCI process data output word 1.

Note!

Additional logic operations in the application prevent the basic function "Homing" from being activated if the positioning program is processed.

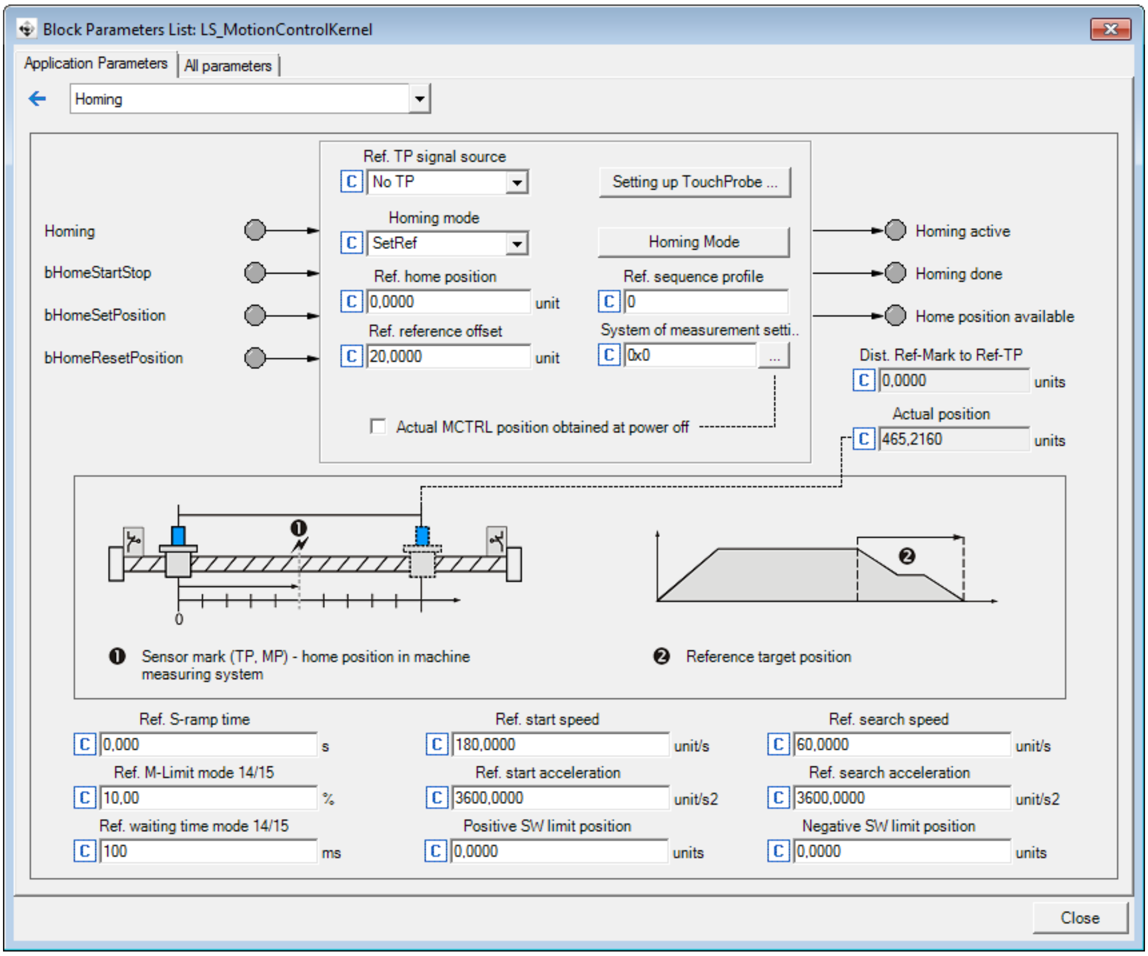
Tip!

In addition to "manual" homing via bit 6, also the possibility of defining homing in the program flow of the positioning sequence control is provided (ideally as the first program step).

See description of the "Homing" action type in the appendix. (📖 43)

How to go to the parameterisation dialog of the basic "Homing" function:

1. Open the parameter list for the **LS_MotionControlKernel** SB.
2. In the *Block Parameters List...* dialog box on the **Application parameters** tab, select the "Homing" entry in the upper list field.



Note!

For a reference search with touch probe detection:

If the reference signal is to follow a real touch probe, the touch probe interface must be configured accordingly via the **Setting up TouchProbe...** button!

Parameter	Info	Lenze setting	
		Value	Unit
C01221	MCK: Homing mode	100: SetRef	
C01224/1	MCK: Ref. initial speed	180.0000	unit/s
C01225/1	MCK: Ref. initial acceleration	3600.0000	unit/s2
C01224/2	MCK: Ref. search speed	60.0000	unit/s
C01225/2	MCK: Ref. search acceleration	3600.0000	unit/s2
C01226/1	MCK: Ref. S-ramp time	0.000	s

Parameter	Info	Lenze setting	
		Value	Unit
C01222	MCK: Ref. M limit mode 14/15	10.00	%
C01223	MCK: Ref. waiting time mode 14/15	100	ms
C01227/1	MCK: Ref. offset reference degree	20.0000	unit
C01227/2	MCK: Ref. home position	0.0000	unit
C01228	MCK: Ref. sequence profile	0	
C01229/1	MCK: Positive SW limit position	0.0000	units
C01229/2	MCK: Negative SW limit position	0.0000	units
C01246/1	MCK: Ref. TP signal source	0: No TP	

4.2.2 Manual jog

In this operating mode, the drive can be traversed manually in a clockwise or anticlockwise direction ("jogging mode").

- The basic function "Manual jog" is activated via bit 12 and bit 13 of the application control word (**L_ConvBitsToWord_2** FB). In the default setting, bit 12 and bit 13 are linked to digital inputs DI5 and DI6.
- An activation via the fieldbus interface (MCI) is possible after a corresponding adaptation of the user interface. See commissioning; Step 2 (optional): Establish control via the fieldbus interface (MCI). (📖 16)

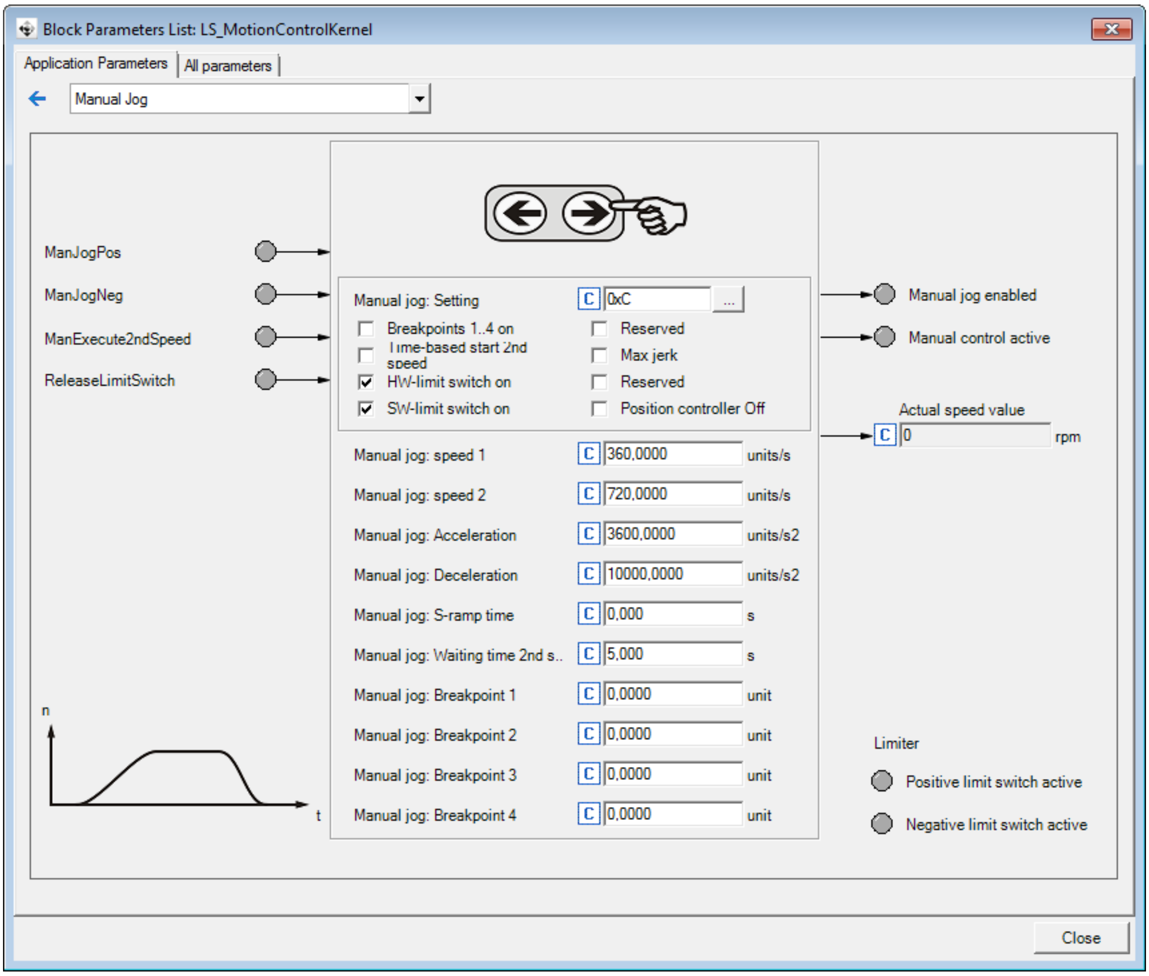
Note!

Additional logic operations in the application prevent the basic function "Manual jog" from being activated if the positioning program is processed.

In the Lenze setting, the software limit positions parameterised are active for manual jog if the home position is known. ▸ Limit position monitoring (📖 39)

How to go to the parameterisation dialog of the basic "Manual jog" function:

1. Open the parameter list for the **LS_MotionControlKernel** SB.
2. In the *Block Parameters List...* dialog box on the **Application parameters** tab, select the "Manual Jog" entry in the upper list field.



Parameter	Info	Lenze setting	
		Value	Unit
C01230	MCK: Manual jog setting	Bit coded	
C01231/1	Manual jog: speed 1	360.0000	units/s
C01231/2	Manual jog: Speed 2	720.0000	units/s
C01232/1	Manual jog: Acceleration	3600.0000	units/s2
C01232/2	Manual jog: Deceleration	10000.0000	units/s2
C01233/1	Manual jog: S-ramp time	0.000	s
C01235/1	Waiting time 2nd speed	5.000	s
C01234/1	Manual jog: Breakpoint 1	0.0000	unit
C01234/2	Manual jog: Breakpoint 2	0.0000	unit
C01234/3	Manual jog: Breakpoint 3	0.0000	unit
C01234/4	Manual jog: Breakpoint 4	0.0000	unit

4.2.3 Positioning

For starting a positioning process, 50 actions of the "Positioning" type are provided for the positioning sequence control. If such an action is active, the basic function "Positioning" is requested by the **L_Sequencer_1** via corresponding control outputs.

- The profile is entered via the parameters of the basic function "Positioning".
- For the 8400 TopLine inverter, 15 different profiles can be parameterised.

A detailed explanation of all profile parameters can be found in the appendix in the description for the "Positioning" action type. (📖 44)

4.2.4 Holding brake control

This basic function is used for low-wear control of a holding brake.

Danger!

Please note that the holding brake is an important element of the safety concept of the entire machine.

Thus, proceed very carefully when commissioning this system part!

Detailed information on how to parameterise the holding brake control can be found in the reference manual/online help of the controller in the chapter "Basic drive functions (MCK)".

The documentation of the holding brake control contains safety instructions which must be observed!

Application-specific notes on the holding brake control:

- In the Lenze setting, the mode 0 (brake control off) is preset in C02580.
- The technology application is prepared for the control of a 24 V holding brake via the high current output (terminal strip X107).
- The application of the holding brake causes controller inhibit, and when the inverter is inhibited, the following error is reset. Starting from version 14.00.00, a following error value can be set in C01215/3 that remains stored even if the controller is inhibited.
- In mode 12 (controlled automatically), the speed thresholds do not apply to the operating modes with a setpoint request via control signal (e.g. "PosExecute" in the "Positioning" operating mode). Here the control logics open and close the holding brake by internal commands in the **Motion Control Kernel**.

4.3 Speed/position output via axis bus for a slave drive

In order to be able to directly connect a slave drive to the positioning drive, the **LS_AxisBusOut** SB is provided in the application.

- In the default setting, the speed setpoint integrated to an angle (path) is output via line data words 1 & 2, and the speed setpoint is output via line data word 3.
- A slave drive connected via axis bus with the "Electrical Shaft Slave" technology application is executable immediately after the initialisation of the axis bus; however, without the bus monitoring functions of the electrical shaft being available.
- If the drive is to be used as master for a slave drive with the "Position Follower" technology application, set the value "1" in C00470/4. Then, instead of the speed integrated, the current position is output via axis bus.

STOP **Stop!**

If operation as master position follower is switched on (C00470/4 = "1"), homing of the master causes a step for the slave position follower if the current position is set to the home position!

Remedy: Inhibit the slave position follower when you are referencing the master!

The signal to be output via axis bus is selected using C00470/2 and C00470/4:

C00470/4		**C00470/2**		**Axis bus output**
Operation as position follower master		Setpoint/actual value selection		
0	**Off**	**0**	**Setpoint**	Line data words 1 & 2 = speed setpoint integrated to an angle (path) Line data word 3 = speed setpoint
		1	Actual value	Line data words 1 & 2 = current speed integrated to an angle (path) Line data word 3 = current speed
1	On			Line data words 1 & 2 = current position Line data word 3 = current speed

Details on the signal flow

The changeover between the setpoint and actual value is effected via the **L_SignalSwitch_2** FB.

In the default setting C00470/4 = 0, the position value for SB **LS_AxisBusOut** is generated with the **L_PhaseIntK_2** FB. Loading or a reset is not required, since in the slave drive merely consistency with the speed value is compared without attaching importance to the absolute value.

With the setting C00470/4 = 1 (operation as position follower master), as position value for SB **LS_AxisBusOut** the current position *dnMotorPosAct_p* is used by the **LS_MotorInterface** SB.

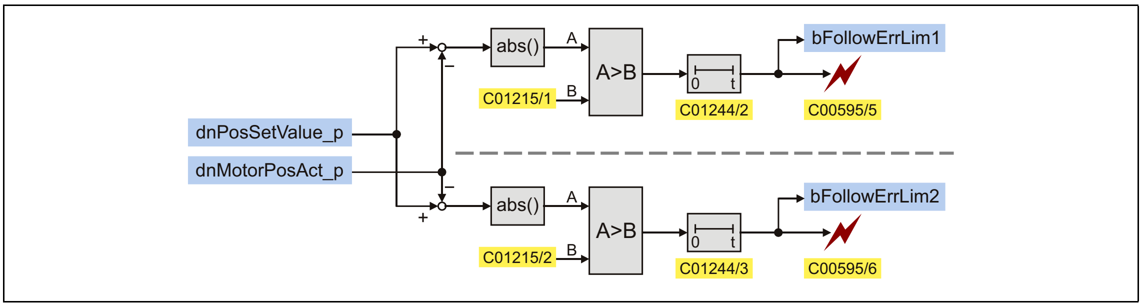
4.4 Monitoring functions

4.4.1 Following error monitoring system

The difference between set position and actual position is called the following error. Ideally, the following error should be "0". The set position is created by the internal definition of the traversing profiles of the **Motion Control Kernel**. The actual position is created by the integration of the speed supplied by the position encoder. The following error is always compensated dynamically. With an optimum setting of the position control, only a minimum following error occurs which does not increase continuously.

Certain processes, however, require that a defined limit as a difference between set position and actual position is not exceeded. If it is exceeded, it may have been caused by a mechanical blocking in the machine and the system part is not situated at the position defined at that time. In such a case, it makes sense to activate the "Fault" error response to make the motor torqueless.

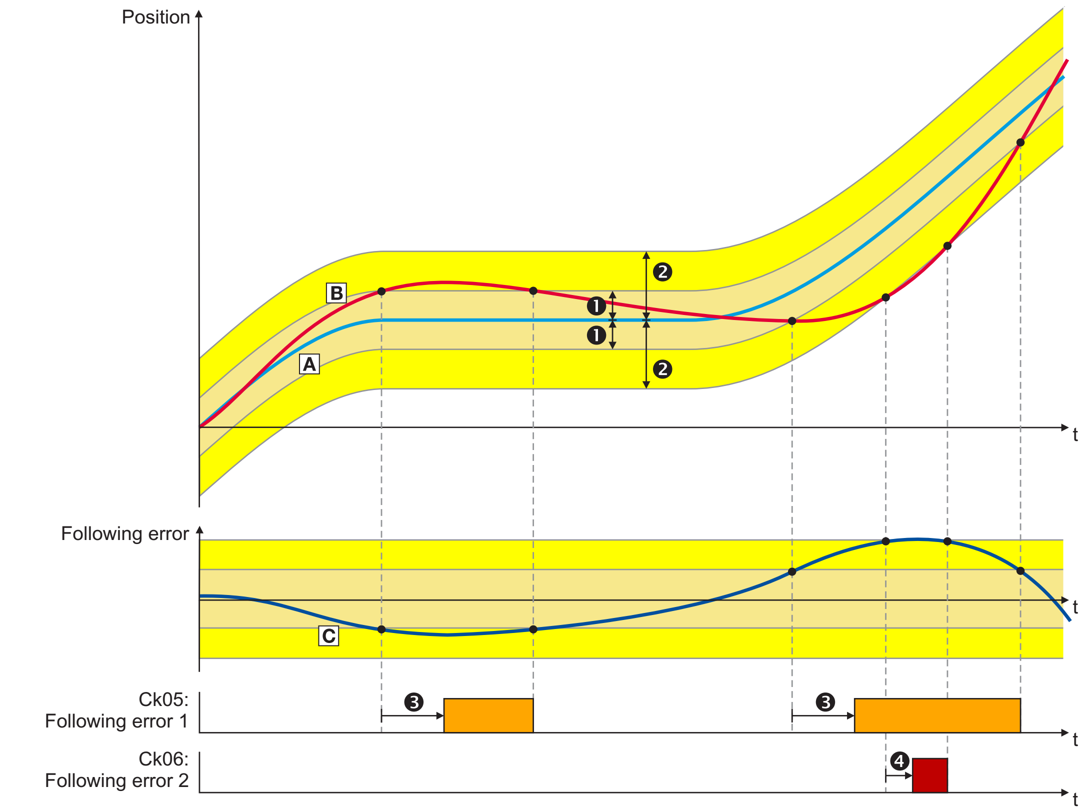
In the 8400 TopLine controller, two independent following error monitoring systems can be parameterised:



[4-1] Two-channel following error monitoring system

Parameter (Block)	Possible settings			Info
C01215/1 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 1 • The setting "0" deactivates following error monitoring 1
	Lenze setting: 5.0000 units			
C01215/2 (LS_MotionControl Kernel)	0.0001	units	214748.3647	Limit for following error monitoring 2 • The setting "0" deactivates following error monitoring 2
	Lenze setting: 10.0000 units			
C01244/2...3 (LS_MotionControl Kernel)	0	ms	60000	Waiting time for following error monitoring 1 & 2 • In order to avoid that an error is triggered by acceleration and a narrow tolerance limit can be nevertheless monitored at standstill in the target, the response of the following error monitoring system can be delayed by setting a waiting time.
	Lenze setting: 0 ms			
C00595/5...6 (LS_SetError_1)				Response at the activation of following error monitoring • Subcode 5: response at the activation of following error monitoring 1. • Subcode 6: response at the activation of following error monitoring 2.
	0	No Reaction		
	1	Fault		
	3	TroubleQuickStop		
	4	WarningLocked		
	5	**Warning**		
	6	Information		

Example of the evaluation of the following error



A	Target position	❶	Limit for following error monitoring 1 (C01215/1)
B	Position actual value	❷	Limit for following error monitoring 2 (C01215/2)
C	Current following error	❸	Waiting time for following error monitoring 1 (C01244/2)
		❹	Waiting time for following error monitoring 2 (C01244/3)

The above example clearly shows that there are two monitoring functions acting independently of each other. Normally, one following error monitoring function with a low tolerance is set as warning, and the other with a higher tolerance is set as TroubleQuickStop.

By means of the waiting times (C1244/2 and C1244/3), the error responses can be delayed, so that disconnection does not result immediately when the following error limits are temporarily exceeded. This "switch-on delay" makes it possible to effectively distinguish between dynamic processes and actual error states such as mechanical obstacles or sluggishness.

4.4.2 Limit position monitoring

For safety reasons, drives with a limited traversing range must always be made safe using corresponding safety mechanisms. These are firstly the hardware limit switches, which must effect a standstill of the axes as quickly as possible during the approaching process. In addition, the parameterisable software limit positions are to inhibit all travel commands which would entail an exit of the permissible travel range.

Take the following items into consideration for placing the limit positions:

- The hardware limit switches must be mounted with a sufficient clearance in front of the mechanical limit positions of the linear axis. The clearance to the mechanical limit position should be calculated for a quick stop (QSP) from maximum speed, so that in the event of an error, standstill is attained safely before the mechanical limit position is reached.
- The software limit positions must be placed with a sufficient distance from the hardware limit switch positions. The distance should be calculated for a quick stop (QSP) from maximum speed without the hardware limit switches being approached.

4.4.2.1 Hardware limit switch

In the default setting, the two digital inputs DI1 and DI2 are provided for the connection of the hardware limit switches.

- The connection is configured in a fail-safe fashion (LOW = limit switch activated).
- The two digital inputs are linked with the *bLimitSwitchPos* and *bLimitSwitchNeg* monitoring inputs of the **LS_MotionControlKernel** system block.

Stop!

The limit switches are only evaluated if the limit switches for the respective operating mode have been activated (see the following table)!

Operating mode	Hardware limit switch effective
Homing	Depending on the homing mode selected (see description of the homing modes in the reference manual/online help of the inverter)
Manual jog	Yes (adjustable in C01230 - bit 2)
Positioning	Yes

Parameter (Block)	Possible settings		Info
C00595/1...2 (LS_MotionControl Kernel)			Response at the activation of limit position monitoring • Subcode 1: response at the approach of the positive limit switch. • Subcode 2: response at the approach of the negative limit switch.
	0	No Reaction	
	1	Fault	
	3	**TroubleQuickStop**	
	4	WarningLocked	
	5	Warning	
	6	Information	

Behaviour when hardware limit switches are active

If one of the two monitoring inputs is set to TRUE, in the Lenze setting the "TroubleQuickStop" error response is triggered: Irrespective of the setpoint selection, the drive is brought to a standstill in the deceleration time set for the quick stop function. Depending on the error response parameterised, the drive can then only be traversed again after the error has been acknowledged.

4.4.2.2 Software limit positions

The parameterisable limit positions are used by the software to limit the traversing range.

Stop!

The software limit positions are only evaluated and monitored if the home position is known to the drive and the software limit positions for the respective operating mode have been activated (see following table)!

If the drive is stopped at a high deceleration, depending on the mass inertia and friction, an overvoltage in the DC bus may occur, and pulse inhibit is set as error response. The use of a brake resistor can prevent this response.

Operating mode	Software limit positions active (if home position is known)
Homing	Yes
Manual jog	Yes (adjustable in C01230 - bit 3)
Positioning	Yes

<table>
<tr><th>Parameter
(Block)</th><th colspan="3">Possible settings</th><th>Info</th></tr>
<tr><td rowspan="2">C01229/1...2
(LS_MotionControl Kernel)</td><td>-214748.3647</td><td>units</td><td>214748.3647</td><td rowspan="2">Positive and negative software limit position for limiting the valid traversing range
• The positive software limit position must be set to a greater value than the negative software limit position!</td></tr>
<tr><td colspan="3">Lenze setting: 0.0000 units</td></tr>
<tr><td rowspan="7">C00595/3...4
(LS_MotionControl Kernel)</td><td colspan="3"></td><td rowspan="7">Response at the activation of limit position monitoring
• Subcode 3: response at overtravelling the positive software limit position (C01229/1).
• Subcode 4: response at overtravelling the negative software limit position (C01229/2).</td></tr>
<tr><td>0</td><td colspan="2">No Reaction</td></tr>
<tr><td>1</td><td colspan="2">Fault</td></tr>
<tr><td>3</td><td colspan="2">TroubleQuickStop</td></tr>
<tr><td>4</td><td colspan="2">WarningLocked</td></tr>
<tr><td>5</td><td colspan="2">Warning</td></tr>
<tr><td>6</td><td colspan="2">Information</td></tr>
</table>

Behaviour in the case of active software limit positions

Note!

The "travel commands" mentioned in the following description are no speed setpoint selections. In the "Speed follower" and "Position follower" operating modes, an acknowledged software limit position error ensures that traversing to the impermissible travel range remains possible afterwards. This is because in these two operating modes, there is no preview of whether a software limit position is approached with a setpoint selection.

If the software limit positions are active, travelling commands that would result in the exit from the permissible travel range can no longer be executed.

In positioning operation, the warning "Ck14: target position outside SW limit position" is output for target positions that are outside of the software limit positions. The positioning process is not aborted, but is executed to the software limit position instead of to the target position outside the software limit position.

If the drive is already outside the permissible travel range and the software limit positions have been activated, only travel commands that result in the drive moving back into the permissible travel range can be executed.

If the software limit positions are active, and one of the software limit positions is overtravelled, in the Lenze setting the "TroubleQuickStop" error response is triggered: Irrespective of the setpoint selection, the drive is brought to a standstill in the deceleration time set for the quick stop function. Depending on the error response parameterised, the drive can then only be traversed again after the error has been acknowledged.

4.4.3 Error and status messages of the positioning sequence control

The application provides the **LS_SetError_1** SB for the purpose of error handling.

- The application can trip up to four different user error messages with parameterisable error IDs and error responses via the four boolean inputs of the **LS_SetError_1** SB.
- In the "Position Sequencer" technology application, the inputs of the **LS_SetError_1** SB are linked with error/status outputs of the **L_Sequencer_1** FB.
- You can gather the application-specific meaning of the user errors as well as the preset response from the following table:

Error message	Meaning	Response (Lenze setting)	can be set in
User error 1	Time monitoring for "positioning" action has been triggered.	TroubleQuickStop	C00581/1
User error 2	Positioning sequence control reports an error.	TroubleQuickStop	C00581/2
User error 3	Positioning program started, break active.	Information	C00581/3
User error 4	- (not assigned, can be used freely)	No Reaction	C00581/4

5 Appendix: Action types for the positioning sequence control

For the positioning sequence control, the **L_Sequencer_1** FB is used. This FB processes a positioning program on the basis of a sequence table (also referred to as "sequencer"), which can contain up to 100 references to so-called "Actions".

Overview

Action type	Number of actions available	Info	
Homing	1	Start of homing	These action types are "active". When these action types are processed, the respective basic function is triggered via corresponding control outputs.
Positioning	50	Execution of a profile	
Branch	16	Conditional or unconditional branch (jump)	These action types are "passive". When these action types are processed, the "Stop" operating mode are active, and a brake will be applied, if available. * The "Standby" action type is active when a setpoint has been set for the speed follower or a setpoint position is transferred to the **LS_MotionControlKernel** SB when "Position follower" has been selected.
Variable branch	2	Variable branch as a function of the input value of *wBranch1* or *wBranch2*.	
Switch	16	Switching of digital output signals	
Set counter	5	Setting one of the 5 counters available to a specific starting value	
Count	8	Counting processes including comparison operation	
Wait	8	Entering waiting times into the program flow	
Standby*	5	Temporary activation of a setpoint follower	
End	1	Determination of the program end	

You can find detailed information on the action types in the following subchapters.

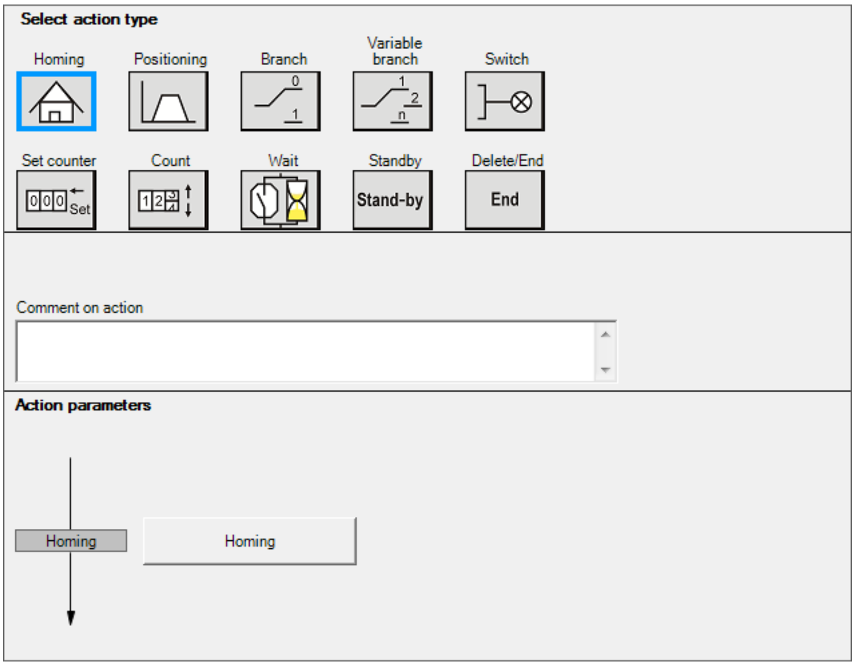
Tip!

For users switching from 9300 to 8400:

Whereas with the Servo Drive 9300 it was usual to query several digital inputs in succession to then obtain a specific positioning process, for 8400 and 9400 the "Variable branch" action can be used.

5.1 Homing

For starting a homing process, the "Homing" action type is provided. When homing has been completed (*bHomingDone* = TRUE), the program flow is continued with the next step.

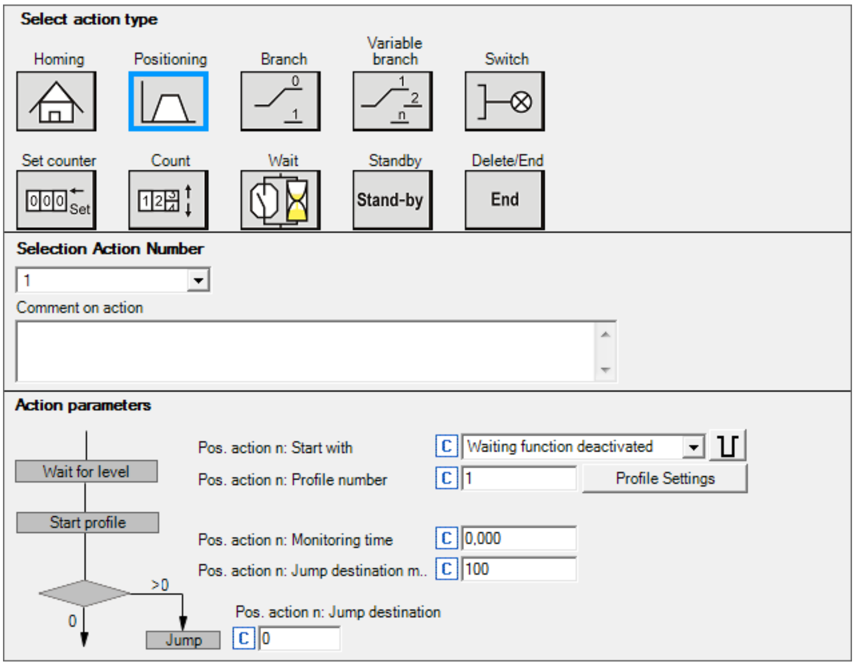


Note!

The "Homing" action has no individual parameters. The settings for homing (e.g. homing mode) are carried out via the parameters of the basic function "Homing". Click the **Homing** button to navigate to the corresponding parameterisation dialog.

5.2 Positioning

For starting a positioning process, 50 actions of the "Positioning" type are provided.

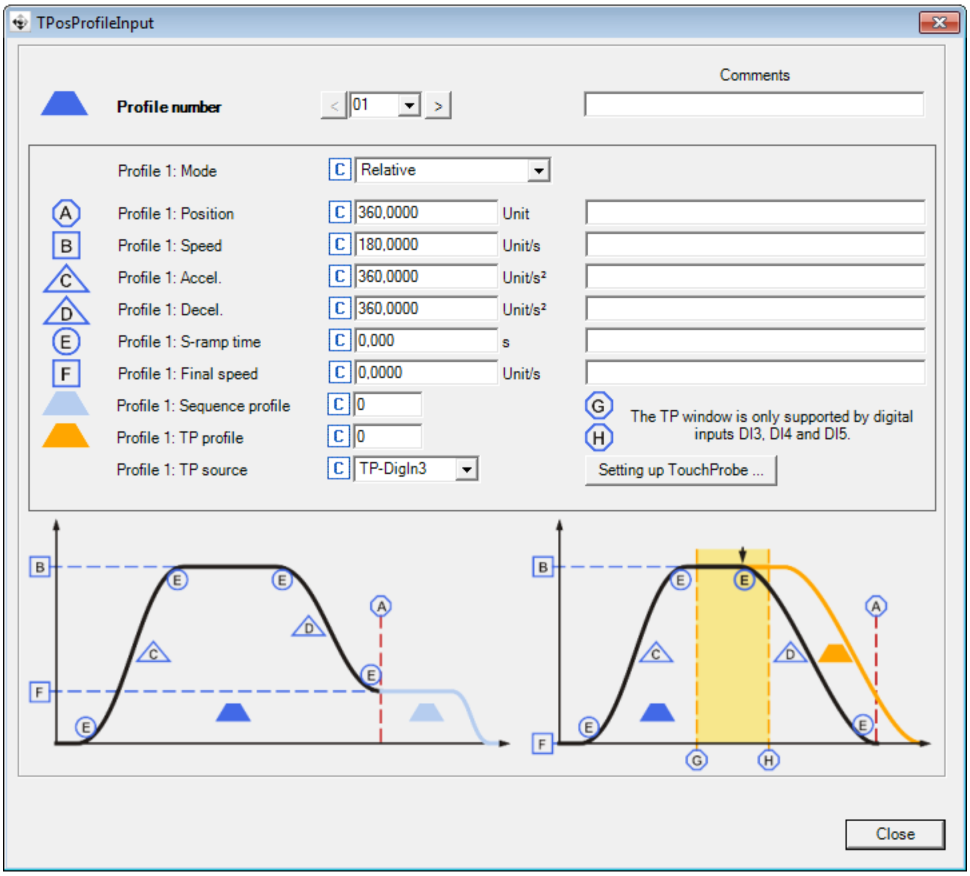


> **Note!**
>
> Although 50 "Positioning" action types are provided, only 15 different profiles can be parameterised.
>
> The profile is entered via the parameters of the basic function "Positioning". Click the **Profile settings** button to navigate to the corresponding parameterisation dialog (see "Profile settings" section).

Parameter	Possible settings		Info
Start with (C01405/1...50)	**0**	**Waiting function deactivated**	In the default setting, execution of the profile is started immediately.
	1	Sequencer input 1 (Bit 0 of *wDigitalInputs*)	"Wait for level": Execution of the profile is only started when the sequencer input selected has the polarity set.
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)	
	...	...	
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)	
Polarity of input (C01406/1...50)	⊔	**Condition is bit state "0"**	State which the sequencer input selected for the profile start must have.
	⊓	Condition is bit state "1"	
Profile number (C01407/1...50)	0	No profile executed	Selection of the profile which is to be executed. A sequence profile can be set in the corresponding profile parameter.
	1	**Execute profile No. 1.**	
	2...15	Execute profiles No. 2 ... 15	

Parameter	Possible settings			Info
Monitoring time (C01409/1...50)	0.000	s	2147480.000	If the positioning process takes longer than the monitoring time set, user error 1 is set. The response to this error can be set in C00581/1 (Lenze setting: "TroubleQuickStop").
	When "0.000 s" is set (Lenze setting), the time monitoring function is deactivated.			
Jump destination monit. (C01410/1...50)	0	**Sequence step**		Step which is executed after the monitoring time has been exceeded.
	1...100	Step 1 ... 100		
Jump destination (C01408/1...50)	0	**Sequence step**		Step within the sequence table which is processed after the profile has been executed.
	1...100	Step 1 ... 100		

Profile settings

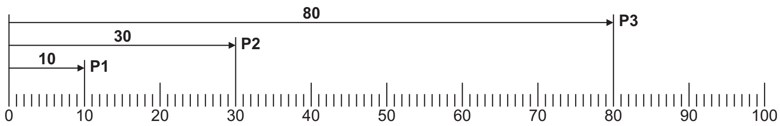


A profile is described by the following profile parameters:

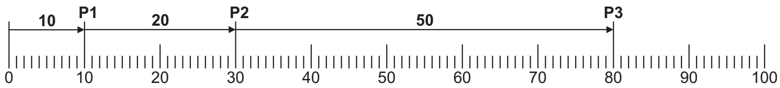
Icon	Profile parameter
	**(Standard) profile** Profile data set (profile numbers 1 ... 15), in which the profile data are stored.
	Mode (C01300/1...15) Selection of the way in which positioning is to be carried out.

<table>
<tr><th>Icon</th><th colspan="2">Profile parameter</th></tr>
<tr><td>A</td><td colspan="2">

Position (C01301/1...15)
Target position or distance to be traversed.
When the position is indicated, a distinction is made between absolute position and relative position:
- An absolute position always indicates the distance to the defined zero position:
  Absolute position = Target position



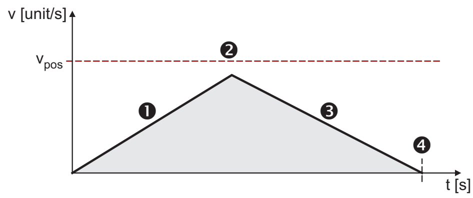
- A relative position indicates the distance to the starting position (current position):
  Relative position = Target position - Starting position



</td></tr>
<tr><td rowspan="2">B</td><td colspan="2">

Speed (C01302/1...15)
Maximum speed during the positioning process.
- Depending on the profile parameters of position, acceleration and deceleration , it is possible that the drive will not even reach the maximum speed. In this case, the graphic representation will be a trapezium instead of a triangle:

</td></tr>
<tr><td>



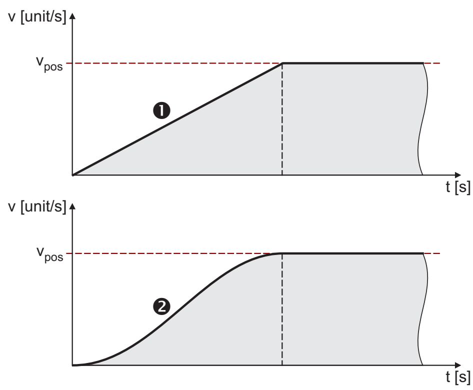
</td><td>

❶ Acceleration
❷ Travelling speed (is not reached in this case)
❸ Deceleration
❹ Target position (or traversing distance)

</td></tr>
<tr><td rowspan="2">C</td><td colspan="2">

Acceleration (C01303/1...15)
Maximum acceleration during the positioning process.
- Two types of acceleration are distinguished:
  - Constant acceleration: the speed increases linearly.
  - Linearly increasing acceleration: Speed increases in S-shape.
    A linearly increasing acceleration (S-profile) results from the setting of an S-ramp time (see more below).

</td></tr>
<tr><td>



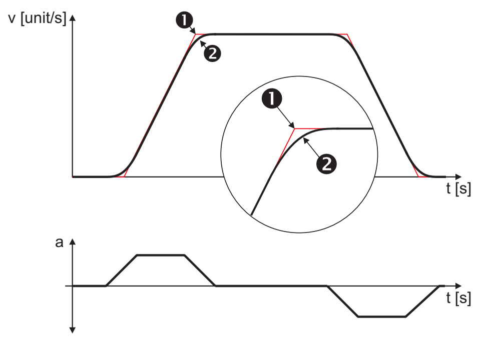
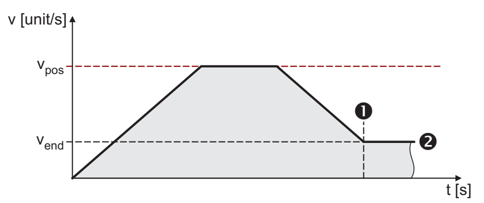
</td><td>

❶ Constant acceleration (L-profile)
❷ Linearly increasing acceleration (S-profile)

</td></tr>
<tr><td>D</td><td colspan="2">

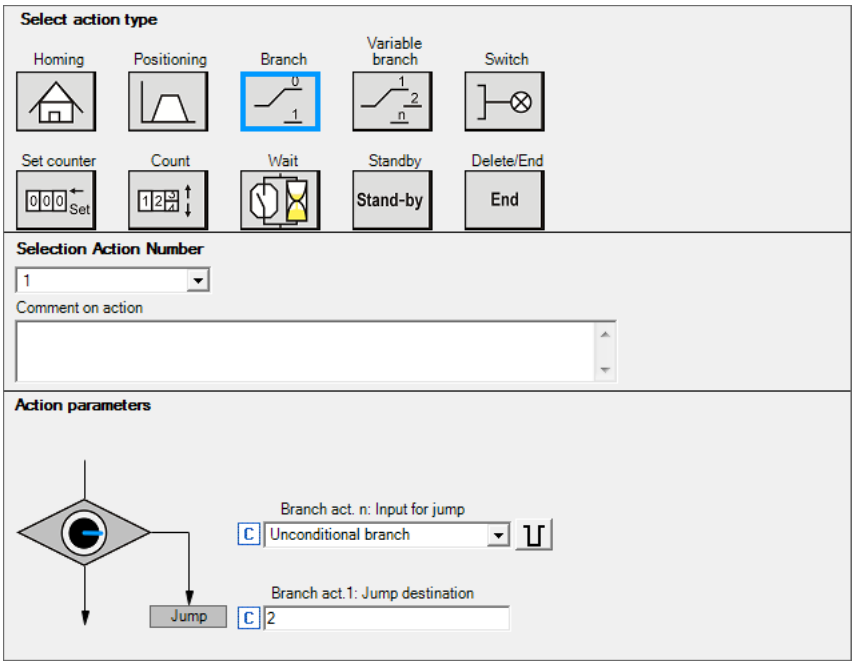
Deceleration (C01304/1...15)
Maximum deceleration during the positioning process.

</td></tr>
</table>

Icon	Profile parameter	
E	**S-ramp time** (C01305/1...15) Due to stipulation of an S-ramp time for a profile, the profile is executed with S-shaped ramps, i.e. acceleration and braking processes are initiated smoothly in order to reduce jerk and thus the stress on the drive components. • The acceleration/deceleration stipulated in the profile is not achieved until after the specified S-ramp time. • This kind of acceleration/deceleration is needed for sensitive machine parts with a certain amount of play. • The unavoidable consequence of the slower increase in acceleration in the case of the S profile is that the positioning time is longer compared to the L profile, which is more efficient in terms of time.	
		❶ Without jerk limitation (L profile) ❷ With jerk limitation (S profile)
F	**Final speed** (C01305/1...15) This specifies the speed at which the drive is to start the next profile after reaching the target position. With a final speed not equal to "0", "velocity changeover" or "overchange" is possible, i.e. when the target position is reached, a second positioning process is started immediately without the drive coming to a standstill at the first target position.	
		❶ Target position ❷ Final speed (in this case, not equal to "0")
	Sequence profile (C01307/1...15) **for profile linkage/subsequent block control** A special feature is the automatic advancing to sequence profiles with and without velocity changeover. For this purpose, the profile number of the desired sequence profile (1 ... 15) is simply set in the "Sequence profile" profile parameter. After execution of the profile (target position reached), the set following (subsequent) profile is started automatically. In this way, profile chains can be stipulated without additional control processes. • If the "Final speed" profile parameter is set to <> "0", a velocity changeover to the sequence profile takes place at the final speed set. • If "0" is set for the following (i.e. subsequent) profile, profile linkage does not take place. • This function can be performed in all positioning modes.	
	TP profile (C01308/1...15) Profile number of the profile (1 ... 15) that is to be executed after a touch probe has been detected. • If "0" is set, there will be no profile stepping through touch probe. • Only relevant for positioning modes with touch-probe.	
	TP signal source (C01309/1...15) Selection of the signal source for touch probe detection. • Only relevant for positioning modes with touch-probe.	

5.3 Branching

For conditional and unconditional branching (jumps), 16 actions of "Branching" Type are available.

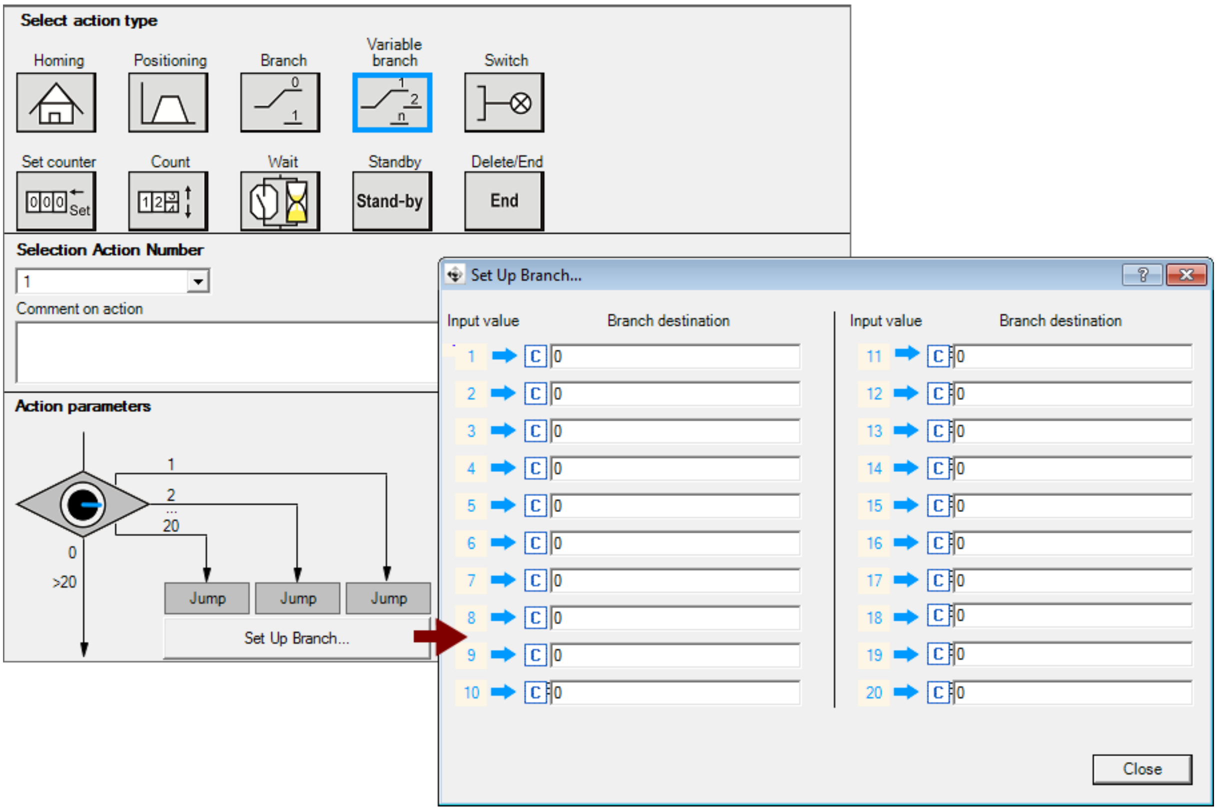


Parameter	Possible settings		Info
Input f. jump (C01415/1...16)	**0**	**Unconditional branch**	"Unconditional branch": There is always a branch to the jump destination set.
	1	Sequencer input 1 (Bit 0 of *wDigitalInputs*)	"Conditional branch": There only is a branch to the jump destination set if the sequencer input selected has the polarity set. Otherwise the next step in the sequence table is processed.
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)	
	...	...	
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)	
Polarity of input (C01416/1...16)	⊔	**Condition is bit state "0"**	State which the sequencer input selected must have for a conditional branch.
	⊓	Condition is bit state "1"	
Jump destination (C01417/1...16)	**0**	**Sequence step**	For a conditional branch, there only is a branch to the jump destination set here if the sequencer input selected has the polarity set. Otherwise the next step in the sequence table is processed.
	1...100	Step 1 ... 100	

5.4 Variable branch

For variable branching (jumps), two actions of "variable branching" type are available.

- Branching to one of 20 possible branching destinations is carried out as a function of the input value of *wBranch1* or *wBranch2* at the time of processing.
  - The input value of *wBranch1* defines branching for action 1.
  - The input value of *wBranch2* defines branching for action 2.
  - If the input value is 0 or greater 20, the next step in the sequence table is processed.
- In the default setting, the *wBranch1* input is linked with MCI process data input word 3. ▶ Pre-assignment of the process data input words (📖 18)
- In the default setting, the *wBranch2* input is not connected.

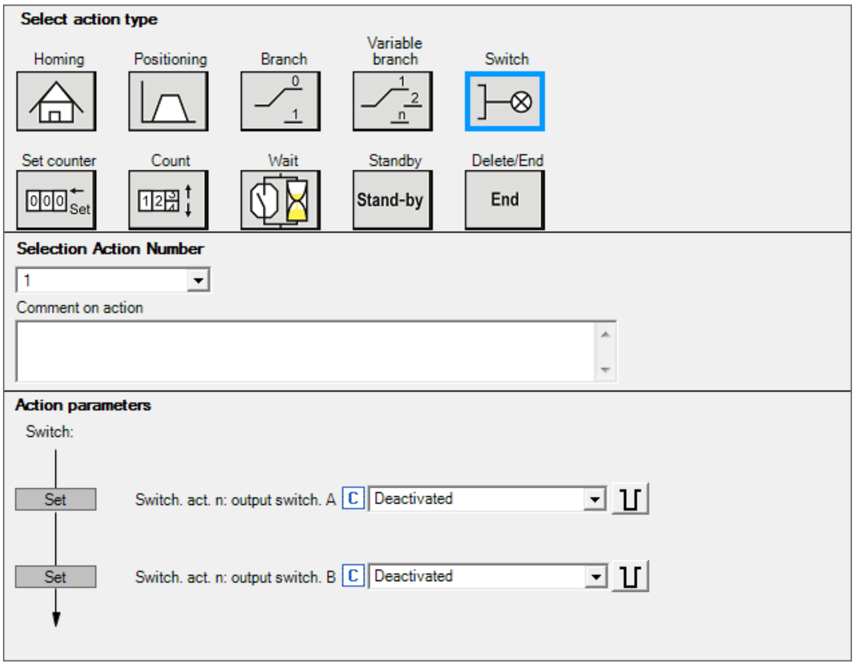


For entering the branch destinations, click the **Set Up Branch...** button in the parameterisation dialog.

Parameter	Possible settings		Info
Branch destination (C01418/1...2 - C01437/1...2)	0	**Branching deactivated** (The following step in the sequence table is processed.)	Branch destinations for input values 1 ... 20 of *wBranch1...2*.
	1...100	Step 1 ... 100	

5.5 Switch

For switching digital output signals, 16 actions of the "Switch" type are provided. Each action can set two bits of the *wDigitalOutputs* output signal, which can be selected, to "0" or "1", alternatively and independently of each other.

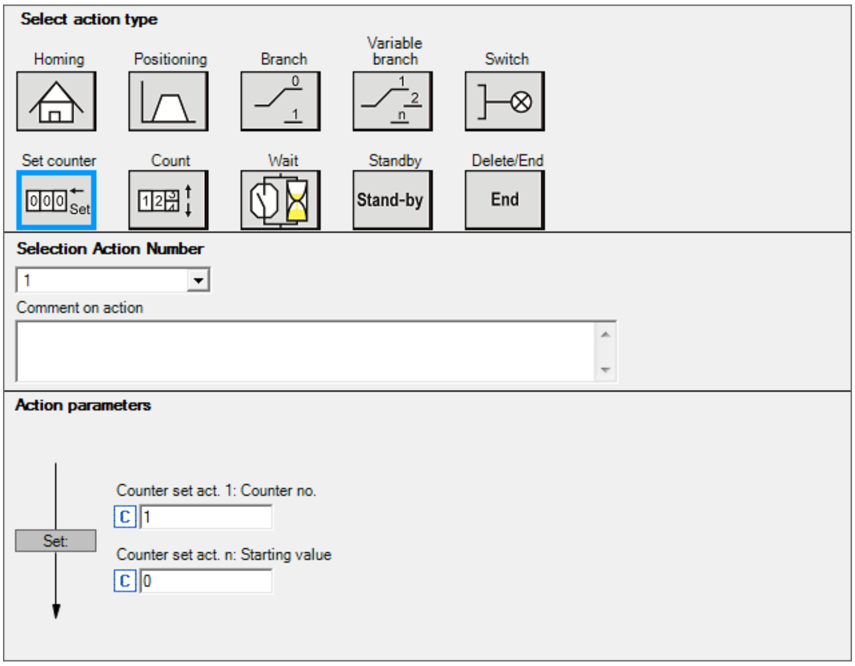


Parameter	Possible settings		Info
Output switch. A (C01411/1...16)	**0**	**Deactivated**	Selection of the sequencer output which is to be set to the set polarity by this action.
	1	Sequencer output 1 (Bit 0 of *wDigitalOutputs*)	
	2	Sequencer output 2 (Bit 1 of *wDigitalOutputs*)	
	...	...	
	16	Sequencer output 16 (Bit 15 of *wDigitalOutputs*)	
Pol. switch. A (C01412/1...16)		**Set output bit to "0"**	State to which the sequencer output is to be set.
		Set output bit to "1"	
Output switch. B (C01413/1...16)	**0**	**Deactivated**	Selection of the sequencer output which is to be set to the set polarity by this action.
	1	Sequencer output 1 (Bit 0 of *wDigitalOutputs*)	
	2	Sequencer output 2 (Bit 1 of *wDigitalOutputs*)	
	...	...	
	16	Sequencer output 16 (Bit 15 of *wDigitalOutputs*)	
Pol. switch. B (C01414/1...16)		**Set output bit to "0"**	State to which the sequencer output is to be set.
		Set output bit to "1"	

5.6 Counter setting

For setting one of the 5 available counters to a certain starting value, 5 actions of "Counter setting" type are available.

- The 5 actions of "counter setting" type are <u>not</u> permanently assigned to the 5 counters.
- You can, e.g., set a counter to a value using an action of "counter setting" type and at a later program time set the same counter to a different value using another action of "counter setting" type.



Tip!

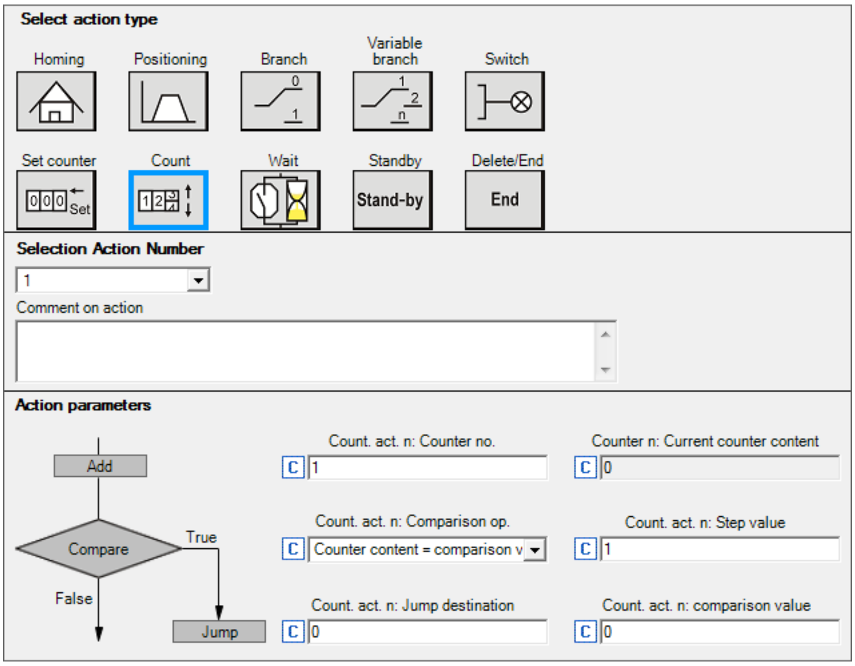
Since for the "Count" action also negative step values can be set, it is also possible to count down from a starting value that is set.

Parameter	Possible settings		Info
Counter No. (C01441/1...5)	0	Counter setting deactivated	Selection of the counter
	1	**Counter 1**	
	...	...	
	5	Counter 5	
Starting value (C01442/1...5)	-2147483647	2147483647	Value to which the counter selected is to be set.
	Initialisation: 0		

5.7 Count

For counting processes, 8 actions of the "Count" type are provided. Each time the action is processed, the counter content of the corresponding counter is increased or decreased by the step value set (counting up or down).

- The comparison operation enables a conditional branch depending on the current counter content.
- For setting a counter to a starting value, 5 actions of "counter setting" type are available.



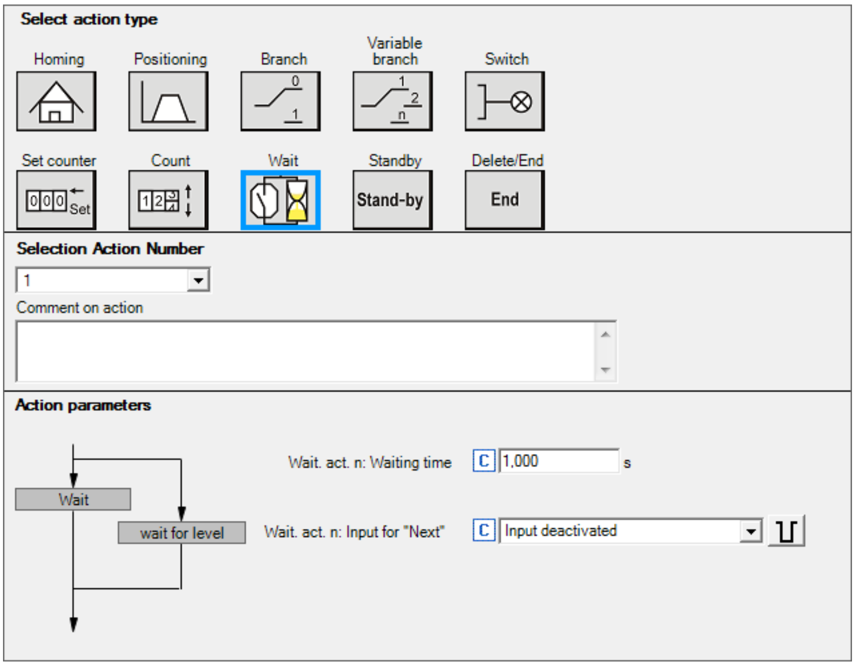
Parameter	Possible settings		Info
Counter No. (C01444/1...5)	0	Counting deactivated (Sequence step is processed.)	Selection of the counter
	1	**Counter 1**	
	...	...	
	5	Counter 5	
Comparison operation (C01448/1...8)	**1**	**Counter content = comparison value**	If the comparison operation is true, a branch to the jump destination set is executed. Otherwise the next step in the sequence table is processed.
	2	Counter content > comparison value	
	3	Counter content ≥ comparison value	
	4	Counter content < comparison value	
	5	Counter content ≤ comparison value	
Jump destination (C01447/1...8)	**0**	**Sequence step**	Jump destination if the comparison operation is true.
	1...100	Step 1 ... 100	
Current counter content (C01443/1...5)	-2147483647	2147483647	Read only
Step value (C01445/1...8)	-2147483647	2147483647	Value by which the counter is increased or decreased.
	Initialisation: 1		
Comparison value (C01446/1...8)	-2147483647	2147483647	Value to which the counter is compared.
	Initialisation: 0		

5.8 Wait

For integrating waiting times into the program flow, 8 actions of the "Wait" type are provided.

The sequence step is only processed

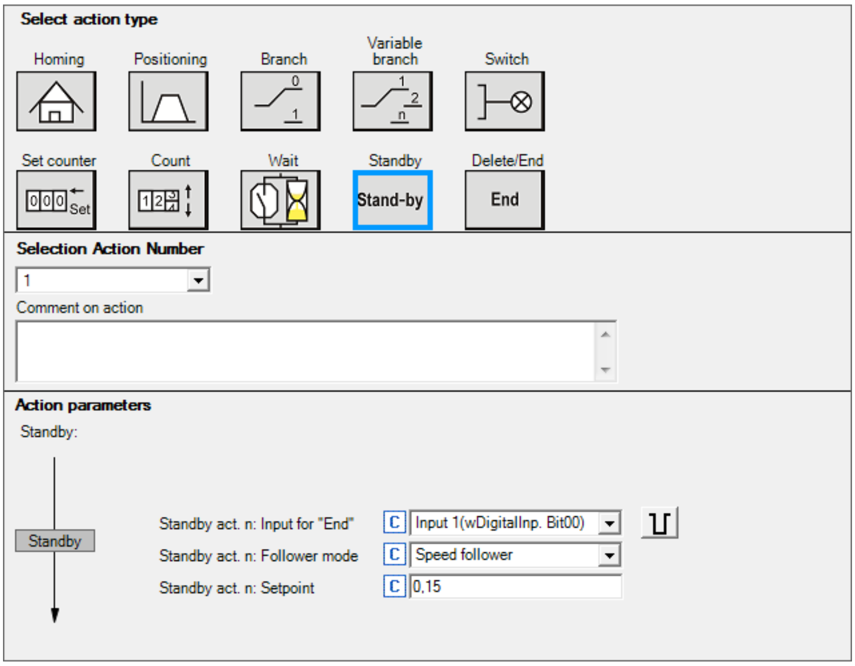
- after a waiting time has elapsed,
  OR
- when a selectable sequencer input has a specific level.



Parameter	Possible settings			Info
Waiting time (C01438/1...8)	0.000	s	2127480.000	Waiting time for "Wait" function • The setting "0" deactivates the waiting time.
	Initialisation: 1.000 s			
Input for "Next" (C01439/1...8)	**0**	**Input deactivated**		"Wait": Sequence step is only processed after the waiting time set has elapsed.
	1	Sequencer input 1 (Bit 0 of *wDigitalInputs*)		"Wait for level": Sequence step is only processed when the sequencer input selected has the polarity set. **Note!** However, if a waiting time > 0 s is set, the sequence step is processed at the latest after the waiting time set has elapsed.
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)		
	...	...		
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)		
Polarity of input (C01440/1...16)	⊔	**Condition is bit state "0"**		State which the sequencer input selected must have so that the sequence step is processed.
	⊓	Condition is bit state "1"		

5.9 Standby

For temporary activation of a speed or position follower, 5 actions of the "Standby" type are provided.

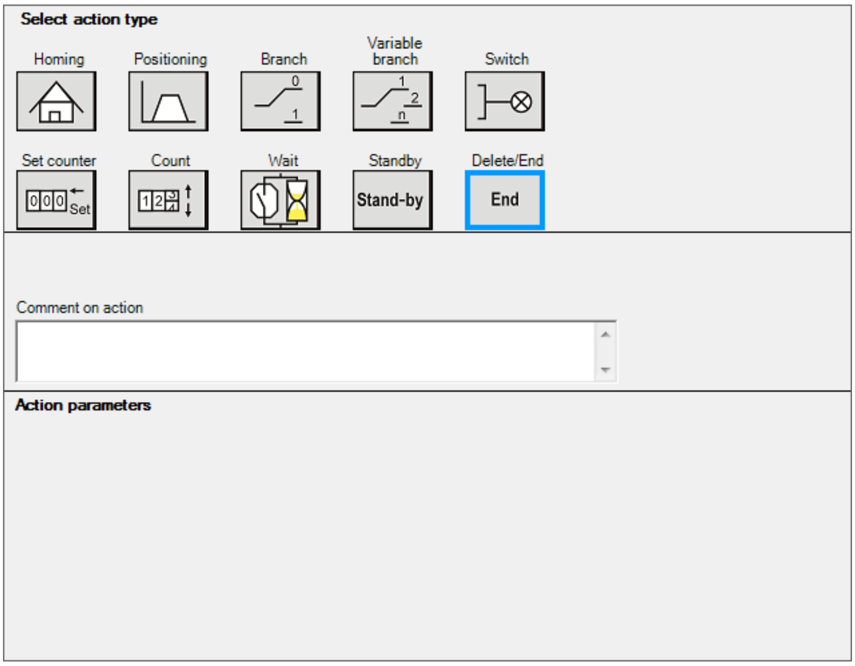


Parameter	Possible settings		Info
Input f. "End" (C01449/1...5)	**1**	**Sequencer input 1** (Bit 0 of *wDigitalInputs*)	The follower mode selected remains enabled until the sequencer input selected here has the polarity set.
	2	Sequencer input 2 (Bit 1 of *wDigitalInputs*)	
	...	...	
	16	Sequencer input 16 (Bit 15 of *wDigitalInputs*)	
Polarity of input (C01450/1...5)	⊔	**Condition is bit state "0"**	State which the sequencer input selected must have so that "Standby" is exited and the sequence step is processed.
	⊓	Condition is bit state "1"	
Follower mode (C01451/1...5)	**0**	**Speed follower**	Selection of the follower that is to be activated in standby mode.
	1	Position follower	
Setpoint (C01452/1...5)	-200	% 200	Setpoint for the speed follower (displayed at *nSet_a* output).
	Initialisation: 0.15 %		

Detailed information relating to the basic drive functions "Speed follower" and "Position follower" as well as to the corresponding parameters can be found in the reference manual/online help of the inverter in the "Basic drive functions (MCK)" chapter.

5.10 End

In order to define the program end within the sequence table, one action of the "End" type is provided. If, at the **L_Sequencer** FB, the *bStart* input is reset to FALSE while the positioning program is running, processing is only continued until the program end.



The "End" action has no individual parameters.

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

8400 "Position Sequencer" technology application · Software manual · 13467299 · DMS 1.0 EN · 07/2014 · TD05

Lenze Drives GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
+49 5154 82-0
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Inverter

8400

"Winder Dancer-controlled" technology application
for 8400 TopLine C ____________________________

Software Manual EN

13538921

Lenze

Contents

1 About this documentation

This documentation describes the software-based solution of a task. The transferability of the described solution to the respective application needs to be checked by the user. If required, the user has to adapt the solution accordingly. Physical aspects such as the drive dimensioning are not part of this documentation.

Danger!

The inverter is a source of danger which may lead to death or the severe injury of persons.

To protect yourself and others against these dangers, observe the safety instructions before switching on the inverter.

Please read the safety instructions provided in the **Inverter Drives 8400 mounting instructions** and in the **Inverter Drives 8400 hardware manual**. Both documents are supplied with the inverter.

Target group

This documentation addresses to all persons who ...

- want to use the "Winder Dancer-controlled" technology application for the Inverter Drive 8400 TopLine and ...
- who are familiar with handling the device and the »Engineer« software.

Information regarding the validity

The information in this documentation applies to the following technology applications:

Technology application	From version
Winder Dancer-controlled	1.0

Screenshots/application examples

All screenshots provided in this documentation are application examples. Depending on the software version of the inverter and the version of the »Engineer« software installed, the screenshots in this documentation may differ from the representation in the »Engineer«.

Tip!

Information and tools for Lenze products are provided in the Download area at

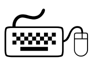
http://www.lenze.com → Download

1.1 Document history

Version			Description
1.0	06/2017	TD29	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the "0x" prefix is used. Example: 0x60F4
Binary number	0b	For binary numbers, the "0b" prefix is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that only applies to a certain inverter software version or higher is identified accordingly in this documentation. Example: This function extension is available from software version V3.0 onwards!
Program name	» «	The »Engineer«... Lenze PC software
Window	*italics*	The *Message window*... / The dialog box *Options*...
Variable names		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button ... / The **Copy** command ... / The **Properties** tab ... / The **Name** input field ...
Sequence of menu commands		If several commands must be used in sequence to carry out a function, the individual commands are separated by an arrow: Select **File**➞**Open** to...
Shortcut	**<bold>**	Use **<F1>** to open the online help.
		If a shortcut is required for a command to be executed, a "+" has been put between the key identifiers: With **<Shift>+<ESC>** ...
Hyperlink	<u>underlined</u>	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(🕮 4)	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

<table>
<tr><th>Term</th><th colspan="2">Meaning</th></tr>
<tr><td rowspan="4">Engineering tools</td><td colspan="2">Software solutions for easy engineering in all project stages</td></tr>
<tr><td></td><td>»EASY Navigator« – ensures easy operator guidance
• All convenient Lenze engineering tools at a glance
• Tools can be quickly selected
• The clear structure simplifies the engineering process from the start</td></tr>
<tr><td></td><td>»EASY Starter« – simple tool for service technicians
• Specially designed for the commissioning and maintenance of Lenze devices
• Graphical user interface with only a few buttons
• Simple online diagnostics, parameterisation, and commissioning
• No risk of an unintended change in applications
• Loading of ready-to-use applications to the device</td></tr>
<tr><td></td><td>»Engineer« – multi-drive engineering
• For all products in our L-force portfolio
• Practical user interface
• Easy handling by graphical user interfaces
• Can be applied in every phase of a project (project planning, commissioning, production)
• Parameter setting and configuration</td></tr>
<tr><td>Code</td><td colspan="2">Parameter used for inverter parameterisation or monitoring.
Is usually referred to as "index".</td></tr>
<tr><td>Subcode</td><td colspan="2">If a code contains several parameters, they are stored in "subcodes".
This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3").
Is usually referred to as "subindex".</td></tr>
<tr><td>Lenze setting</td><td colspan="2">This setting is the default factory setting of the device.</td></tr>
<tr><td>FB Editor</td><td colspan="2">Abbreviation for "function block editor". Graphic interconnection tool which is available in the »Engineer« Abbreviation for "function block editor". Graphic interconnection tool which is available in the FB Editor tab.</td></tr>
<tr><td>Function block</td><td colspan="2">General designation of a function block for free interconnection in the FB Editor.
A function block (short: "FB") can be compared to an integrated circuit which contains a specific control logic and supplies one or more values when it is executed.
Example: "L_Arithmetic_1" (FB for arithmetic operations)
Many function blocks are available several times (e.g. L_And_1, L_And_2, and L_And_3).</td></tr>
<tr><td>System block</td><td colspan="2">In the function block editor of the »Engineer«, system blocks provide interfaces to basic functions, "free codes", and to the hardware of the inverter (e.g. to the digital inputs). Each system block is available only once.</td></tr>
<tr><td>Port block</td><td colspan="2">Block for implementing the process data transfer via a fieldbus</td></tr>
<tr><td>LP</td><td colspan="2">Abbreviation for Lenze Port block
Example: "LP_CanIn1" (CAN1 port block)</td></tr>
<tr><td>LS</td><td colspan="2">Abbreviation for Lenze System block
Example: "LS_DigitalInput" (system block for digital input signals)</td></tr>
<tr><td>MCI</td><td colspan="2">Abbreviation for Motionbus Communication Interface (fieldbus interface)
The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system.</td></tr>
<tr><td>Technology application</td><td colspan="2">A technology application is a drive solution based on Lenze's experience and know-how, in which function blocks interconnected to a signal flow form the basis for implementing typical drive tasks.</td></tr>
<tr><td>USB diagnostic adapter</td><td colspan="2">The USB diagnostic adapter is used for the operation, parameterisation, and diagnostics of the inverter. Data are exchanged between the PC (USB connection) and the inverter (diagnostic interface on the front) via the diagnostic adapter. Order designation: E94AZCUS</td></tr>
</table>

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

> **Pictograph and signal word!**
>
> (characterise the type and severity of danger)
>
> **Note**
>
> (describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling

2 Properties of the technology application (TA)

The dancer position control is mainly used in applications requiring a high absolute tensile force accuracy and tensile force stability. In this control mode, the winding motor is speed-controlled.

When dancer position control is used, the web tension is exclusively generated by a dancer device installed upstream of the winder. The forte of this process is its compensation performance with dynamic disturbances like they can for instance occur during acceleration or in the case of imbalances in the reel. When this happens, the dancer position may vary, however, the tensile force will remain constant.

With the "Winder Dancer-controlled" technology application, a dancer position-controlled winding drive can be implemented on an Inverter Drive 8400 TopLine **from V14.00 onwards**.

2.1 Control mode for winding processes

The following technology applications (TA) designed by Lenze can be used for winding processes:

"Winder Dancer-controlled" TA

With **dancer position control**, the drive is operated in the speed control mode. For feedforward control, the line speed signal multiplied by the reciprocal value of the diameter is used. The dancer position is recorded and compared with the setpoint position. If there is a deviation, the dancer position controller corrects the speed setpoint.

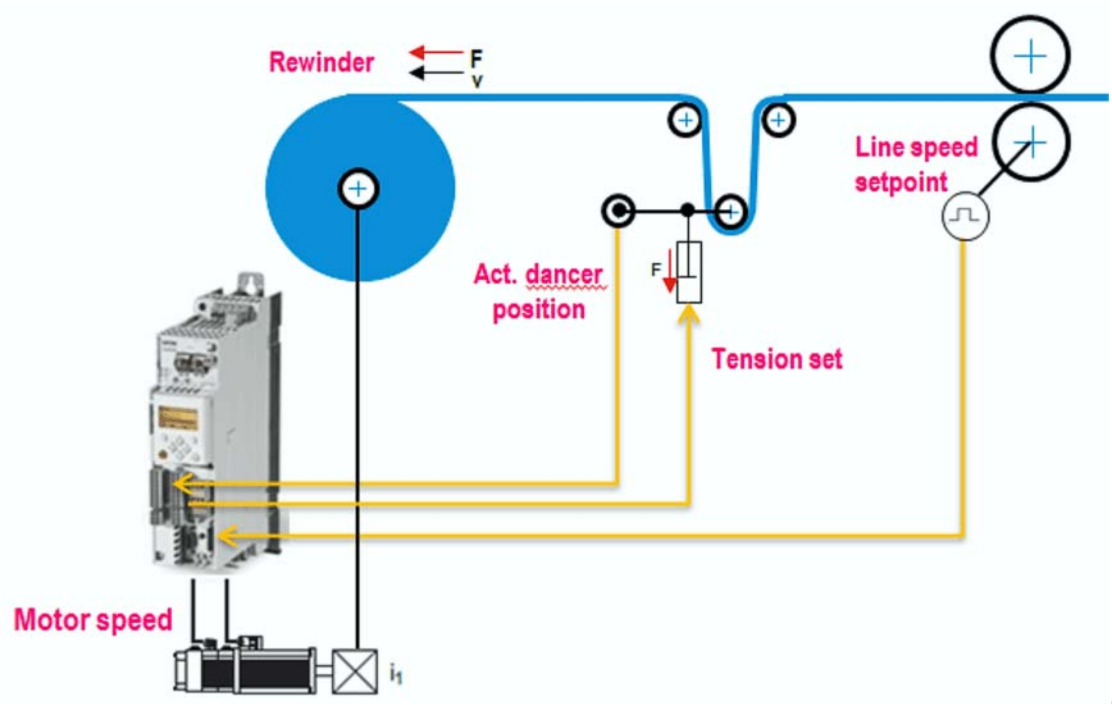
"Winder Tension-controlled" TA

With **tension control open loop/closed loop**, the drive torque is directly provided. A higher-level speed control only takes corrective action in the event of a web break, in order to limit the speed of the drive. To prevent the setpoint torque from being affected by the speed limitation during normal operation, a speed offset must be added to the speed setpoint calculated from the current line speed and the current diameter. The torque setpoint is composed of the tensile force setpoint multiplied by the current radius, the correcting signal for the compensation of the mechanical friction and the correcting signal for the compensation of the acceleration torque.

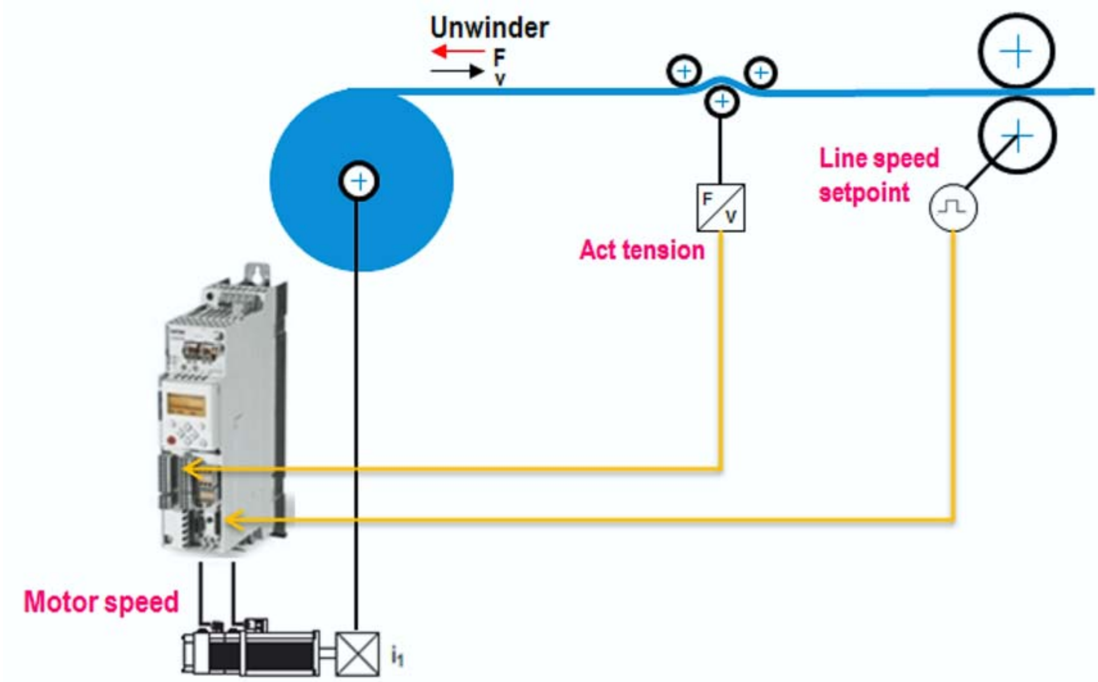
2.1.1 Comparison of the functional principles

Winder Dancer-controlled -> dancer position control	**Winder Tension-controlled -> tension control open loop/closed loop**
• The web tension force is determined by the dancer mechanics	• Depending on the tensile force setpoint and the current diameter of the winding material, the drive torque directly defines the tensile force at the winding material.
• The drive is actuated in speed control mode.	• The drive is operated as torque actuator.
• The line speed signal multiplied by the reciprocal value of the winding material diameter is used for feedforward control. • The dancer position is recorded and compared to the setpoint position. If a deviation is detected, the dancer position controller corrects the speed setpoint. • An acceleration torque can be compensated. • A dancer actuator can be optionally controlled from the inverter, e.g. in order to implement a reduction in tensile force with an increasing diameter.	• The line speed signal multiplied with the reciprocal value of the reel diameter + offset serves as a limit for speed limitation. • If an optional tensile force recognition system is provided, deviations can be compensated using a process controller (closed loop). • An acceleration torque and the friction losses can be compensated.

• The diameter is calculated in the inverter.
• The control and setpoint selection are optionally taken over by ...
  • a HMI.
  • the digital and analog interfaces of the inverter.
  • a higher-level control via a bus system.	



[2-1] Functional principle of a dancer position control ("Winder Dancer-controlled" TA)



[2-2] Functional principle of a tension control open loop/closed loop ("Winder Tension-controlled" TA)

2.1.2 Functional overviews

<table>
<tr><th>Winder Dancer-controlled
-> dancer position control</th><th>Winder Tension-controlled
-> closed loop/open loop tension control</th></tr>
<tr><td colspan="2">• Inching mode with ramp generator
• Diameter calculation from line speed and winding speed
• Holding/setting the diameter value
• Web break monitoring using the diameter calculator
• Winding from the top or the bottom
• Automatic detection of the winding direction (unwinding/rewinding) by means of the sign of the line speed</td></tr>
<tr><td>• If the dancer actuator is controlled from the inverter: reduction of tensile force via the characteristic function for rewinders (winding characteristic)</td><td>• Reduction of tensile force via the characteristic function for rewinders (winding characteristic)</td></tr>
<tr><td>• Acceleration compensation</td><td>• Acceleration compensation
• Friction compensation
• Ramp generator for tensile force setpoint
• Speed limitation via the line speed plus the offset, 1/d-evaluated</td></tr>
<tr><td>• PI controller dancer position control with various possibilities of adaptation
• "Teach-In" function for the dancer limit positions
• consideration of the dancer movement in the diameter calculation</td><td>• PI controller tension control with various possibilities of adaptation</td></tr>
<tr><td colspan="2">• Adaptation of the speed controller gain as a function of the current moment of inertia</td></tr>
</table>

In the following, the "Winder Dancer-controlled" technology application is described. A description of the "Winder Tension-controlled" TA can be found in the separate software manual of the TA.

2.2 Application ranges

- Rewinding and unwinding of material webs in the field of surface finishing
- Winding facilities in the textile industry
- Unwinders in packaging machines
- Rewinders/unwinders of round materials (threads, wires, cables, tubes,...)

... and many things more.

2.3 System requirements

Software

Product	Order designation	From version
»Engineer«	ESPEV-EHNNN	2.22.1.0

Hardware

Product	Order designation	From hardware version	From software version
Inverter Drive 8400 TopLine	E84AVTCxxxxx	VD	14.00

2.4 Parameter setting in the function block editor view

Enter application-specific parameters directly in the function block editor (FB Editor). In this way, the signal flow is traceable and the interaction of the blocks is illustrated.

Furthermore you can use the FB Editor to configure the I/O interconnection and monitor the applications running on the device, e.g. for diagnostics purposes.

- Open the parameterisation dialog or the parameter list for the block ...
  - via the icon in the block header,
  - by double-clicking the block, or
  - by executing the **Parameter...** command in the *context menu* for the block.
- Colour coding and coloured comments provide for a clearly arranged structure of the FB Editor.
  - The areas highlighted in turquoise represent the "user interface". If required, the pre-assignment of the I/O terminals can be adapted here and a control via the fieldbus interface (MCI) can be established.
  - In the areas highlighted in yellow, application-specific settings are required.

Reference manual / online help for the Inverter Drive 8400

In the "Working with the FB Editor" chapter you'll find some detailed information on how to handle the FB Editor.

2.5 Pre-assignment of the user interface

<table>
<tr><th>Terminal</th><th colspan="3">Function</th></tr>
<tr><td colspan="4">Digital input terminals</td></tr>
<tr><td rowspan="4">X5/RFR</td><td colspan="3">Controller enable</td></tr>
<tr><td colspan="2">RFR</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Inhibit drive</td></tr>
<tr><td colspan="2">HIGH</td><td>Enable drive</td></tr>
<tr><td>X5/DI1</td><td colspan="3">- (reserved for HTL encoder)</td></tr>
<tr><td>X5/DI2</td><td colspan="3">- (reserved for HTL encoder)</td></tr>
<tr><td rowspan="4">X5/DI3</td><td colspan="3">Enable winder (follow line speed)</td></tr>
<tr><td colspan="2">DI3</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Inhibit winder</td></tr>
<tr><td colspan="2">HIGH</td><td>Enable winder</td></tr>
<tr><td rowspan="4">X5/DI4</td><td colspan="3">Enable dancer control</td></tr>
<tr><td colspan="2">DI3</td><td>Function</td></tr>
<tr><td colspan="2">LOW</td><td>Inhibit dancer control</td></tr>
<tr><td colspan="2">HIGH</td><td>Enable dancer control</td></tr>
<tr><td rowspan="6">X5/DI5
X5/DI6</td><td colspan="3">Selection of a preset setpoint for inching mode. At activation, an additional acceleration time is activated; positive manual inching</td></tr>
<tr><td>DI5</td><td>DI6</td><td>Function</td></tr>
<tr><td>LOW</td><td>LOW</td><td>Follow the line speed</td></tr>
<tr><td>HIGH</td><td>LOW</td><td>Selection of preset setpoint 1 = C00039/1 = 10 %
• "Positive inching</td></tr>
<tr><td>LOW</td><td>HIGH</td><td>Selection of preset setpoint 2 = C00039/2 = -10 %
• "Negative inching</td></tr>
<tr><td>HIGH</td><td>HIGH</td><td>Selection of preset setpoint 3 = C00039/3 = 0 %</td></tr>
<tr><td>X5/DI7</td><td colspan="3">HIGH = reset error</td></tr>
<tr><td colspan="4">Analog input terminals</td></tr>
<tr><td>X3/A1U</td><td colspan="3">Line velocity
• Scaling: 10 V = 100 % reference line speed (C0471/3)</td></tr>
<tr><td>X3/A2U</td><td colspan="3">Actual value of dancer position
• Scaling: 10 V = 100 % reference)</td></tr>
<tr><td colspan="4">Digital output terminals</td></tr>
<tr><td>X4/DO1</td><td colspan="3">HIGH ≡ "Drive is ready" state</td></tr>
<tr><td>X4/DO2</td><td colspan="3">Not assigned, can be freely used</td></tr>
<tr><td>X4/DO3</td><td colspan="3">Not assigned, can be freely used</td></tr>
<tr><td>X107/BD1, BD2</td><td colspan="3">Control of a holding brake by the basic function "holding brake control"</td></tr>
<tr><td>X101/COM, NO</td><td colspan="3">Relay contact closed ≡ "An error is pending" state</td></tr>
<tr><td colspan="4">Analog output terminals</td></tr>
<tr><td>X3/O1U</td><td colspan="3">Actual speed value
• Scaling: 10 V ≡ 100 % reference speed (C00011)</td></tr>
<tr><td>X3/O2U</td><td colspan="3">Actual torque value
• Scaling: 10 V ≡ 100 % maximum torque (C00057)</td></tr>
</table>

3 Short setup of the technology application

3.1 Preconditions

For the execution of the short setup described in the following, it is assumed that the basic parameters (motor, feedback system, etc.) have been set.

The "8400 commissioning wizard" enables guided commissioning of the Inverter Drive 8400, taking the Lenze parameter setting as a basis.

How to proceed:

1. Before switching on: ensure that the inverter is inhibited (digital input terminal X5/RFR open).
2. Switch on voltage supply of the inverter.
   The parameterisation and diagnostics of the inverter without motor operation solely requires an external 24 V supply by a safely separated power supply unit (SELV/PELV).
3. Establish a communication link between the inverter and the Engineering PC, e.g. via a USB diagnostic adapter (E94AZCUS):
   - connect the USB diagnostic adapter to the X6 diagnostic interface.
   - establish a connection between the USB diagnostic adapter and the PC via a free USB port.
4. Start the »Engineer« on the Engineering PC, e.g. via the Windows start menu:
   Start → All programs → Lenze → Engineering → L-force Engineer...
   After the program start, no project has been loaded first and the *start-up wizard* is displayed.

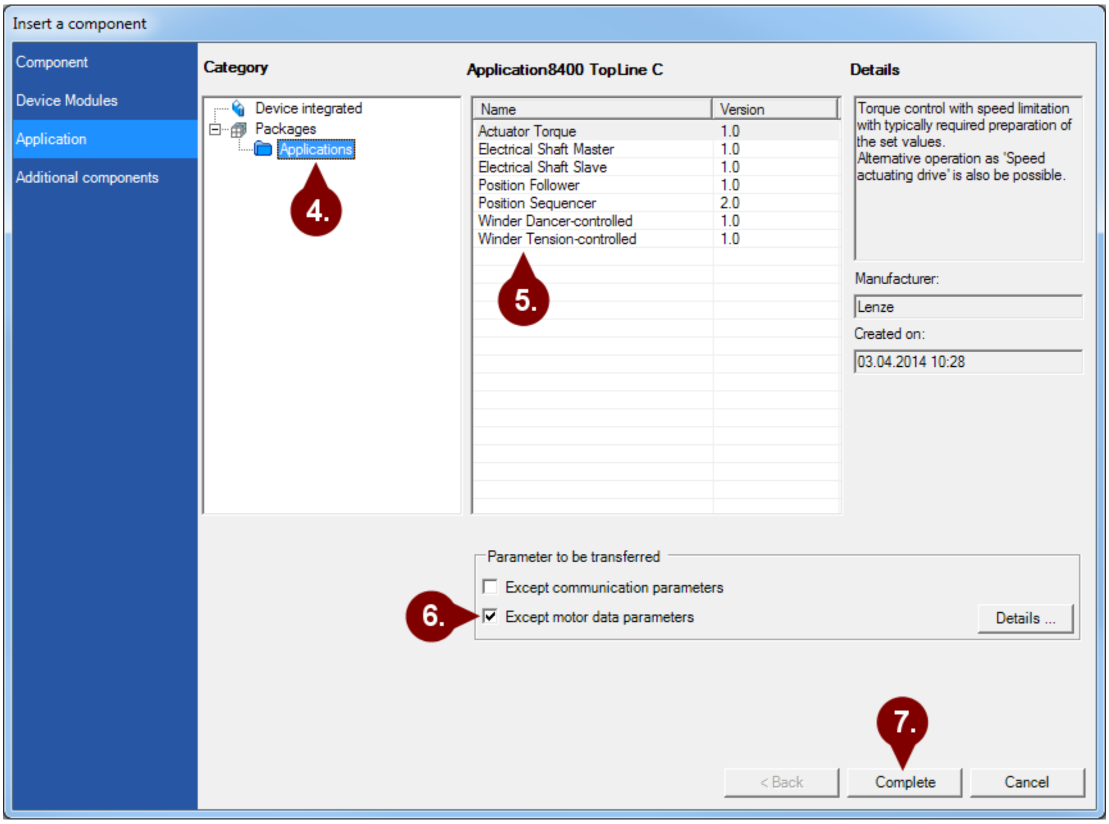
> **»Engineer« Online help**
> Here you'll find some detailed information with regard to the options of the start-up wizard and to the general handling of the »Engineer«.

5. Create a new project or open a project already available.
6. Go to the *Project view* and select the 8400 inverter.
7. Click the icon to go online.
   When the connection to the inverter has been established successfully, the *status line* shows ONLINE .
8. Click on the icon to start the "8400 commissioning wizard".
   - Now the commissioning wizard guides you step by step through the setting of the important parameters for a quick commissioning.
   - The **Next** button can only be activated again after all parameter settings in the device have been reset via the **Load Lenze setting** button.
   - Execute the commissioning wizard right up to the end.
   - You can skip the "Control mode" step by clicking **Next** (only relevant for "Speed actuating drive" technology application).

3.2 Step 1: load "Winder Dancer-controlled" technology application

In the Lenze setting, the inverter uses the "Actuator speed" technology application integrated in the device. Execute the following steps to use the "Winder Dancer-controlled" technology application instead:

1. Go to the *Project view* and select the inverter.
2. If there is still an online connection to the inverter, click on the icon to go offline.
   The application can only be selected when you are offline.
3. Click the icon to select another application.
   The *Insert application* dialog box appears:



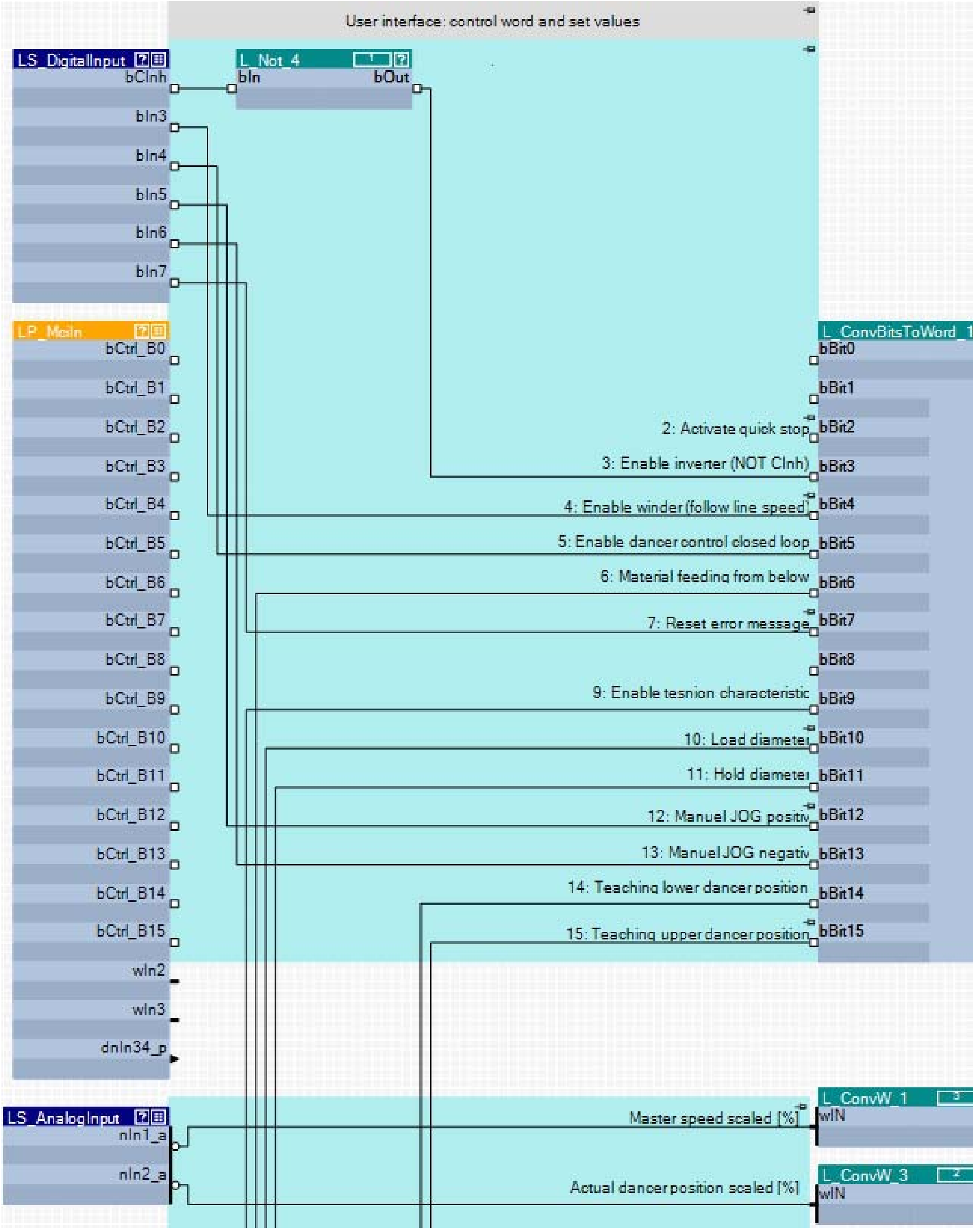
4. In the left field, select the **Packages → Applications** category.
5. In the right field, select the **"Winder Dancer-controlled"** application.
6. Activate the **Except motor data parameters** option in order that the settings of the motor data parameters made before will not be overwritten.
7. Press **Complete** to close the dialog box again and load the application selected into the »Engineer« project.
8. Confirm the prompt on whether the current application is to be replaced by the new application with **Yes**.

3.3 Step 2 (optional): set up control via the fieldbus interface (MCI)

The Lenze default setting provides for controlling the application via the digital input terminals and specifying setpoints via the analog input terminals. If this is desired, continue with step 3.

If the fieldbus interface (MCI) is to be used instead of the terminals, the user interface in the FB Editor (area highlighted in turquoise) is to be adapted accordingly for the application control word and the setpoints.

- The assignment of the outputs (on the left) to the inputs (on the right) can be changed at will.
- The inputs (on the right) are permanently linked to functions of the application.



[3-1] Example of an interconnection

3.3.1 Pre-assignment of the process data input words

In the model connection (see Fig. [3-1]), the process data input words are assigned as follows:

Input word	Assignment
Word 1	Control word (for bit assignment see the following table)
Word 2	Line velocity • Scaling: 16384 ≡ 100 % reference speed C00011
Word 3	Current dancer position • Scaling: 16384 ≡ 100 % dancer in the position with a minimum stored material length (upper limit position)
Word 4 ... 16	Not preconfigured

<table>
<tr><th>Control word</th><th colspan="3">Function</th></tr>
<tr><td>Bit 0</td><td colspan="3">Not preconfigured</td></tr>
<tr><td>Bit 1</td><td colspan="3">Not preconfigured</td></tr>
<tr><td>Bit 2</td><td colspan="3">1 ≡ Activate quick stop (QSP)</td></tr>
<tr><td>Bit 3</td><td colspan="3">1 ≡ Enable inverter (RFR)</td></tr>
<tr><td>Bit 4</td><td colspan="3">1 ≡ enable winder</td></tr>
<tr><td>Bit 5</td><td colspan="3">1 ≡ enable dancer control</td></tr>
<tr><td>Bit 6</td><td colspan="3">0 ≡ material feeding from the top
1 ≡ material feeding from the bottom</td></tr>
<tr><td>Bit 7</td><td colspan="3">1 ≡ reset error (Trip reset)</td></tr>
<tr><td>Bit 8</td><td colspan="3">Not preconfigured</td></tr>
<tr><td>Bit 9</td><td colspan="3">1 ≡ enable reduction in tensile strength for rewinder (tension characteristic)</td></tr>
<tr><td>Bit 10</td><td colspan="3">1 ≡ load diameter</td></tr>
<tr><td>Bit 11</td><td colspan="3">1 ≡ hold diameter</td></tr>
<tr><td rowspan="6">Bit 12 ... 13</td><td colspan="3">Selection of preset setpoints for inching mode</td></tr>
<tr><td>Bit 12</td><td>Bit 13</td><td>Function</td></tr>
<tr><td>0</td><td>0</td><td>Inching mode not active</td></tr>
<tr><td>1</td><td>0</td><td>Selection of fixed setpoint 1 = C00039/1 = 10 % reference speed (C00011)
-> manual inching in positive direction</td></tr>
<tr><td>0</td><td>1</td><td>Selection of fixed setpoint 2 = C00039/2 = -10 % reference speed (C00011)
-> manual inching in negative direction</td></tr>
<tr><td>1</td><td>1</td><td>Selection of fixed setpoint 3 = C00039/3 = 0 % reference speed (C00011)</td></tr>
<tr><td>Bit 14</td><td colspan="3">1 ≡ teach-in of the lower dancer limit position</td></tr>
<tr><td>Bit 15</td><td colspan="3">1 ≡ teach-in of the upper dancer limit position</td></tr>
</table>

3.3.2 Pre-assignment of the process data output words

In the model connection (see Fig. [3-1]), the process data output words are assigned as follows:

Output word	Assignment
Word 1	Status word (for bit assignment see the following table)
Word 2	Actual speed value • Scaling: 16384 = 100 % reference speed C00011
Word 3	Current diameter • Scaling: 16384 = 100 % maximum diameter C0471/1
Word 4 ... 16	Not preconfigured

Status word	Status					
Bit 0	1 ≡ group error active (configurable in C00148)					
Bit 1	1 ≡ inverter control is inhibited (pulse inhibit is active)					
Bit 2	1 ≡ inverter is ready for operation					
Bit 3	1 ≡ quick stop is active					
Bit 4	1 ≡ setpoint torque is in limitation					
Bit 5	1 ≡ speed controller is in limitation					
Bit 6 **Bit 6**	During open-loop operation: 1 ≡ speed setpoint < comparison value (C00024)					
	During closed-loop operation: 1 ≡ actual speed value < comparison value (C00024)					
Bit 7	1 ≡ inverter is inhibited (controller inhibit is active)					
Bit 8 ... 11	Bit 11	Bit 10	Bit 9	Bit 8	Device status	Meaning
	0	0	0	0	FirmwareUpdate	Firmware update function is active
	0	0	0	1	Init	Initialisation active
	0	0	1	0	Ident	Identification active
	0	0	1	1	ReadyToSwitchOn	Device is ready to start
	0	1	0	0	SwitchedOn	Device is switched on
	0	1	0	1	OperationEnabled	Operation
	0	1	1	0	-	-
		1	1	1	Trouble	Trouble active
	1	0	0	0	Fault	Fault active
	1	0	0	1	TroubleQSP	TroubleQSP is active
	1	0	1	0	SafeTorqueOff	Safe torque off is active
	1	0	1	1	SystemFault	System fault active
Bit 12	1 ≡ a warning is indicated					
Bit 13	1 ≡ inverter is in the "Trouble" state					
Bit 14	1 ≡ dancer position-controlled operation active					
Bit 15	1 ≡ web break (implausible diameter change)					

3.4 Step 3: Setting the commissioning parameters

Open the parameterisation dialog or the parameter list for the block ...

- via the icon in the block header,
- by double-clicking the block, or
- by executing the **Parameter...** command in the *context menu* for the block.

For quick commissioning, only the following application-specific parameters have to be set or their default setting has to be checked:

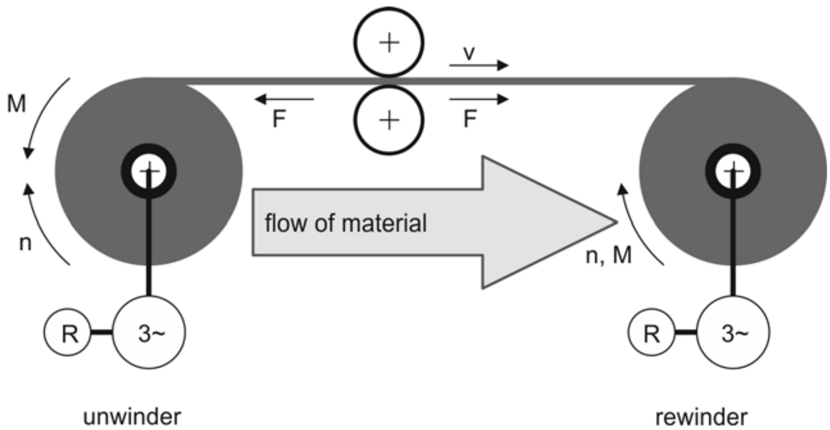
- "Normal" winding direction (rewinder or unwinder)
- Winding from the top or the bottom
- Reference variables

In the following, these parameters are described in detail.

3.4.1 Winding direction (rewinder or unwinder)

To ensure that the feedforward control values, the disturbance compensation, and the correcting signal of the dancer position controller always actin the correct direction, the "normal" winding direction must be defined once.

- If the drive **unwinds** the material while the line speed is positive, **C00470/1 = 1** (unwinder) must be set.
- If the drive **rewinds** the material while the line speed is positive, **C00470/1 = 0 (**rewinder) must be set.



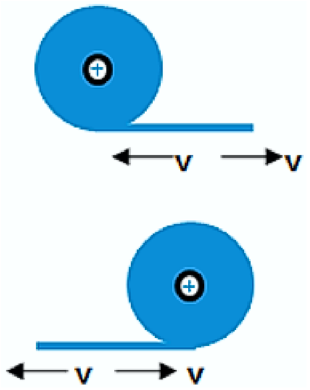
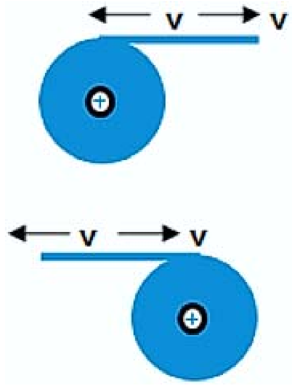
Note!

When the "normal" winding direction has been defined once, the winding drives can also run in the opposite direction, with a negative line speed. Intervening in the signal flow is not necessary for this.

By means of the parameter C01206/1, "motor mounting direction = inverted", a potential inversion of the direction of rotation due to uneven-numbered gearbox stages and/or an inverted motor mounting position can be eliminated.

3.4.2 Material feeding (winding from the top or the bottom)

Define the material feeding (effective direction of speed and torque at the motor shaft) in code C00470/2 or using control bit 6.

C00470/2 = 1 (material feeding from the bottom)	C00470/2 = 0 (material feeding from the top)
	

3.4.3 Defining reference variables

The setpoints for the speed, line speed, and dancer position are processed as scaled values. Therefore the reference variables have to be parameterised accordingly.

The reference variable for all scaled speeds is C00011. This value must be parameterised so that it matches the present machine data (gearbox factor i, minimum diameter d_{min}) and the reference line speed v_100 %.

The reference of the dancer position is defined during commissioning by the teach-in of the dancer limit positions.

▸ Step 7 (optional): Teach-in of the dancer limit positions (📖 27)

The reference variables are preselected via the basic parameters described in the following.

Note!

The **basic parameters** must be defined at the start of commissioning.

The **other parameters** can be adapted as optimisation parameters in the course of commissioning.

The Lenze settings are selected so that the technology applications can be tested on the Lenze winder training model:

- Left -> "Winder Tension-controlled" TA as unwinder
- Right -> "Winder Dancer-controlled" TA as rewinder

Parameter (block)	Possible setting		Info
Basic parameter			
C00011	50 ... 60000		Motor speed at reference line speed and minimum diameter C00011 = ROUNDS(i/(d_min/1000*π)*v_100 %) i: total ratio between motor shaft and winding shaft d_min: minimum diameter [mm] v_100 %: reference line speed [m/min] **Please note:** This value must be defined accordingly and adapted when v_100%, i or d_min is changed. • Lenze setting: 955 rpm
C00470/1 (LS_ParFree_b)			Function of the winder with a positive line speed signal • Lenze setting: 1
	0	Rewinder	
	1	Unwinder	
C00470/2 (LS_ParFree_b)			Material feeding • Lenze setting: 1
	0	From the top	
	1	From the bottom	
C01202/1 (LS_MotionControlKernel)	1 ... 65535		Gearbox factor numerator Z2 • Lenze setting: 1
C01202/2 (LS_MotionControlKernel)	1 ... 65535		Gearbox factor denominator Z1 • Lenze setting: 1
C00471/1 (LS_ParFree)	1 ... 65535		Maximum diameter [mm] • Lenze setting: 200
C00471/2 (LS_ParFree)	1 ... 65535		Minimum diameter [mm] • Lenze setting: 50
C00471/3 (LS_ParFree)	1 ... 65535		Reference line speed [0.1 x m/min] • Lenze setting: 1500
C00471/4 C00471/5 (LS_ParFree)	1 ... 65535		Conversion factor tensile force to torque C00471/4 = numerator = F_100% x 10 C00471/5 = denominator = d_max / 1000 x C00057 x 10 F_100 % = rated tensile force [Nm] d_max = maximum diameter [mm] C00057 = maximum torque (online value according to corresponding parameterisation of the motor and C00022) • Lenze setting: C00471/4=50; C00471/5=102
C00472/5 (LS_ParFree)	-199.99 % ... 199.99 %		Threshold of minimum line speed • Lenze setting: 1 %
C00472/6 (LS_ParFree)	-199.99 % ... 199.99 %		Comparison value of the line speed sign recognition • Lenze setting: -0.2 %
Condition speed setpoint /limitation threshold values; inching mode			
C00470/9 (LS_ParFree_b)			Adaptive adjustment of the speed controller gain **Please note:** requires a correspondingly parameterised acceleration compensation. • Lenze setting: 0
	0	Adaptation active	The speed controller gain is set to a permanent value (C00070).
	1	Adaptation not active	The speed controller gain is increased linearly to the current moment of inertia.

Parameter (block)	Possible setting	Info
C00660/2 (L_FixSet_a_1)	-199.99 % ... 199.99 %	Line speed offset for the speed limitation with a positive line speed signal Lenze setting: 5 %
C00660/4 (L_FixSet_a_1)	-199.99 % ... 199.99 %	Line speed offset for the speed limitation with a negative line speed signal • Lenze setting: 5 %
C00721/2 (L_DigitalDelay_2)	0.00 ... 3600.00 s	Delay of the deceleration time deactivation for the inching mode • Lenze setting: 0.1 s
C00039/1	-199.99 % ... 199.99 %	Preset setpoint 1: positive inching • Lenze setting: 10 %
C00039/2	-199.99 % ... 199.99 %	Preset setpoint 2: negative inching • Lenze setting: -10 %
C00039/3	-199.99 % ... 199.99 %	Preset setpoint 3: alternative inching speed • Lenze setting: 5 %
C00101/1	0.001 ... 999.99 s	Inching mode: acceleration time • Lenze setting: 5 s
C00101/3	0.001 ... 999.99 s	Inching mode: deceleration time • Lenze setting: 5 s
C00677/1	-199.99 % ... 199.99 %	Adaptation factor of the moment of inertia for the speed controller adaptation • Lenze setting: 60 %
C00677/1	-199.99 % ... 199.99 %	Lower limitation of the speed controller adaptation • Lenze setting: 40 %

Parameter (block)	Possible setting		Info
Tensile force open loop control/closed loop control, tensile force characteristic, tensile force controller			
C00470/3 (LS_ParFree_b)			Enable tension control. • Lenze setting: 0
	0	Tensile force controller reset	
	1	Tensile force controller active	
C00472/2 (LS_ParFree)	-199.99 % ... 199.99 %		Tensile force setpoint scaled [%] • Lenze setting: 20 %
C01040/1 (L_SRFG_1)	0 ... 999.99 s		Acceleration time for tensile force setpoint • Lenze setting: 0.5 s
C00472/7 (LS_ParFree)	-199.99 % ... 199.99 %		Influence of tensile force controller • Lenze setting: 5 %
C01056/1 (LS_ProcessCtrl_1)	0 ... 100		Tensile force controller gain • Lenze setting: 1
C01053/4 (LS_ProcessCtrl_1)	0 ... 30 s		Tensile force controller reset time • Lenze setting: 5 s
C01053/2 (LS_ProcessCtrl_1)	0 ... 30 s		Filter time constant for actual tensile force value • Lenze setting: 0.1 s
C00470/4 (LS_ParFree_b)			Enable tensile force reduction for rewinder (tensile force characteristic). • Lenze setting: 0
	0	Tensile force adaptation not active	The tensile force setpoint is not adapted.
	1	Tensile force adaptation active	Tensile force is reduced with an increased diameter according to the characteristic in the L_Curve_3.
C01030/1 (L_Curve_3)	4	Characteristic	L_Curve_3: do not change function! • Lenze setting: 4
C01035/1 (L_Curve_3)	0	Linear tensile force profile	L_Curve_3: selection of the tensile force characteristic • Lenze setting: 0
	1	Linear torque profile	
	2	Tensile force profile according to characteristic	
C00472/3 (LS_ParFree)	-199.99 % ... 199.99 %		Starting point of the tensile force characteristic • Lenze setting: 50 %
C00472/4 (LS_ParFree)	-199.99 % ... 199.99 %		Slope of the tensile force characteristic • Lenze setting: 60 %

Parameter (block)	Possible setting		Info
Friction compensation			
C00960/1 (L_Curve_1)			Activate friction compensation. • Lenze setting: 0
	0	Inactive	The compensation torque is permanently set to 0 %.
	3	Active	Via the characteristic of *L_Curve_1*, a speed-dependent compensation torque is connected.
C00963/1 ... 32 (L_Curve_1)	-32767 ... 32767		X values of the tensile force characteristic (speed)
C00964/1 ... 32 (L_Curve_1)	-32767 ... 32767		Y values of the tensile force characteristic (torque)
Acceleration compensation			
C01025/1 (L_Curve_2)			Enable acceleration compensation. • Lenze setting: 0 **Note:** The acceleration compensation can only be used reasonably if the cam table (C01028/C01029) and blocks *L_DT1*, *L_ConW_2*, and *L_ConW_4* are parameterised accordingly. Use the "Tool_TA8400_TensionControlled_V1-0.xlsx" Excel support tool to determine the matching parameters of these blocks!
	0	Inactive	The compensation torque is permanently set to 0 %.
	3	Active	Via the characteristic of the *L_Curve_2* block and the current diameter, a compensation torque is connected, which is proportional to the inertia.
C01028/1 c 32 (L_Curve_2)	-32767 ... 32767		X values of the characteristic adaptation of the moment of inertia
C01029/1 c 32 (L_Curve_2)	-32767 ... 32767		Y values of the characteristic adaptation of the moment of inertia
C00251/0 (L_DT1)	10 ... 5000 ms		L_DT1 time constant • Lenze setting: 10 ms (do not change)
C00252/0 (L_DT1)	-320 ... 320		Speed differentiation gain • Lenze setting: 100
C00940/2 (L_ConvW_2)	-32767 ... 32767		Acceleration scaling • Lenze setting: 1000
C00940/4 C0941/4 (L_ConvW_4)	-32767 ... 32767		Scaling of the moment of inertia • Lenze setting: C00940/4 = 330; C00941/4 = 1020
C00472/8 (LS_ParFree)	-199,99 % ... 199,99 %		Material width scaled • Lenze setting: 100 %

Parameter (block)	Possible setting		Info
Calculation of the diameter			
C00470/5 (LS_ParFree_b)			Load diameter.
	0	-	
	1	Load diameter	The *nSetDiameter_a* input value is accepted as the current diameter.
C00470/6 (LS_ParFree_b)			Hold last diameter value.
	0		Diameter is recalculated cyclically.
	1		No recalculation of diameter. The value calculated or loaded last is maintained.
C00470/7 (LS_ParFree_b)			Activate web break monitoring (*bUniDirect*). • Lenze setting: 0
	0	Inactive	The diameter may change in rewinding and unwinding direction.
	1	Active	An implausible change in diameter opposed to the winding direction preselected via *bUnwind* AND beyond the C01052/1 tolerance window sets the *bWebBreak* output to TRUE. The calculation of the change in diameter in the opposite direction is prevented. Therefore only activate the web break monitoring if the *dwOutDiameter* output value corresponds to the real diameter.
C00470/8 (LS_ParFree_b)			Calculation cycle for the diameter calculation • Lenze setting: 0
	0	Standard cycle	Use diameter recalculation 0 (C01050/1).
	1	Short cycle	Use diameter recalculation 1 (C01050/1).
C00472/1 (LS_ParFree)			Start diameter scaled to *d_max* C471_1 • Lenze setting: 50 %

3.5 Step 4: Transferring the parameter set to the inverter

1. Click the icon to go online.
2. Click the icon to transmit the parameter set to the inverter.
3. After a successful transmission, click the icon to save the parameter set safe against mains failure in the integrated memory module.

3.6 Step 5: enabling the inverter

- Set motor (»Engineer« motor catalogue or nameplate)
  The motor rotates freely with an empty winding shaft.
- Load the diameter calculator with the minimum diameter so that the diameter is held on the smallest diameter possible during the calculation.
  In order to achieve this, ...
  - set C00472/1 = 0;
  - set the "Load diameter" control bit (C00470/5 = 1, bit 10 = 1);
  - set the "Hold diameter" control bit (C00470/5 = 1, bit 10 = 1).

3.7 Step 6: checking the parameter setting

3.7.1 Checking the winding direction

The "normal" winding direction has been defined in step 3:

▸ Winding direction (rewinder or unwinder) (📖 18)

The effective direction of a positive tensile force setpoint at the winding shaft is determined as follows:

- Select operation without feedback of the tensile force (open loop).
- Specify a small tensile force setpoint for line speed = 0.
- Enable winder and tension control open loop (DI3 = 1 and DI4 = 0 or bit 4 = 1 and bit 9 = 1).

The following effective directions are to be expected:

Winding direction	**Unwinder (C00470/1 = 1)**		**Rewinder (C00470/1 = 0)**	
Material feed	From the top (C00470/2 = 0)	From the bottom (C00470/2 = 1)	From the top (C00470/2 = 0)	From the bottom (C00470/2 = 1)
Movement of the winder	Counter-clockwise	Clockwise	Clockwise	Counter-clockwise

Note!

"Motor mounting direction = inverted" (C01206/1) may eliminate a potential inversion of the direction of rotation due to uneven-numbered gearbox stages and/or an inverted motor mounting position.

3.7.2 Check speed feedforward control

Check the speed feedforward control as follows:

- Enable the winder (DI3 = 1, bit 4 = 1) to actuate the drive in the "Follow line speed" operating mode.
- Start the line speed master and increase the speed to 50 %.

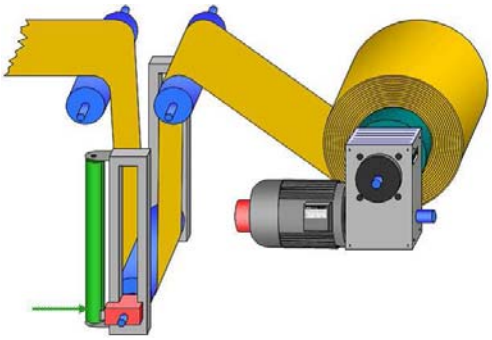
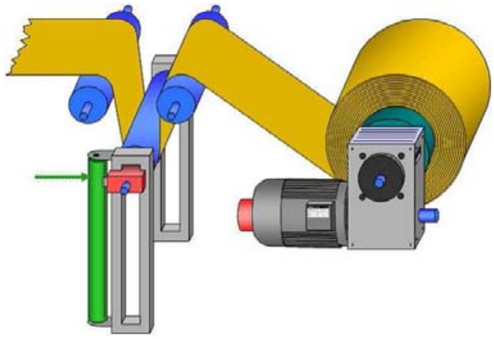
The winder must now rotate with half the speed parameterised in C00011.

When the material is fed from the top, the winding shaft rotates clockwise, whereas material feeding from the bottom makes the winding shaft rotate in counter-clockwise direction.

If the speed or direction of rotation is incorrect, check the above-mentioned definition of the basic parameters (C00470/1, C00470/2, C01206/1).

3.8 Step 7 (optional): Teach-in of the dancer limit positions

In order that the dancer signal between the lower and upper limit position is adjusted to an analog value of 0 ... 100 %, the respective reference values can either be entered directly in C01461/1 and C01461/2 or accepted using the teach-in function.

Dancer in the lower limit position (maximum material length saved)	Dancer in the upper limit position (minimum material length saved)
	

In order to carry out the teach-in process, proceed as follows:

- Traverse the dancer to the lower limit position and transfer the current analog value to C01461/2 using C00470/10 (control bit 14 = TRUE).
- Traverse the dancer to the upper limit position and transfer the current analog value to C01461/1 using C00470/11 (control bit 15 = TRUE).

The C00002/11 device command, "Save all parameter sets", is executed with the falling edge of the teach-in command in each case.

3.9 Step 8 (optional): Setting the disturbance compensation

In order to improve the dynamic properties, the following disturbance can be compensated:

- Acceleration torque

3.9.1 Setting the acceleration compensation

The disturbance of the acceleration torque can be compensated via a characteristic. This characteristic is defined via an Excel support tool (Tool_TA8400_DancerControlled_V1-0.xlsx) and is imported into the »Engineer« in the form of a gdc file.

Function of the Excel support tool

The acceleration torque required for instance depends on the constant mass inertia (motor + mechanics) and the variable moment of inertia (reel). By means of the Excel support tool, the moment of inertia is determined as a function of the diameter and is scaled to the maximum moment of inertia. The grid points of this function are stored in the *L_CurveW_2* block. The input variable is the scaled diameter of the reel. The output is scaled, i.e. to the total maximum moment of inertia (constant + variable) relating to the motor shaft.

The quantisation of the acceleration signal at the output of the *L_DT_1* block significantly increases with great acceleration times (> 5 s). In order to counteract this, the acceleration is scaled to a standard acceleration time. This is taken into consideration with the factors in the *L_DT_1* and *L_ConvW_2* blocks when determining the acceleration. The moment of inertia is scaled to the acceleration required for this purpose using block *L_ConvW_4*.

If possible, the line speed signal should be derived from a stable setpoint of the speed-determining drive in the machine. For a less stable line speed signal, the number of significant bits which are taken into consideration in the differentiation process so that acceleration jumps are avoided can be set in the *L_DT1_1* block.

Conditional equations for the scaling:

- Acceleration scaling
  Acceleration time t_acc less than or equal to 3.2 s:

$$\text{L\_DT1\_1, C00252: } t_{acc} * 10$$

$$\text{L\_ConvW\_2, C00940/2: } 1000$$

  Acceleration time t_acc greater than 3.2 s:

$$\text{L\_DT1\_1, C00252: } 320$$

$$\text{L\_ConvW\_2, C00940/2: } t_{acc} * \frac{100000}{320}$$

- Scaling of the moment of inertia to the reference acceleration:

$$\text{L\_ConvW\_4, C00940/4: } 2\pi * \frac{C00011}{60} * \frac{1}{t_{acc}} * J_{max}$$

$$\text{L\_ConvW\_4, C00941/4: } C00057 * 1000$$

J_max: maximum moment of inertia relating to the motor shaft in kgm2

t_acc: rated acceleration time of the winder in s

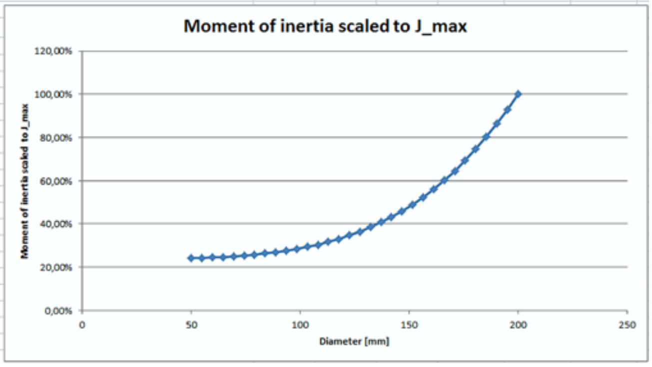
Handling the Excel support tool

The entries required for calculating the parameters are made in the support tool in the cells highlighted in yellow. Afterwards the development of the moment of inertia as a function of the current diameter is calculated and represented graphically.

Note!

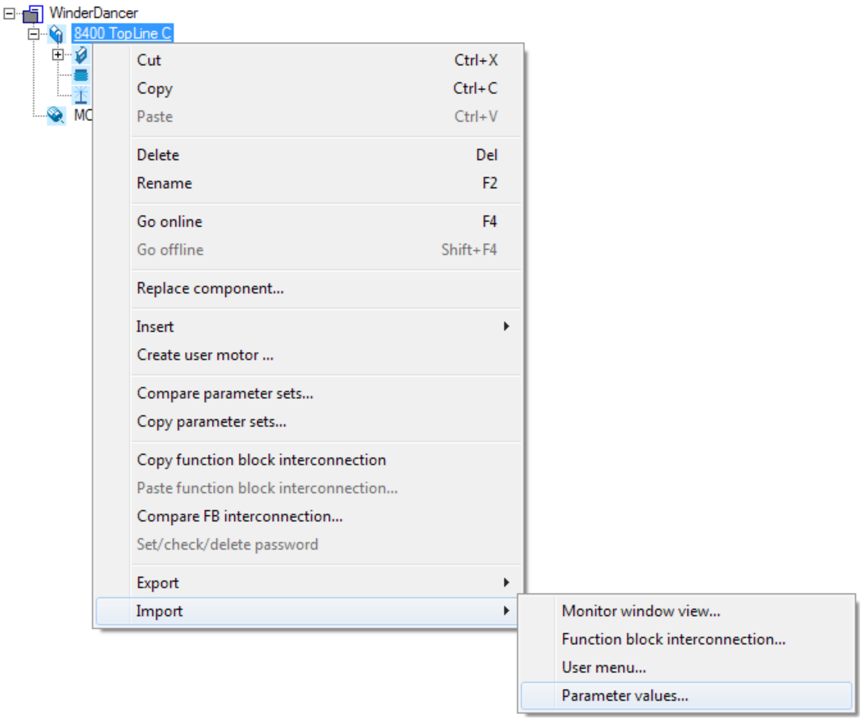
The maximum torque C00057 can only be read out online after having entered the motor model and after having adapted C00022 in the Inverter Drive 8400 TopLine.

	A	B	C	D	E	F	G	H
1	**Input Parameter**							
2	**Application**				**Motor**			
3	Maximum diameter	d_max	200	mm	Motor: Rated speed	n_motor	4050	rpm
4	Minimum diameter	d_min	50	mm	Motor: Rated power	P_motor	0,25	kW
5	Gearbox factor	i	1,0000		Motor: Rated current	I_motor	1,3	A
6	Reference line speed	v_100%	150,0	m/min	Maximum torque	C57	1,02	Nm
7	Minimum ramp-up time to reference line speed	t_acc	1,0	s	**Calculated values**			
8	Constant moment of inertia (Mechanic+Motor)	J_const	8,0	kgcm²	Motor: Rated torque	M_motor	0,6	Nm
9	Variable moment of inertia (Coil)	J_var	25,0	kgcm²	Max tension with dmax and rated motor torque	F_max	5,9	N
10	Maximum moment of inertia	J_max	33,0	kgcm²	Max tension with dmax and max torque C57	F_C57	10,2	N
11	Constant moment of inertia scaled to J_max	J_const_%	24,2	%	n_max / n_motor		0,236	
12					Required acceleration torque with J_max	M_acc	0,3	Nm
13								
14	**Calculated Values for Inverter**							
15	**Standard Parameter**					**Legend:**		
16	Motor: Reference speed	C11	955	rpm		Input		
17	Maximum diameter	C471/1	200	mm		Calculation		
18	Minimum diameter	C471/2	50	mm		Parameter for inverter		
19	Reference line speed	C471/3	1500	0,1m/min				
20								
21	**Acceleration compensation (optional)**							
22	C252 Gain L_DT1	C252	100					
23	L_ConvW Zähler	C940/2	1000					
24	Scaling factor Acceleration compensation	C940/4	330					
25		C941/4	1020					
26	L_Curve_2 x-Value_1	C1028/1	4096					
27	L_Curve_2 x-Value_2	C1028/2	4492					
28	L_Curve_2 x-Value_3	C1028/3	4889					
29	L_Curve_2 x-Value_4	C1028/4	5285					
30	L_Curve_2 x-Value_5	C1028/5	5682					
31	L_Curve_2 x-Value_6	C1028/6	6078					
32	L_Curve_2 x-Value_7	C1028/7	6474					
33	L_Curve_2 x-Value_8	C1028/8	6871					
34	L_Curve_2 x-Value_9	C1028/9	7267					
35	L_Curve_2 x-Value_10	C1028/10	7663					
36	L_Curve_2 x-Value_11	C1028/11	8060					
37	L_Curve_2 x-Value_12	C1028/12	8456					
38	L_Curve_2 x-Value_13	C1028/13	8853					
39	L_Curve_2 x-Value_14	C1028/14	9249					
40	L_Curve_2 x-Value_15	C1028/15	9645					
41	L_Curve_2 x-Value_16	C1028/16	10042					



How to import parameters in the »Engineer«:

1. Open the "Inertia_Export_Dancer" worksheet in the Excel tool.
2. Copy the content of cells E1 to E76 to a text editor using the Windows clipboard.
3. Save this text file and rename its file extension to ".gdc".
4. Click on the axis in the project tree using the left mouse button.
5. Select **Import → Parameter** values and select the file created.



3.10 Step 9 (optional): Optimising the closed-loop control parameters

3.10.1 Optimising the closed-loop speed control

In order to obtain a scaled setpoint speed for the speed controller, the scaled line speed signal is multiplied by the reciprocal diameter ($1/d_{act}$).

In order to ensure that the scaled setpoint winder speed complies with the scaled setpoint motor speed and the scaled line speed signal, carrying out the correct setting in C00011 (motor reference speed) is absolutely necessary. The scaled setpoint winder speed refers to the motor speed which, with the minimum diameter (d_{min}), is required to obtain the reference line speed at the circumference of the reel.

Adaptation of the speed controller gain

Under ideal circumstances, the gain of the speed controller must increase linearly with the moment of inertia effective at the motor shaft.

When the acceleration compensation has been parameterised correctly, the current moment of inertia is calculated automatically and is output at *L_ConW_4* as a scaled value. This signal can be used for the percentage speed controller adaptation.

Since, in practice, the gain must not increase proportionally with the moment of inertia, the process of adaptation is carried out via *L_GainOffsetP_1*:

- The scaled moment of inertia is multiplied by C00677/1.
- C00677/2 indicates the lower limit value of the speed controller gain.

In addition, the speed controller adaptation must be enabled with C00470/9 = TRUE.

The speed controller in the marginal areas should be optimised with a small and a very great diameter.

Online help for the Inverter Drive 8400 TopLine

In the "Optimising the speed controller" chapter, the general optimisation procedure is described.

3.10.2 Optimising the dancer position control

The control path of the dancer position control depends on many factors in the application and must therefore be optimised individually.

- Gain: C01056/1
- Reset time: C01053/4
- C00740/2 = TRUE (control bit 5 = TRUE) enables the PI controller. The influence is determined in C00472/7.

3.11 Step 10 (optional): Activating the web break monitoring function

For the purpose of diameter calculation, web break monitoring can be implemented using the *L_CalcDiameter_1* block. It is based on the fact that, in the event of a web break, the diameter calculated develops so that it opposes the winding direction.

The web break monitoring function is activated with C00470/7 = TRUE. This only makes a change in diameter opposite to the winding direction possible and permissible within the window parameterised in C01052.

The rewinding or unwinding operation is detected automatically, on the basis of the line speed sign and the winding direction parameterised. The response to this monitoring function is stored in the "Warning locked" Lenze setting and can be adapted in C00581/1.

Note!

If the web break monitoring function is active, a change in diameter opposing the winding direction specified via *bUnwindActive* is prevented. After loading a start diameter which significantly deviates from the real diameter in the direction opposing the winding direction, this may cause unintended triggering of the monitoring function.

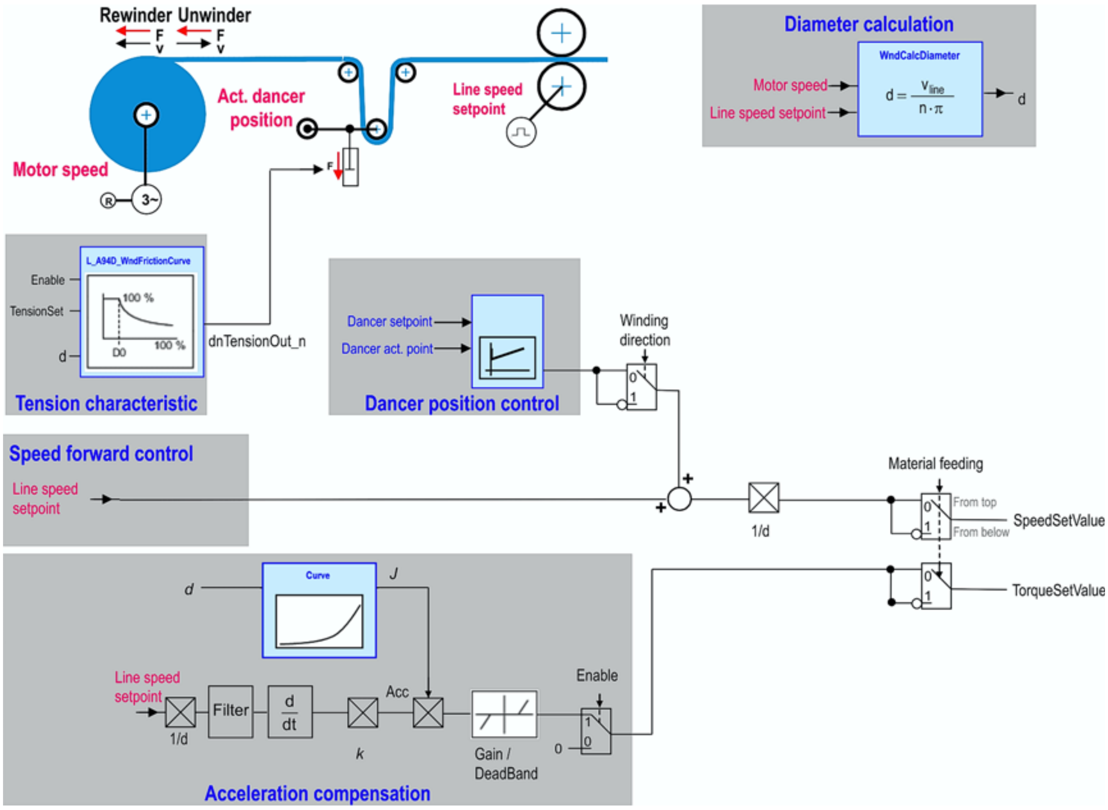
Example :

In the case of the rewinder, a start diameter of 50 % is loaded, however, the real diameter is only 45 %. The change in the diameter value to the real 45 % is prevented with an active web break monitoring function.

Web break monitoring may only be activated when the calculated diameter corresponds to the real diameter!

4 Detailed functions of the technology application

4.1 Signal flow of the technology application



4.2 Functions for winding operation

4.2.1 Speed feedforward control

In order to obtain a scaled winder setpoint speed for the speed controller, the scaled line speed signal is multiplied with the reciprocal diameter (1/d_{act}).

To make the scaled winder setpoint speed comply with the scaled motor setpoint speed and the scaled line speed signal, the matching setting in C00011 (motor reference speed) is absolutely required.

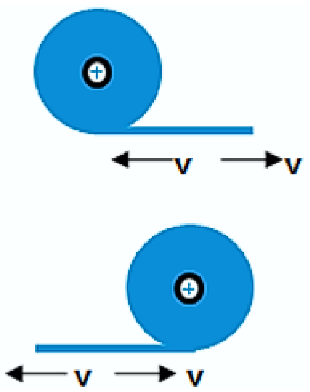
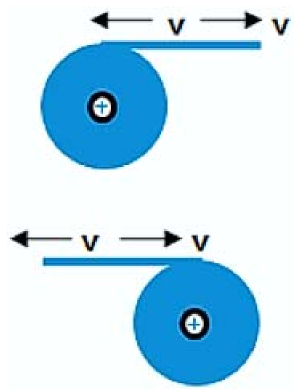
The scaled winder setpoint speed refers to the motor speed required with a minimum diameter d_{min} to achieve the reference line speed at the circumference of the reel.

▶ Adaptation of the speed controller gain (📖 39)
▶ Optimising the closed-loop speed control (📖 31)

4.2.2 Material feeding

The basic adjustment of the direction of rotation of the winder to the material flow is effected via parameter C01206/1 "motor mounting direction = inverted".

When the material is applied in an alternating fashion (from the top or from the bottom), the direction of rotation can be set via parameter C00470/2 or control bit 6. The signs of the speed follower setpoints are correspondingly inverted.

C00470/2 = 1 (material feeding from the bottom)	C00470/2 = 0 (material feeding from the top)
	

4.2.3 Inching mode

In order to traverse the winding shaft manually in inching mode, the generation of preset setpoints of the block *L_NSet_1* is used.

This automatically causes the main setpoint path to be decoupled via the line speed signal. This means that the winder can be moved independently of the speed-determining drive of the machine.

The overall setpoint is then evaluated with 1/d. Therefore parameterised setpoints in C00039/1 or C00039/2 for the positive and negative inching mode also refer to the circumferential speed or the line speed again and not to the motor speed.

4.2.4 Calculation of the diameter

The current diameter is calculated by dividing the line speed by the motor speed. The following relationship can be established between the diameter, circumferential speed, and winding shaft speed:

$$d[mm] = \frac{v_{line}\left[\frac{m}{min}\right]}{n_{Wickler}\left[\frac{1}{min}\right] * \pi} * 1000$$

For further information, please see the description for the *L_CalcDiameter* block.

Setting the diameter value (transferring the sensor signal)

At the start of a winding procedure, the start diameter must be defined. It can be set or gathered from the signal of a diameter sensor.

- Parameterising the start diameter: C00472/1
- Loading the start diameter: C00470/5 = TRUE

An external diameter value (e.g. from an ultrasonic sensor) can be transferred via the same setting function. The sensor signal can also be loaded permanently.

Holding the diameter

For some operating states of the winder, in which the line speed signal does not correspond to the circumferential speed of the reel, the current diameter cannot be calculated from the line speed and the motor speed. In this case, the calculation of new values must be prevented and the diameter must be held at the old value. This is done automatically if:

- the line speed is smaller than the minimum line speed from C00472/5;
- control bit 11 "Hold diameter" is TRUE;
- tension-controlled (open loop) operation is not enabled;
- the motor speed is lower than C00024.

4.2.5 Tension control open loop and winding characteristic

In the case of some winding materials it may be required to adapt the tensile force with the diameter becoming smaller. Otherwise the reel may drift to the sides, meaning it telescopes.

The diameter-dependent winding characteristic or tensile force characteristic is defined via a characteristic function in the *L_Curve_3* block. For more information please see the description for the *L_Curve_3* block.

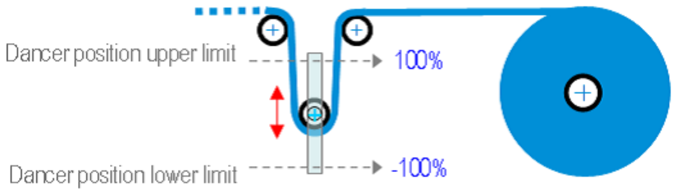
Note!

In the case of the dancer position control, the tensile force is not determined by the motor torque, but by the dancer actuator. This means that if the tensile force is to be controlled by the inverter, the inverter must be able to preselect the manipulating variable of the actuator via an analog output or via a fieldbus data word.

4.2.6 consideration of the dancer movement in the diameter calculation

In order to correct the dancer position, the circumferential speed of the winder must be faster or slower than that of the line. If the circumferential speed significantly increases or decreases compared to the line speed upstream of the dancer, the resulting circumferential speed $V_{Line;\ total}$ should be used for the diameter calculation. This is usually the case in applications storing longer material lengths in the dancer mechanism.

The speed resulting from the movement of the dancer can be determined from the differentiation of the dancer position. The maximum material length stored corresponds to a change of the analog dancer position of 200 %.



The storage volume for instance results from twice the distance between the two limit positions multiplied by the number of material wraps.

The dancer position signal is scaled with the *L_ConvActPos_1* block according to the upper and lower limit position. The block also determines the resulting circumferential speed at the reel. For more information please see the description for the *L_ConvActPos_1* block.

4.2.7 Monitoring of the dancer position

For the operation of the winder, monitoring of the dancer is of importance in more than one way.

- When the dancer position controller has been enabled, the machine should only be started if the dancer is in the set position. For this, the status signal bit 14 "bInSetPosition_b = TRUE" can be evaluated.
- If the dancer is in the lower limit position during ongoing operation for a longer time (e.g. 500 ms), it is likely that a web break has occurred. The response to this monitoring can be set in C00581/2 and C00581/3 (Lenze setting: warning).

4.2.8 Teach-in of the dancer limit positions

The position of the dancer is scaled to the upper or lower limit position. For this purpose, the teach-in function of the *L_ConvActPos* block is used.

For further information, please see the description for the *L_ConvActPos* block.

▶ Step 7 (optional): Teach-in of the dancer limit positions (📖 27)

4.2.9 PI controller for the dancer position control

When the *L_ProcessCTRL_1* process controller has been enabled, the dancer position is controlled. Use parameter C00472/7 to define the impact which the PI controller is to have on the motor control.

The analog signal of the current dancer position can be filtered with a PT1 characteristic. The filter time is set in C01053/2 (Lenze setting: 50 ms).

The I component of the controller can be reset via the *bIOff* input. This function is used in the sample project to deactivate the I component if the line speed is below the minimum line speed from C00472/5. In this way the integration of a system deviation is prevented if the winder is rotating although the line setpoint is still zero. This is for instance the case with the setting-up operation when first the rated tension via the winder is applied before the whole line starts.

After the dancer position control has been activated, the dancer usually has to be brought to the setpoint position first. In order that the dancer is lifted in a controlled fashion, the ramp generator for the position setpoint is previously loaded with the actual position value. This makes it unnecessary to show the impact of the dancer controller.

The PID controller that is used for dancer position control offers various possibilities of conditioning the setpoints and actual points.

For further information, please see the description for the *L_ProcessCTRL_1* block.

4.3 Disturbance compensation functions

4.3.1 Acceleration compensation

The acceleration in the line speed setpoint constitutes a disturbance in the winding process, because the torque that is "consumed" for the process of acceleration is missing in the tensile force. Therefore the acceleration torque must be calculated and pilot-controlled as an additional torque.

In this process, scaled values are used for calculation. The additional scaling to the acceleration time is to improve the resolution of the acceleration torque.

In practice, a line speed signal that does not ideally increase is to be expected. Use parameter C00253 to set the resolution of the signal that is differentiated.

Only the number of higher-order bits that is set is taken into consideration for the differentiation. Furthermore the signal is smoothed via a PT1 functionality subsequently. The time constant can be set via parameter C00251.

The gain of the DT1 element and the scaling of the *L_ConvW_2* are used for taking the acceleration time *t_acc* into consideration.

The mass inertia of the drive results from the current diameter by a characteristic in which the moment of inertia scaled to the maximum moment of inertia *J_max* as a function of the diameter scaled to *d_max* is stored. Then the moment of inertia is evaluated with *M_acc / M_max* in order to achieve an improved scaling of the acceleration.

The moment of inertia is multiplied with parameter C00741/8 in order to provide for different material widths with all other settings remaining consistent.

Note!

The speed is resolved with 16 bits in the Inverter Drive 8400, 100 % = 2^14 = 16384 corresponding to the rated speed (C00011).

The rated speed is to be selected so that it is obtained with a minimum diameter at a line speed of 100 %. In the case of slow acceleration processes, the limited speed resolution produces quantisation steps of the same size as the change in setpoint values. Therefore, in particular for slow acceleration processes (>10 s), the feedforward control value is to be checked and deactivated, if necessary.

A greater filter time constant in the PT1 filter of the acceleration (C00250) can bring about an improvement.

If possible, the line speed signal should be derived from a stable setpoint instead of from an actual speed value, in order to obtain a utilisable acceleration feedforward control.

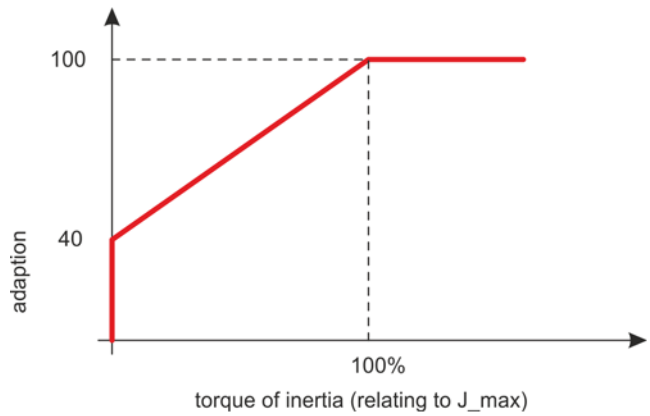
4.4 Adaptation of the speed controller gain

If we consider the motor and the reel as a rigid one-mass system, the gain of the speed controller is directly proportional to the moment of inertia.

However, since the moment of inertia usually changes significantly during the winding process, a good control response may require continuous adaptation of the speed controller gain to the moment of inertia.

The adaptation of the speed controller gain can only be used reasonably if the acceleration compensation is configured (J characteristic stored) so that the calculation of the current moment of inertia supplies correct values.

In the default setting, the speed controller is adapted linearly in the range from 40 ... 100 % of the moment of inertia. Below this range, the gain is held constant.



FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

8400 "Winder Dancer-controlled" technology application · SHG84WindDancer · 13538921 · DMS 1.0 EN · 07/2017 · TD29

Lenze Drives GmbH
Postfach 10 13 52, D-31763 Hameln
Breslauer Straße 3, D-32699 Extertal
Germany
HR Lemgo B 6478
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Inverter

8400

"Winder Tension-controlled" technology application
for 8400 TopLine C

Software Manual EN

13538923

Lenze

Contents

1 About this documentation

This documentation describes the software-based solution of a task. The transferability of the described solution to the respective application needs to be checked by the user. If required, the user has to adapt the solution accordingly. Physical aspects such as the drive dimensioning are not part of this documentation.

Danger!

The inverter is a source of danger which may lead to death or the severe injury of persons.

To protect yourself and others against these dangers, observe the safety instructions before switching on the inverter.

Please read the safety instructions provided in the **Inverter Drives 8400 mounting instructions** and in the **Inverter Drives 8400 hardware manual**. Both documents are supplied with the inverter.

Target group

This documentation addresses to all persons who ...

- want to use the "Winder Tension-controlled" technology application for the Inverter Drive 8400 TopLine and ...
- who are familiar with handling the device and the »Engineer« software.

Information regarding the validity

The information in this documentation applies to the following technology applications:

Technology application	From version
Winder Tension-controlled	1.0

Screenshots/application examples

All screenshots provided in this documentation are application examples. Depending on the software version of the inverter and the version of the »Engineer« software installed, the screenshots in this documentation may differ from the representation in the »Engineer«.

Tip!

Information and tools for Lenze products are provided in the Download area at

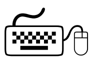
http://www.lenze.com → Download

1.1 Document history

Version			Description
1.0	07/2017	TD29	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the "0x" prefix is used. Example: 0x60F4
Binary number	0b	For binary numbers, the "0b" prefix is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that only applies to a certain inverter software version or higher is identified accordingly in this documentation. Example: This function extension is available from software version V3.0 onwards!
Program name	» «	The »Engineer«... Lenze PC software
Window	*italics*	The *Message window*... / The dialog box *Options*...
Variable names		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button ... / The **Copy** command ... / The **Properties** tab ... / The **Name** input field ...
Sequence of menu commands		If several commands must be used in sequence to carry out a function, the individual commands are separated by an arrow: Select **File**➞**Open** to...
Shortcut	**<bold>**	Use **<F1>** to open the online help.
		If a shortcut is required for a command to be executed, a "+" has been put between the key identifiers: With **<Shift>+<ESC>** ...
Hyperlink	underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(🕮 4)	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Engineering tools	Software solutions for easy engineering in all project stages
	»EASY Navigator« – ensures easy operator guidance • All convenient Lenze engineering tools at a glance • Tools can be quickly selected • The clear structure simplifies the engineering process from the start
	»EASY Starter« – simple tool for service technicians • Specially designed for the commissioning and maintenance of Lenze devices • Graphical user interface with only a few buttons • Simple online diagnostics, parameterisation, and commissioning • No risk of an unintended change in applications • Loading of ready-to-use applications to the device
	»Engineer« – multi-drive engineering • For all products in our L-force portfolio • Practical user interface • Easy handling by graphical user interfaces • Can be applied in every phase of a project (project planning, commissioning, production) • Parameter setting and configuration
Code	Parameter used for inverter parameterisation or monitoring. Is usually referred to as "index".
Subcode	If a code contains several parameters, they are stored in "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). Is usually referred to as "subindex".
Lenze setting	This setting is the default factory setting of the device.
FB Editor	Abbreviation for "function block editor". Graphic interconnection tool which is available in the »Engineer« Abbreviation for "function block editor". Graphic interconnection tool which is available in the **FB Editor** tab.
Function block	General designation of a function block for free interconnection in the FB Editor. A function block (short: "FB") can be compared to an integrated circuit which contains a specific control logic and supplies one or more values when it is executed. Example: "L_Arithmetic_1" (FB for arithmetic operations) Many function blocks are available several times (e.g. L_And_1, L_And_2, and L_And_3).
System block	In the function block editor of the »Engineer«, system blocks provide interfaces to basic functions, "free codes", and to the hardware of the inverter (e.g. to the digital inputs). Each system block is available only once.
Port block	Block for implementing the process data transfer via a fieldbus
LP	Abbreviation for **L**enze **P**ort block Example: "LP_CanIn1" (CAN1 port block)
LS	Abbreviation for **L**enze **S**ystem block Example: "LS_DigitalInput" (system block for digital input signals)
MCI	Abbreviation for **M**otionbus **C**ommunication **I**nterface (fieldbus interface) The Inverter Drives 8400 can accommodate plug-in communication modules and can therefore take part in the data transfer of an existing fieldbus system.
Technology application	A technology application is a drive solution based on Lenze's experience and know-how, in which function blocks interconnected to a signal flow form the basis for implementing typical drive tasks.
USB diagnostic adapter	The USB diagnostic adapter is used for the operation, parameterisation, and diagnostics of the inverter. Data are exchanged between the PC (USB connection) and the inverter (diagnostic interface on the front) via the diagnostic adapter. Order designation: E94AZCUS

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

> **Pictograph and signal word!**
>
> (characterise the type and severity of danger)
>
> **Note**
>
> (describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling

2 Properties of the technology application (TA)

With the "Winder Tension-controlled" technology application, a tension-controlled (open loop) or tension-controlled (closed loop) winding drive can be implemented on an Inverter Drive 8400 TopLine **from V14.00 onwards**.

2.1 Control modes for winding processes

The following technology applications (TA) designed by Lenze can be used for winding processes:

"Winder Dancer-controlled" TA

With **dancer position control**, the drive is operated in the speed control mode. For feedforward control, the line speed signal multiplied by the reciprocal value of the diameter is used. The dancer position is recorded and compared with the setpoint position. If there is a deviation, the dancer position controller corrects the speed setpoint.

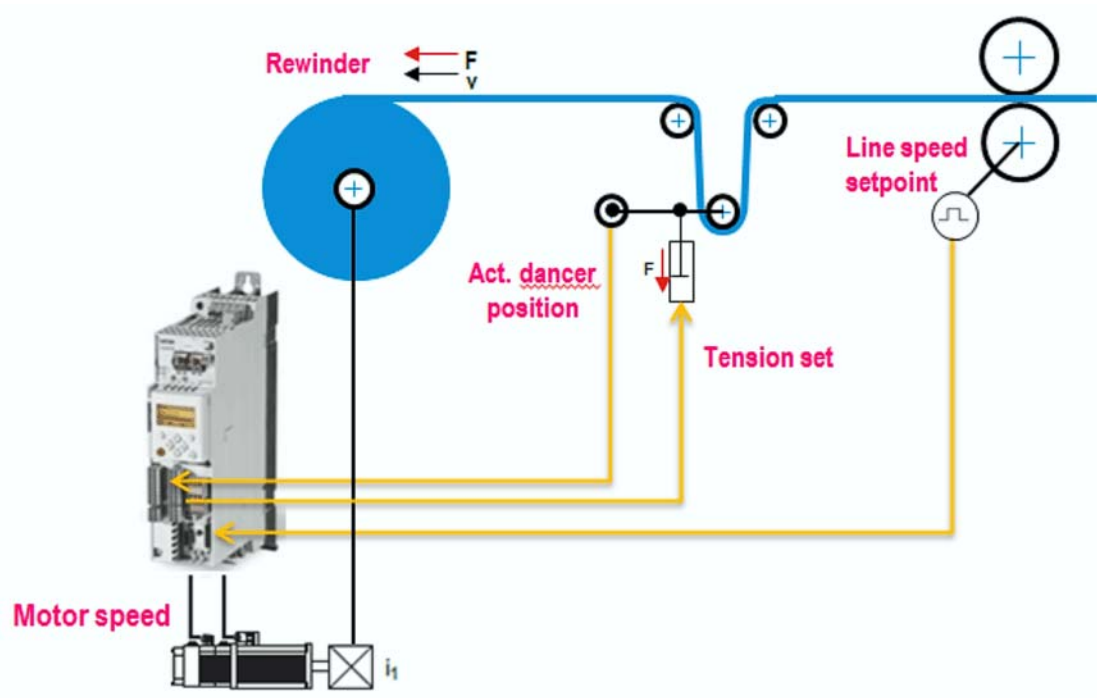
"Winder Tension-controlled" TA

With **tension control open loop/closed loop**, the drive torque is directly provided. A higher-level speed control only takes corrective action in the event of a web break, in order to limit the speed of the drive. To prevent the setpoint torque from being affected by the speed limitation during normal operation, a speed offset must be added to the speed setpoint calculated from the current line speed and the current diameter. The torque setpoint is composed of the tensile force setpoint multiplied by the current radius, the correcting signal for the compensation of the mechanical friction and the correcting signal for the compensation of the acceleration torque.

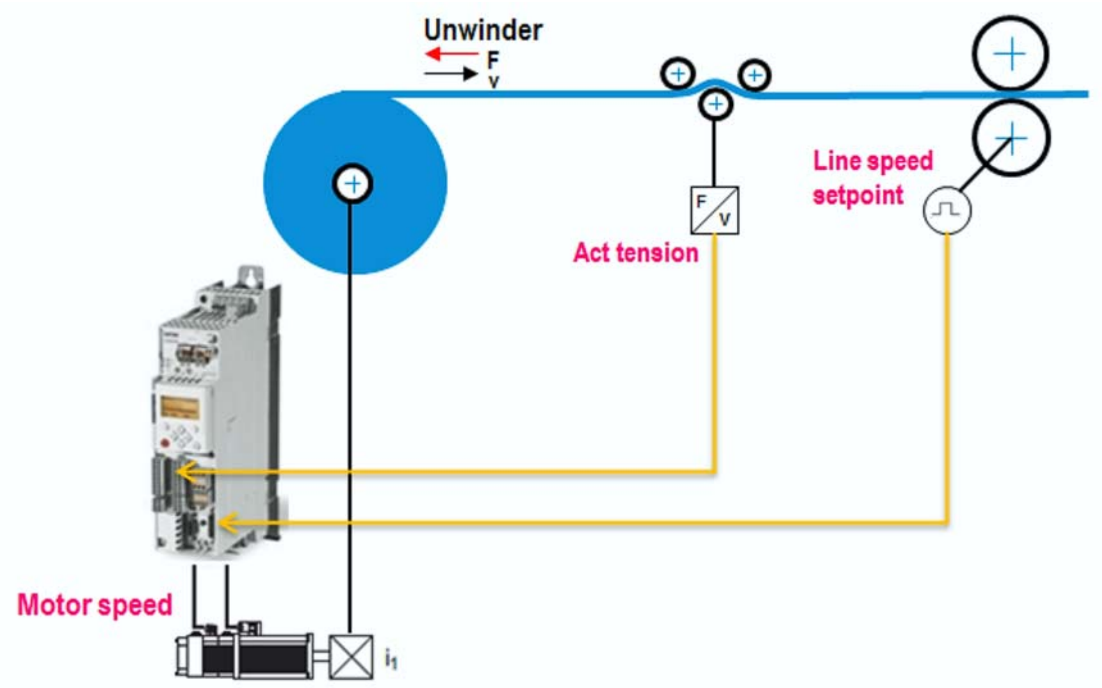
2.1.1 Comparison of the functional principles

Winder Dancer-controlled **-> dancer position control**	**Winder Tension-controlled** **-> tension control open loop/closed loop**
• The web tension force is determined by the dancer mechanics.	• Depending on the tensile force setpoint and the current diameter of the winding material, the drive torque directly defines the tensile force at the winding material.
• The drive is actuated in speed control mode.	• The drive is operated as torque actuator.
• The line speed signal multiplied by the reciprocal value of the winding material diameter is used for feedforward control. • The dancer position is recorded and compared to the setpoint position. If a deviation is detected, the dancer position controller corrects the speed setpoint. • An acceleration torque can be compensated. • A dancer actuator can be optionally controlled from the inverter, e.g. in order to implement a reduction in tensile force with an increasing diameter.	• The line speed signal multiplied with the reciprocal value of the reel diameter + offset serves as a limit for speed limitation. • If an optional tensile force recognition system is provided, deviations can be compensated using a process controller (closed loop). • An acceleration torque and the friction losses can be compensated.

• The diameter is calculated in the inverter.
• The control and setpoint selection are optionally taken over by ...
  • an HMI.
  • the digital and analog interfaces of the inverter.
  • a higher-level control via a bus system.	



[2-1] Functional principle of a dancer position control ("Winder Dancer-controlled" TA)



1

[2-2] Functional principle of a tension control open loop/closed loop ("Winder Tension-controlled" TA)

2.1.2 Functional overviews

<table>
<tr><th>Winder Dancer-controlled
-> dancer position control</th><th>Winder Tension-controlled
-> closed loop/open loop tension control</th></tr>
<tr><td colspan="2">• Inching mode with ramp generator
• Diameter calculation from line speed and winding speed
• Holding/setting the diameter value
• Web break monitoring using the diameter calculator
• Winding from the top or the bottom
• Automatic detection of the winding direction (unwinding/rewinding) by means of the sign of the line speed</td></tr>
<tr><td>• If the dancer actuator is controlled from the inverter: reduction of tensile force via the characteristic function for rewinders (winding characteristic)</td><td>• Reduction of tensile force via the characteristic function for rewinders (winding characteristic)</td></tr>
<tr><td>• Acceleration compensation</td><td>• Acceleration compensation
• Friction compensation
• Ramp generator for tensile force setpoint
• Speed limitation via the line speed plus the offset, 1/d-evaluated</td></tr>
<tr><td>• PI controller dancer position control with various possibilities of adaptation
• "Teach-In" function for the dancer limit positions
• consideration of the dancer movement in the diameter calculation</td><td>• PI controller tension control with various possibilities of adaptation</td></tr>
<tr><td colspan="2">• Adaptation of the speed controller gain as a function of the current moment of inertia</td></tr>
</table>

In the following, the "Winder Tension-Controlled" technology application is described. A description of the "Winder Dancer-Controlled" TA can be found in the separate software manual of the TA.

2.2 Application ranges

- Rewinding and unwinding of material webs in the field of surface finishing
- Winding facilities in the textile industry
- Unwinders in packaging machines
- Rewinders/unwinders of round materials (threads, wires, cables, tubes,...)

... and many things more.

2.3 System requirements

Software

Product	Order designation	From version
»Engineer«	ESPEV-EHNNN	2.22.1.0

Hardware

Product	Order designation	From hardware version	From software version
Inverter Drive 8400 TopLine	E84AVTCxxxxx	VD	14.00

2.4 Parameter setting in the function block editor view

Enter application-specific parameters directly in the function block editor (FB Editor). In this way, the signal flow is traceable and the interaction of the blocks is illustrated.

Furthermore you can use the FB Editor to configure the I/O interconnection and monitor the applications running on the device, e.g. for diagnostics purposes.

- Open the parameterisation dialog or the parameter list for the block ...
  - via the icon in the block header,
  - by double-clicking the block, or
  - by executing the **Parameter...** command in the *context menu* for the block.
- Colour coding and coloured comments provide for a clearly arranged structure of the FB Editor.
  - The areas highlighted in turquoise represent the "user interface". If required, the pre-assignment of the I/O terminals can be adapted here and a control via the fieldbus interface (MCI) can be established.
  - In the areas highlighted in yellow, application-specific settings are required.

Reference manual / online help for the Inverter Drive 8400

In the "Working with the FB Editor" chapter you'll find some detailed information on how to handle the FB Editor.

2.5 Pre-assignment of the user interface

Terminal	Function		
Digital input terminals			
X5/RFR	Controller enable		
	RFR		Function
	LOW		Inhibit drive
	HIGH		Enable drive
X5/DI1	- (reserved for HTL encoder)		
X5/DI2	- (reserved for HTL encoder)		
X5/DI3	Enable winder (follow line speed)		
	DI3		Function
	LOW		Inhibit winder
	HIGH		Enable winder
X5/DI4	Enable dancer control		
	DI3		Function
	LOW		Inhibit dancer control
	HIGH		Enable dancer control
X5/DI5 **X5/DI6**	Selection of a preset setpoint for inching mode. At activation, an additional acceleration time is activated; positive manual inching		
	DI5	DI6	Function
	LOW	LOW	Follow the line speed
	HIGH	LOW	Selection of preset setpoint 1 = C00039/1 = 10 % • "Positive inching
	LOW	HIGH	Selection of preset setpoint 2 = C00039/2 = -10 % • "Negative inching
	HIGH	HIGH	Selection of preset setpoint 3 = C00039/3 = 0 %
X5/DI7	HIGH = reset error		
Analog input terminals			
X3/A1U	Line velocity • Scaling: 10 V = 100 % reference line speed (C0471/3)		
X3/A2U	Actual value of dancer position • Scaling: 10 V = 100 % reference)		
Digital output terminals			
X4/DO1	HIGH ≡ "Drive is ready" state		
X4/DO2	Not assigned, can be freely used		
X4/DO3	Not assigned, can be freely used		
X107/BD1, BD2	Control of a holding brake by the basic function "holding brake control"		
X101/COM, NO	Relay contact closed ≡ "An error is pending" state		
Analog output terminals			
X3/O1U	Actual speed value • Scaling: 10 V ≡ 100 % reference speed (C00011)		
X3/O2U	Actual torque value • Scaling: 10 V ≡ 100 % maximum torque (C00057)		

3 Short setup of the technology application

3.1 Preconditions

For the execution of the short setup described in the following, it is assumed that the basic parameters (motor, feedback system, etc.) have been set.

The "8400 commissioning wizard" enables guided commissioning of the Inverter Drive 8400, taking the Lenze parameter setting as a basis.

How to proceed:

1. Before switching on: ensure that the inverter is inhibited (digital input terminal X5/RFR open).
2. Switch on voltage supply of the inverter.
   The parameterisation and diagnostics of the inverter without motor operation solely requires an external 24 V supply by a safely separated power supply unit (SELV/PELV).
3. Establish a communication link between the inverter and the Engineering PC, e.g. via a USB diagnostic adapter (E94AZCUS):
   - connect the USB diagnostic adapter to the X6 diagnostic interface.
   - establish a connection between the USB diagnostic adapter and the PC via a free USB port.
4. Start the »Engineer« on the Engineering PC, e.g. via the Windows start menu:
   Start → All programs → Lenze → Engineering → L-force Engineer...
   After the program start, no project has been loaded first and the *start-up wizard* is displayed.

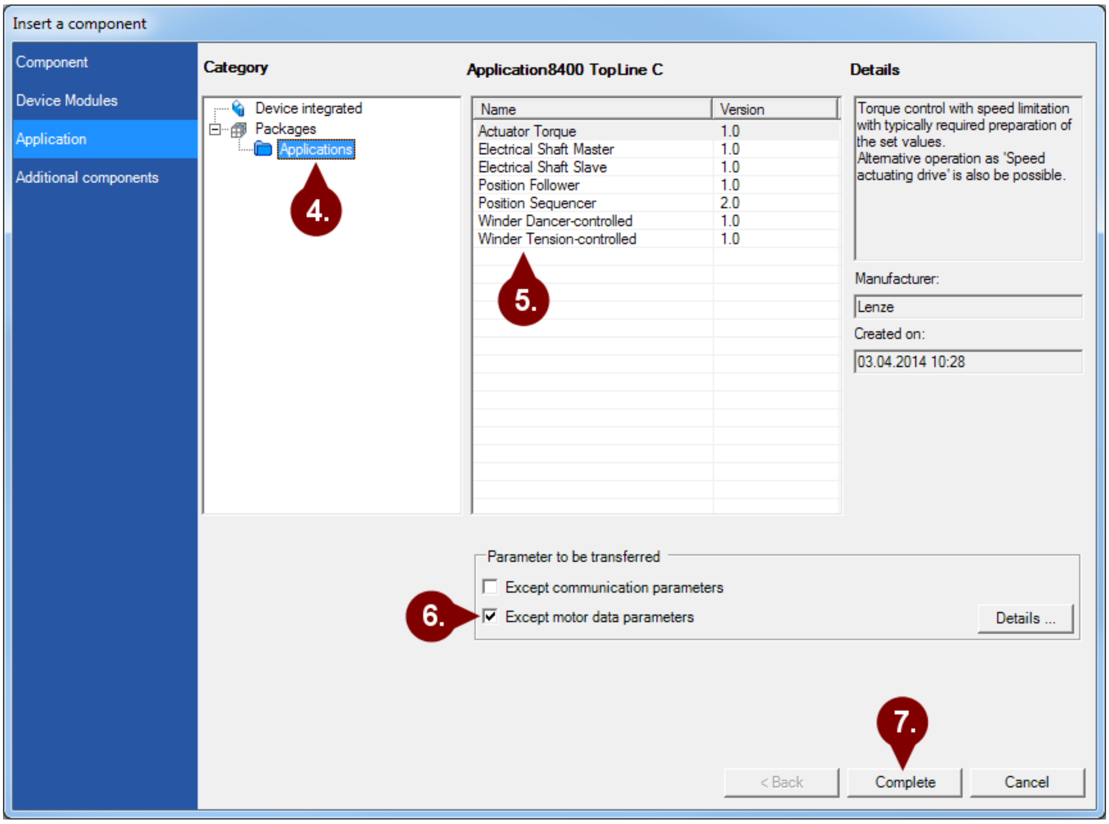
> **»Engineer« Online help**
> Here you'll find some detailed information with regard to the options of the start-up wizard and to the general handling of the »Engineer«.

5. Create a new project or open a project already available.
6. Go to the *Project view* and select the 8400 inverter.
7. Click the icon to go online.
   When the connection to the inverter has been established successfully, the *status line* shows ONLINE .
8. Click on the icon to start the "8400 commissioning wizard".
   - Now the commissioning wizard guides you step by step through the setting of the important parameters for a quick commissioning.
   - The **Next** button can only be activated again after all parameter settings in the device have been reset via the **Load Lenze setting** button.
   - Execute the commissioning wizard right up to the end.
   - You can skip the "Control mode" step by clicking **Next** (only relevant for "Speed actuating drive" technology application).

3.2 Step 1: Load "Winder Tension-Controlled" technology application

In the Lenze setting, the inverter uses the "Actuator speed" technology application integrated in the device. Execute the following steps to use the "Winder Tension-Controlled" technology application instead:

1. Go to the *Project view* and select the inverter.
2. If there is still an online connection to the inverter, click on the icon to go offline.
   The application can only selected when you are offline.
3. Click the icon to select another application.
   The *Insert application* dialog box appears:



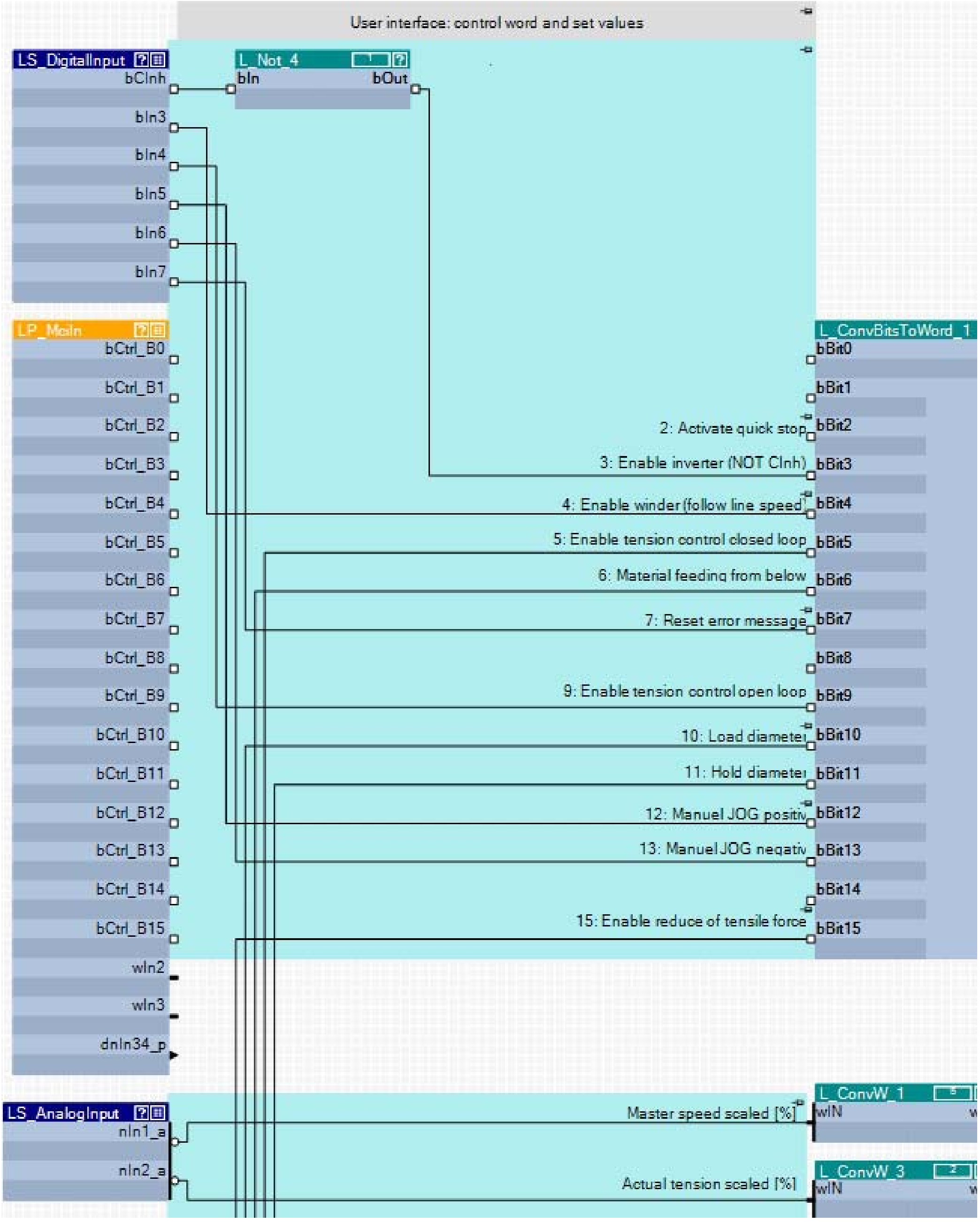
4. In the left field, select the **Packages → Applications** category.
5. In the right field, select the **"Winder Dancer-controlled"** application.
6. Activate the **Except motor data parameters** option in order that the settings of the motor data parameters made before will not be overwritten.
7. Press **Complete** to close the dialog box again and load the application selected into the »Engineer« project.
8. Confirm the prompt on whether the current application is to be replaced by the new application with **Yes**.

3.3 Step 2 (optional): Set up control via the fieldbus interface (MCI)

The Lenze default setting provides for controlling the application via the digital input terminals and specifying setpoints via the analog input terminals. If this is desired, continue with step 3.

If the fieldbus interface (MCI) is to be used instead of the terminals, the user interface in the FB Editor (area highlighted in turquoise) is to be adapted accordingly for the application control word and the setpoints.

- The assignment of the outputs (on the left) to the inputs (on the right) can be changed at will.
- The inputs (on the right) are permanently linked to functions of the application.



[3-1] Example of an interconnection

3.3.1 Pre-assignment of the process data input words

In the model connection (see Fig. [3-1]), the process data input words are assigned as follows:

Input word	Assignment
Word 1	Control word (for bit assignment see the following table)
Word 2	Line velocity • Scaling: 16384 ≡ 100 % reference speed C00011
Word 3	Tensile force setpoint • Scaling: 16384 ≡ 100 % tensile force
Word 4 ... 16	Not preconfigured

<table>
<tr><th>Control word</th><th colspan="3">Function</th></tr>
<tr><td>Bit 0</td><td colspan="3">Not preconfigured</td></tr>
<tr><td>Bit 1</td><td colspan="3">Not preconfigured</td></tr>
<tr><td>Bit 2</td><td colspan="3">1 ≡ Activate quick stop (QSP)</td></tr>
<tr><td>Bit 3</td><td colspan="3">1 ≡ Enable inverter (RFR)</td></tr>
<tr><td>Bit 4</td><td colspan="3">1 ≡ enable winder</td></tr>
<tr><td>Bit 5</td><td colspan="3">1 ≡ enable tension control</td></tr>
<tr><td>Bit 6</td><td colspan="3">0 ≡ material feeding from the top
1 ≡ material feeding from the bottom</td></tr>
<tr><td>Bit 7</td><td colspan="3">1 ≡ reset error (Trip reset)</td></tr>
<tr><td>Bit 8</td><td colspan="3">Not preconfigured</td></tr>
<tr><td>Bit 9</td><td colspan="3">1 ≡ enable tension control open loop</td></tr>
<tr><td>Bit 10</td><td colspan="3">1 ≡ load diameter</td></tr>
<tr><td>Bit 11</td><td colspan="3">1 ≡ hold diameter</td></tr>
<tr><td rowspan="6">Bit 12 ... 13</td><td colspan="3">Selection of preset setpoints for inching mode</td></tr>
<tr><td>Bit 12</td><td>Bit 13</td><td>Function</td></tr>
<tr><td>0</td><td>0</td><td>Inching mode not active</td></tr>
<tr><td>1</td><td>0</td><td>Selection of fixed setpoint 1 = C00039/1 = 10 % reference speed (C00011)
-> manual inching in positive direction</td></tr>
<tr><td>0</td><td>1</td><td>Selection of fixed setpoint 2 = C00039/2 = -10 % reference speed (C00011)
-> manual inching in negative direction</td></tr>
<tr><td>1</td><td>1</td><td>Selection of fixed setpoint 3 = C00039/3 = 0 % reference speed (C00011)</td></tr>
<tr><td>Bit 14</td><td colspan="3">Not preconfigured</td></tr>
<tr><td>Bit 15</td><td colspan="3">1 ≡ enable reduction in tensile strength for rewinder (tension characteristic)</td></tr>
</table>

3.3.2 Pre-assignment of the process data output words

In the model connection (see Fig. [3-1]), the process data output words are assigned as follows:

Output word	Assignment
Word 1	Status word (for bit assignment see the following table)
Word 2	Actual speed value • Scaling: 16384 = 100 % reference speed C00011
Word 3	Current diameter • Scaling: 16384 = 100 % maximum diameter C00471/1
Word 4 ... 16	Not preconfigured

Status word	Status					
Bit 0	1 ≡ group error active (configurable in C00148)					
Bit 1	1 ≡ inverter control is inhibited (pulse inhibit is active)					
Bit 2	1 ≡ inverter is ready for operation					
Bit 3	1 ≡ quick stop is active					
Bit 4	1 ≡ setpoint torque is in limitation					
Bit 5	1 ≡ speed controller is in limitation					
Bit 6 **Bit 6**	During open-loop operation: 1 ≡ speed setpoint < comparison value (C00024)					
	During closed-loop operation: 1 ≡ actual speed value < comparison value (C00024)					
Bit 7	1 ≡ inverter is inhibited (controller inhibit is active)					
Bit 8 ... 11	Bit 11	Bit 10	Bit 9	Bit 8	Device status	Meaning
	0	0	0	0	FirmwareUpdate	Firmware update function is active
	0	0	0	1	Init	Initialisation active
	0	0	1	0	Ident	Identification active
	0	0	1	1	ReadyToSwitchOn	Device is ready to start
	0	1	0	0	SwitchedOn	Device is switched on
	0	1	0	1	OperationEnabled	Operation
	0	1	1	0	-	-
		1	1	1	Trouble	Trouble active
	1	0	0	0	Fault	Fault active
	1	0	0	1	TroubleQSP	TroubleQSP is active
	1	0	1	0	SafeTorqueOff	Safe torque off is active
	1	0	1	1	SystemFault	System fault active
Bit 12	1 ≡ a warning is indicated					
Bit 13	1 ≡ inverter is in the "Trouble" state					
Bit 14	1 ≡ tension-controlled (open loop) operation active					
Bit 15	1 ≡ web break (implausible diameter change)					

3.4 Step 3: Set commissioning parameters

Open the parameterisation dialog or the parameter list for the block ...

- via the icon in the block header,
- by double-clicking the block, or
- by executing the **Parameter...** command in the *Context menu* for the block.

For quick commissioning, only the following application-specific parameters have to be set or their default setting has to be checked:

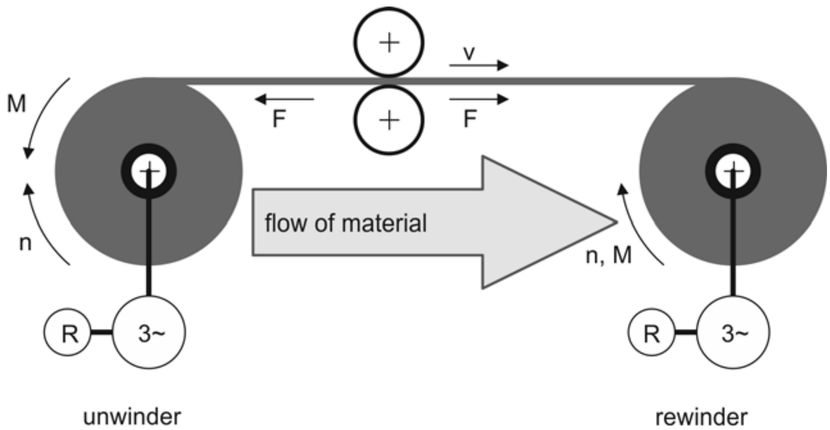
- "Normal" winding direction (rewinder or unwinder)
- Winding from the top or the bottom
- Reference variables

In the following, these parameters are described in detail.

3.4.1 Winding direction (rewinder or unwinder)

To ensure that the feedforward control values, the disturbance compensation, and the correcting signal of the tension control always act in the correct direction, the "normal" winding direction must be defined once.

- If the drive **unwinds** the material while the line speed is positive, **C00470/1 = 1** (unwinder) must be set.
- If the drive **rewinds** the material while the line speed is positive, **C00470/1 = 0 (**rewinder) must be set.



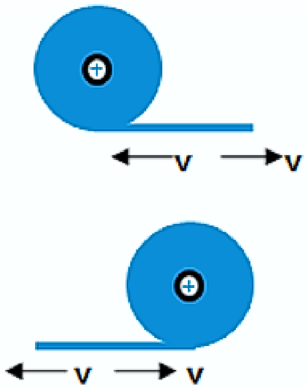
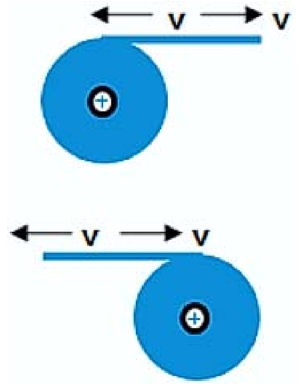
Note!

When the "normal" winding direction has been defined once, the winding drives can also run in the opposite direction, with a negative line speed. Intervening in the signal flow is not necessary for this.

By means of the parameter C01206/1, "motor mounting direction = inverted", a potential inversion of the direction of rotation due to uneven-numbered gearbox stages and/or an inverted motor mounting position can be eliminated.

3.4.2 Material feeding (winding from the top or the bottom)

Define the material feeding (effective direction of speed and torque at the motor shaft) in code C00470/2 or using control bit 6.

C00470/2 = 1 (material feeding from the bottom)	C00470/2 = 0 (material feeding from the top)
	

3.4.3 Defining reference variables

The setpoints for the speed, line speed, and tensile force are processed as scaled values. Therefore the reference variables have to be parameterised accordingly.

The reference variable for all scaled speeds is C00011. This value must be parameterised so that it matches the present machine data (gearbox factor i, minimum diameter d_{min}) and the reference line speed v_100 %.

The reference value for the tensile force is a reference force F_100% provided by the application. The torque setpoint refers to C00057 and is calculated online in the inverter by entry of the motor parameters and C00022.

F_100% is to be selected so that when this tensile force is provided at the motor and with a maximum diameter, still a minimum reserve torque of 10 % to the maximum possible value (C00057) is provided. This reserve is to be kept as a control margin for the acceleration in the friction compensation process. Depending on the actual requirements of the application, also greater reserves may be required here.

The tensile force is adapted to the torque in the inverter.

Conversion factor: C0471/4 / C00471/5

(These factors are calculated from F_100% and C00057 in the Excel support tool designed by Lenze, "Tool_TA8400_TensionControlled_V1-0.xlsx".)

The reference variables are preselected via the basic parameters described in the following.

Note!

The **basic parameters** must be defined at the start of commissioning.

The **other parameters** can be adapted as optimisation parameters in the course of commissioning.

The Lenze settings are selected so that the technology applications can be tested on the Lenze winder training model:

- Left -> "Winder Tension-controlled" TA as unwinder
- Right -> "Winder Dancer-controlled" TA as rewinder

Parameter (block)	Possible setting		Info
Basic parameter			
C00011	50 ... 60000		Motor speed at reference line speed and minimum diameter C00011 = ROUNDS(i/(d_min/1000*π)*v_100 %) i: total ratio between motor shaft and winding shaft d_min: minimum diameter [mm] v_100 %: reference line speed [m/min] **Please note:** This value must be defined accordingly and adapted when v_100%, i or d_min is changed. • Lenze setting: 955 rpm
C00470/1 (LS_ParFree_b)			Function of the winder with a positive line speed signal • Lenze setting: 1
	0	Rewinder	
	1	Unwinder	
C00470/2 (LS_ParFree_b)			Material feeding • Lenze setting: 1
	0	From the top	
	1	From the bottom	
C01202/1 (LS_MotionControlKernel)	1 ... 65535		Gearbox factor numerator Z2 • Lenze setting: 1
C01202/2 (LS_MotionControlKernel)	1 ... 65535		Gearbox factor denominator Z1 • Lenze setting: 1
C00471/1 (LS_ParFree)	1 ... 65535		Maximum diameter [mm] • Lenze setting: 200
C00471/2 (LS_ParFree)	1 ... 65535		Minimum diameter [mm] • Lenze setting: 50
C00471/3 (LS_ParFree)	1 ... 65535		Reference line speed [0.1 x m/min] • Lenze setting: 1500
C00471/4 C00471/5 (LS_ParFree)	1 ... 65535		Conversion factor tensile force to torque C00471/4 = numerator = F_100% x 10 C00471/5 = denominator = d_max / 1000 x C00057 x 10 F_100 % = rated tensile force [Nm] d_max = maximum diameter [mm] C00057 = maximum torque (online value according to corresponding parameterisation of the motor and C00022) • Lenze setting: C00471/4=50; C00471/5=102
C00472/5 (LS_ParFree)	-199.99 % ... 199.99 %		Threshold of minimum line speed • Lenze setting: 1 %
C00472/6 (LS_ParFree)	-199.99 % ... 199.99 %		Comparison value of the line speed sign recognition • Lenze setting: -0.2 %

Parameter (block)	Possible setting		Info
Condition speed setpoint /limitation threshold values; inching mode			
C00470/9 (LS_ParFree_b)			Adaptive adjustment of the speed controller gain **Please note:** requires a correspondingly parameterised acceleration compensation. • Lenze setting: 0
	0	Adaptation active	The speed controller gain is set to a permanent value (C00070).
	1	Adaptation not active	The speed controller gain is increased linearly to the current moment of inertia.
C00660/2 (L_FixSet_a_1)	-199.99 % ... 199.99 %		Line speed offset for the speed limitation with a positive line speed signal Lenze setting: 5 %
C00660/4 (L_FixSet_a_1)	-199.99 % ... 199.99 %		Line speed offset for the speed limitation with a negative line speed signal • Lenze setting: 5 %
C00721/2 (L_DigitalDelay_2)	0.00 ... 3600.00 s		Delay of the deceleration time deactivation for the inching mode • Lenze setting: 0.1 s
C00039/1	-199.99 % ... 199.99 %		Preset setpoint 1: positive inching • Lenze setting: 10 %
C00039/2	-199.99 % ... 199.99 %		Preset setpoint 2: negative inching • Lenze setting: -10 %
C00039/3	-199.99 % ... 199.99 %		Preset setpoint 3: alternative inching speed • Lenze setting: 5 %
C00101/1	0.001 ... 999.99 s		Inching mode: acceleration time • Lenze setting: 5 s
C00101/3	0.001 ... 999.99 s		Inching mode: deceleration time • Lenze setting: 5 s
C00677/1	-199.99 % ... 199.99 %		Adaptation factor of the moment of inertia for the speed controller adaptation • Lenze setting: 60 %
C00677/1	-199.99 % ... 199.99 %		Lower limitation of the speed controller adaptation • Lenze setting: 40 %

Parameter (block)	Possible setting		Info
Tensile force open loop control/closed loop control, tensile force characteristic, tensile force controller			
C00470/3 (LS_ParFree_b)			Enable tension control. • Lenze setting: 0
	0	Tensile force controller reset	
	1	Tensile force controller active	
C00472/2 (LS_ParFree)	-199.99 % ... 199.99 %		Tensile force setpoint scaled [%] • Lenze setting: 20 %
C01040/1 (L_SRFG_1)	0 ... 999.99 s		Acceleration time for tensile force setpoint • Lenze setting: 0.5 s
C00472/7 (LS_ParFree)	-199.99 % ... 199.99 %		Influence of tensile force controller • Lenze setting: 5 %
C01056/1 (LS_ProcessCtrl_1)	0 ... 100		Tensile force controller gain • Lenze setting: 1
C01053/4 (LS_ProcessCtrl_1)	0 ... 30 s		Tensile force controller reset time • Lenze setting: 5 s
C01053/2 (LS_ProcessCtrl_1)	0 ... 30 s		Filter time constant for actual tensile force value • Lenze setting: 0.1 s
C00470/4 (LS_ParFree_b)			Enable tensile force reduction for rewinder (tensile force characteristic). • Lenze setting: 0
	0	Tensile force adaptation not active	The tensile force setpoint is not adapted.
	1	Tensile force adaptation active	Tensile force is reduced with an increased diameter according to the characteristic in the L_Curve_3.
C01030/1 (L_Curve_3)	4	Characteristic	L_Curve_3: do not change function! • Lenze setting: 4
C01035/1 (L_Curve_3)	0	Linear tensile force profile	L_Curve_3: selection of the tensile force characteristic • Lenze setting: 0
	1	Linear torque profile	
	2	Tensile force profile according to characteristic	
C00472/3 (LS_ParFree)	-199.99 % ... 199.99 %		Starting point of the tensile force characteristic • Lenze setting: 50 %
C00472/4 (LS_ParFree)	-199.99 % ... 199.99 %		Slope of the tensile force characteristic • Lenze setting: 60 %

Parameter (block)	Possible setting		Info
Friction compensation			
C00960/1 (L_Curve_1)			Activate friction compensation. • Lenze setting: 0
	0	Inactive	The compensation torque is permanently set to 0 %.
	3	Active	Via the characteristic of *L_Curve_1*, a speed-dependent compensation torque is connected.
C00963/1 ... 32 (L_Curve_1)	-32767 ... 32767		X values of the tensile force characteristic (speed)
C00964/1 ... 32 (L_Curve_1)	-32767 ... 32767		Y values of the tensile force characteristic (torque)
Acceleration compensation			
C01025/1 (L_Curve_2)			Enable acceleration compensation. • Lenze setting: 0 **Note:** The acceleration compensation can only be used reasonably if the cam table (C01028/C01029) and blocks *L_DT1*, *L_ConW_2*, and *L_ConW_4* are parameterised accordingly. Use the "Tool_TA8400_TensionControlled_V1-0.xlsx" Excel support tool to determine the matching parameters of these blocks!
	0	Inactive	The compensation torque is permanently set to 0 %.
	3	Active	Via the characteristic of the *L_Curve_2* block and the current diameter, a compensation torque is connected, which is proportional to the inertia.
C01028/1 c 32 (L_Curve_2)	-32767 ... 32767		X values of the characteristic adaptation of the moment of inertia
C01029/1 c 32 (L_Curve_2)	-32767 ... 32767		Y values of the characteristic adaptation of the moment of inertia
C00251/0 (L_DT1)	10 ... 5000 ms		L_DT1 time constant • Lenze setting: 10 ms (do not change)
C00252/0 (L_DT1)	-320 ... 320		Speed differentiation gain • Lenze setting: 100
C00940/2 (L_ConvW_2)	-32767 ... 32767		Acceleration scaling • Lenze setting: 1000
C00940/4 C0941/4 (L_ConvW_4)	-32767 ... 32767		Scaling of the moment of inertia • Lenze setting: C00940/4 = 330; C00941/4 = 1020
C00472/8 (LS_ParFree)	-199,99 % ... 199,99 %		Material width scaled • Lenze setting: 100 %

Parameter (block)	Possible setting		Info
Calculation of the diameter			
C00470/5 (LS_ParFree_b)			Load diameter.
	0	-	
	1	Load diameter	The *nSetDiameter_a* input value is accepted as the current diameter.
C00470/6 (LS_ParFree_b)			Hold last diameter value.
	0		Diameter is recalculated cyclically.
	1		No recalculation of diameter. The value calculated or loaded last is maintained.
C00470/7 (LS_ParFree_b)			Activate web break monitoring (*bUniDirect*). • Lenze setting: 0
	0	Inactive	The diameter may change in rewinding and unwinding direction.
	1	Active	An implausible change in diameter opposed to the winding direction preselected via *bUnwind* AND beyond the C01052/1 tolerance window sets the *bWebBreak* output to TRUE. The calculation of the change in diameter in the opposite direction is prevented. Therefore only activate the web break monitoring if the *dwOutDiameter* output value corresponds to the real diameter.
C00470/8 (LS_ParFree_b)			Calculation cycle for the diameter calculation • Lenze setting: 0
	0	Standard cycle	Use diameter recalculation 0 (C01050/1).
	1	Short cycle	Use diameter recalculation 1 (C01050/1).
C00472/1 (LS_ParFree)			Start diameter scaled to *d_max* C471_1 • Lenze setting: 50 %

3.5 Step 4: Transferring the parameter set to the inverter

1. Click the icon to go online.
2. Click on the icon to transmit the parameter set to the inverter.
3. After a successful transmission, click the icon to save the parameter set safe against mains failure in the integrated memory module.

3.6 Step 5: enabling the inverter

- Set motor (»Engineer« motor catalogue or nameplate)
  The motor rotates freely with an empty winding shaft.
- Load the diameter calculator with the minimum diameter so that the diameter is held on the smallest diameter possible during the calculation.
  In order to achieve this, ...
  - set C00472/1 = 0;
  - set the "Load diameter" control bit (C00470/5 = 1, bit 10 = 1);
  - set the "Hold diameter" control bit (C00470/5 = 1, bit 10 = 1).

3.7 Step 6: checking the parameter setting

3.7.1 Checking the winding direction

The "normal" winding direction has been defined in step 3:

▸ Winding direction (rewinder or unwinder) (📖 17)

The effective direction of a positive tensile force setpoint at the winding shaft is determined as follows:

- Select operation without feedback of the tensile force (open loop).
- Specify a small tensile force setpoint for line speed = 0.
- Enable winder and tension control open loop (DI3 = 1 and DI4 = 0 or bit 4 = 1 and bit 9 = 1).

The following effective directions are to be expected:

Winding direction	Unwinder (C00470/1 = 1)		Rewinder (C00470/1 = 0)	
Material feed	From the top (C00470/2 = 0)	From the bottom (C00470/2 = 1)	From the top (C00470/2 = 0)	From the bottom (C00470/2 = 1)
Movement of the winder	Counter-clockwise	Clockwise	Clockwise	Counter-clockwise

Note!

"Motor mounting direction = inverted" (C01206/1) may eliminate a potential inversion of the direction of rotation due to uneven-numbered gearbox stages and/or an inverted motor mounting position.

3.7.2 Check speed feedforward control

Check the speed feedforward control as follows:

- Enable the winder (DI3 = 1, bit 4 = 1) to actuate the drive in the "Follow line speed" operating mode.
- Start the line speed master and increase the speed to 50 %.

The winder must now rotate with half the speed parameterised in C00011.

When the material is fed from the top, the winding shaft rotates clockwise, whereas material feeding from the bottom makes the winding shaft rotate in counter-clockwise direction.

If the speed or direction of rotation is incorrect, check the above-mentioned definition of the basic parameters (C00470/1, C00470/2, C01206/1).

3.8 Step 7 (optional): Setting the disturbance compensation

In order to improve the dynamic properties, the following disturbance can be compensated:

- Acceleration torque
- Friction

3.8.1 Setting the acceleration compensation

The disturbance of the acceleration torque can be compensated via a characteristic. This characteristic is defined via an Excel support tool (Tool_TA8400_TensionControlled_V1-0.xlsx) and is imported into the »Engineer« in the form of a gdc file.

Function of the Excel support tool

The acceleration torque required for instance depends on the constant mass inertia (motor + mechanics) and the variable moment of inertia (reel). By means of the Excel support tool, the moment of inertia is determined as a function of the diameter and is scaled to the maximum moment of inertia. The grid points of this function are stored in the *L_CurveW_2* block. The input variable is the scaled diameter of the reel. The output is scaled, i.e. to the total maximum moment of inertia (constant + variable) relating to the motor shaft.

The quantisation of the acceleration signal at the output of the *L_DT_1* block significantly increases with great acceleration times (> 5 s). In order to counteract this, the acceleration is scaled to a standard acceleration time. This is taken into consideration with the factors in the *L_DT_1* and *L_ConvW_2* blocks when determining the acceleration. The moment of inertia is scaled to the acceleration required for this purpose using block *L_ConvW_4*.

If possible, the line speed signal should be derived from a stable setpoint of the speed-determining drive in the machine. For a less stable line speed signal, the number of significant bits which are taken into consideration in the differentiation process so that acceleration jumps are avoided can be set in the *L_DT1_1* block.

Conditional equations for the scaling:

- Acceleration scaling
  Acceleration time t_acc less than or equal to 3.2 s:

$$\text{L\_DT1\_1, C00252: } t_{acc} * 10$$

$$\text{L\_ConvW\_2, C00940/2: } 1000$$

  Acceleration time t_acc greater than 3.2 s:

$$\text{L\_DT1\_1, C00252: } 320$$

$$\text{L\_ConvW\_2, C00940/2: } t_{acc} * \frac{100000}{320}$$

- Scaling of the moment of inertia to the reference acceleration:

$$\text{L\_ConvW\_4, C00940/4: } 2\pi * \frac{C00011}{60} * \frac{1}{t_{acc}} * J_{max}$$

$$\text{L\_ConvW\_4, C00941/4: } C00057 * 1000$$

J_max: maximum moment of inertia relating to the motor shaft in kgm2

t_acc: rated acceleration time of the winder in s

_ _

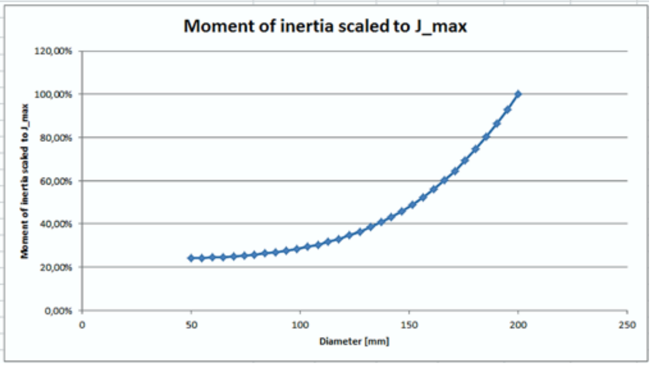
Handling the Excel support tool

The entries required for calculating the parameters are made in the support tool in the cells highlighted in yellow. Afterwards the development of the moment of inertia as a function of the current diameter is calculated and represented graphically.

Note!

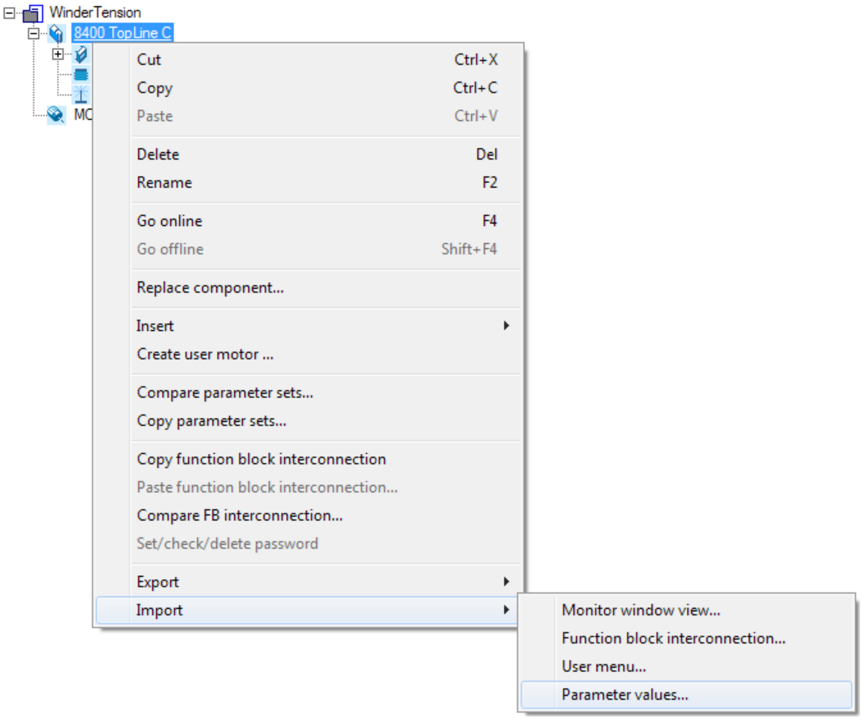
The maximum torque C00057 can only be read out online after having entered the motor model and after having adapted C00022 in the Inverter Drive 8400 TopLine.

	A	B	C	D	E	F	G	H
1	**Input Parameter**							
2	**Application**				**Motor**			
3	Maximum diameter	d_max	200	mm	Motor: Rated speed	n_motor	4050	rpm
4	Minimum diameter	d_min	50	mm	Motor: Rated power	P_motor	0,25	kW
5	Gearbox factor	i	1,0000		Motor: Rated current	I_motor	1,3	A
6	Reference line speed	v_100%	150,0	m/min	Maximum torque	C57	1,02	Nm
7	Minimum ramp-up time to reference line speed	t_acc	1,0	s	**Calculated values**			
8	Constant moment of inertia (Mechanic+Motor)	J_const	8,0	kgcm²	Motor: Rated torque	M_motor	0,6	Nm
9	Variable moment of inertia (Coil)	J_var	25,0	kgcm²	Max tension with dmax and rated motor torque	F_max	5,9	N
10	Maximum moment of inertia	J_max	33,0	kgcm²	Max tension with dmax and max torque C57	F_C57	10,2	N
11	Constant moment of inertia scaled to J_max	J_const_%	24,2	%	n_max / n_motor		0,236	
12					Required acceleration torque with J_max	M_acc	0,3	Nm
13								
14	**Calculated Values for Inverter**							
15	**Standard Parameter**					**Legend:**		
16	Motor: Reference speed	C11	955	rpm		Input		
17	Maximum diameter	C471/1	200	mm		Calculation		
18	Minimum diameter	C471/2	50	mm		Parameter for inverter		
19	Reference line speed	C471/3	1500	0,1m/min				
20								
21	**Acceleration compensation (optional)**							
22	C252 Gain L_DT1	C252	100					
23	L_ConvW Zähler	C940/2	1000					
24	Scaling factor Acceleration compensation	C940/4	330					
25		C941/4	1020					
26	L_Curve_2 x-Value_1	C1028/1	4096					
27	L_Curve_2 x-Value_2	C1028/2	4492					
28	L_Curve_2 x-Value_3	C1028/3	4889					
29	L_Curve_2 x-Value_4	C1028/4	5285					
30	L_Curve_2 x-Value_5	C1028/5	5682					
31	L_Curve_2 x-Value_6	C1028/6	6078					
32	L_Curve_2 x-Value_7	C1028/7	6474					
33	L_Curve_2 x-Value_8	C1028/8	6871					
34	L_Curve_2 x-Value_9	C1028/9	7267					
35	L_Curve_2 x-Value_10	C1028/10	7663					
36	L_Curve_2 x-Value_11	C1028/11	8060					
37	L_Curve_2 x-Value_12	C1028/12	8456					
38	L_Curve_2 x-Value_13	C1028/13	8853					
39	L_Curve_2 x-Value_14	C1028/14	9249					
40	L_Curve_2 x-Value_15	C1028/15	9645					
41	L_Curve_2 x-Value_16	C1028/16	10042					



How to import parameters in the »Engineer«:

1. Open the "Inertia_Export_Dancer" worksheet in the Excel tool.
2. Copy the content of cells E1 to E76 to a text editor using the Windows clipboard.
3. Save this text file and rename its file extension to ".gdc".
4. Click on the axis in the project tree using the left mouse button.
5. Select **Import → Parameter** values and select the file created.



3.9 Step 8 (optional): Optimising the closed-loop control parameters

3.9.1 Optimising the closed-loop speed control

The optimisation of the speed controller is only relevant for the inching mode and for the quick stop procedure, since the speed controller is not used in normal winding operation (torque open-loop-controlled).

In order to obtain a scaled setpoint speed for the speed controller, the scaled line speed signal is multiplied by the reciprocal diameter ($1/d_{act}$).

In order to ensure that the scaled setpoint winder speed complies with the scaled setpoint motor speed and the scaled line speed signal, carrying out the correct setting in C00011 (motor reference speed) is absolutely necessary. The scaled setpoint winder speed refers to the motor speed which, with the minimum diameter (d_{min}), is required to obtain the reference line speed at the circumference of the reel.

Adaptation of the speed controller gain

Under ideal circumstances, the gain of the speed controller must increase linearly with the moment of inertia effective at the motor shaft.

When the acceleration compensation has been parameterised correctly, the current moment of inertia is calculated automatically and is output at *L_ConW_4* as a scaled value. This signal can be used for the percentage speed controller adaptation.

Since, in practice, the gain must not increase proportionally with the moment of inertia, the process of adaptation is carried out via *L_GainOffsetP_1*:

- The scaled moment of inertia is multiplied by C00677/1.
- C00677/2 indicates the lower limit value of the speed controller gain.

In addition, the speed controller adaptation must be enabled with C00470/9 = TRUE.

The speed controller in the marginal areas should be optimised with a small and a very great diameter.

Online help for the Inverter Drive 8400 TopLine

In the "Optimising the speed controller" chapter, the general optimisation procedure is described.

3.9.2 Optimising the dancer position control

The control path of the dancer position control depends on many factors in the application and must therefore be optimised individually.

- Gain: C01056/1
- Reset time: C01053/4
- C00740/2 = TRUE (control bit 5 = TRUE) enables the PI controller. The influence is determined in C00472/7.

3.10 Step 9 (optional): Activating the web break monitoring function

For the purpose of diameter calculation, web break monitoring can be implemented using the *L_CalcDiameter_1* block. It is based on the fact that, in the event of a web break, the diameter calculated develops so that it opposes the winding direction.

The web break monitoring function is activated with C00470/7 = TRUE. This only makes a change in diameter opposite to the winding direction possible and permissible within the window parameterised in C01052.

The rewinding or unwinding operation is detected automatically, on the basis of the line speed sign and the winding direction parameterised. The response to this monitoring function is stored in the "Warning locked" Lenze setting and can be adapted in C00581/1.

Note!

If the web break monitoring function is active, a change in diameter opposing the winding direction specified via *bUnwindActive* is prevented. After loading a start diameter which significantly deviates from the real diameter in the direction opposing the winding direction, this may cause unintended triggering of the monitoring function.

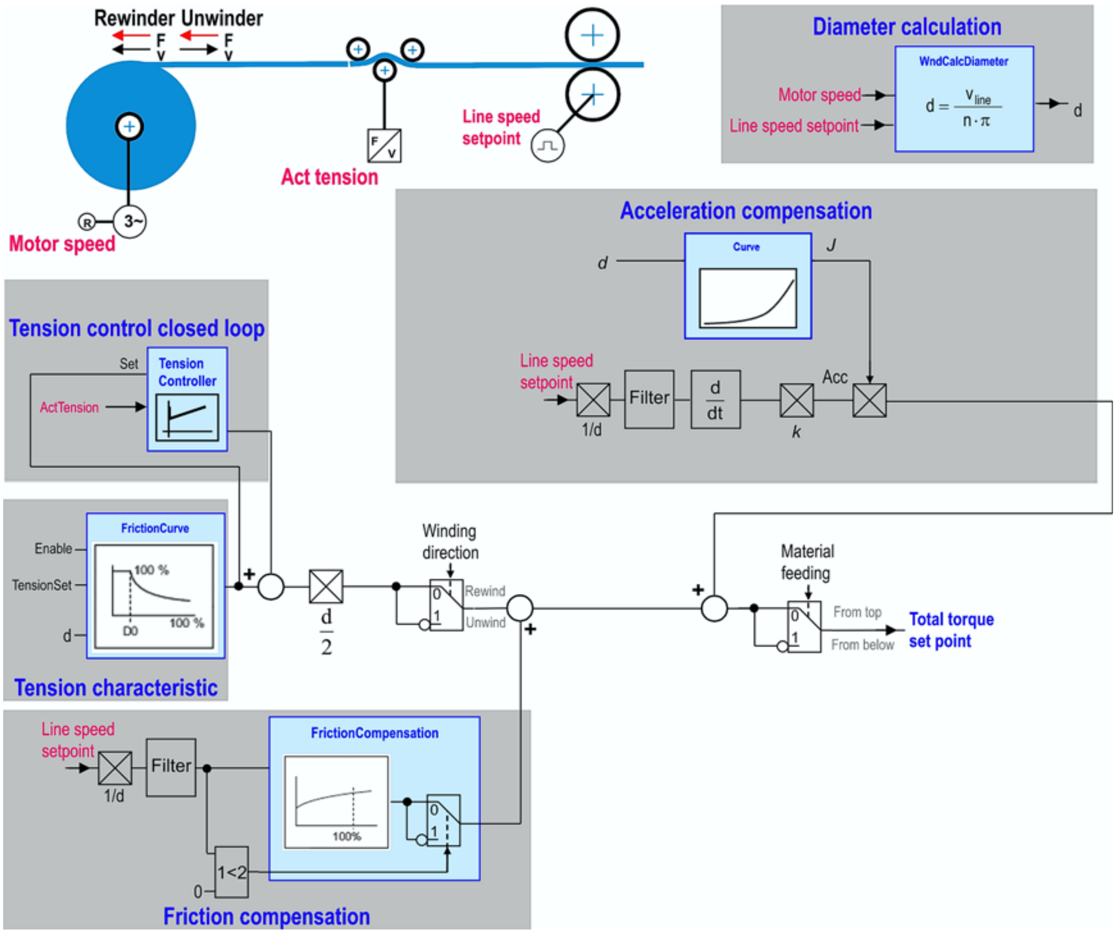
Example :

In the case of the rewinder, a start diameter of 50 % is loaded, however, the real diameter is only 45 %. The change in the diameter value to the real 45 % is prevented with an active web break monitoring function.

Web break monitoring may only be activated when the calculated diameter corresponds to the real diameter!

4 Detailed functions of the technology application

4.1 Signal flow of the technology application



4.2 Functions for winding operation

4.2.1 Speed feedforward control

The speed feedforward control on the one hand provides the speed limits for open/closed loop tension-controlled operation, and on the other hand provides the setpoint for the "Follow line speed" set-up mode.

In order to obtain a scaled winder setpoint speed for the speed controller, the scaled line speed signal is multiplied with the reciprocal diameter ($1/d_{act}$).

To make the scaled winder setpoint speed comply with the scaled motor setpoint speed and the scaled line speed signal, the matching setting in C00011 (motor reference speed) is absolutely required.

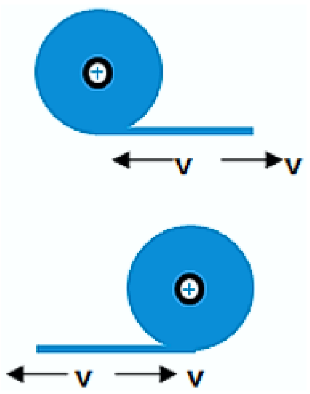
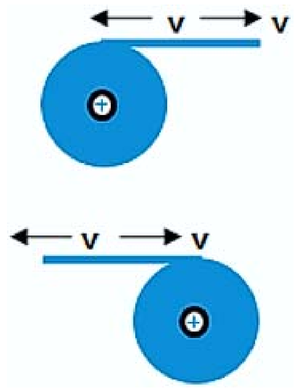
The scaled winder setpoint speed refers to the motor speed required with a minimum diameter d_{min} to achieve the reference line speed at the circumference of the reel.

Adaptation of the speed controller gain: ▸ Optimising the closed-loop speed control (📖 29)

4.2.2 Material feeding

The basic adjustment of the direction of rotation of the winder to the material flow is effected via parameter C01206/1 "motor mounting direction = inverted".

When the material is applied in an alternating fashion (from the top or from the bottom), the direction of rotation can be set via parameter C00470/2 or control bit 6. The signs of the speed follower setpoints are correspondingly inverted.

C00470/2 = 1 (material feeding from the bottom)	**C00470/2 = 0 (material feeding from the top)**
	

4.2.3 Inching mode

In order to traverse the winding shaft manually in inching mode, the generation of preset setpoints of the block *L_NSet_1* is used.

This automatically causes the main setpoint path to be decoupled via the line speed signal. This means that the winder can be moved independently of the speed-determining drive of the machine.

The overall setpoint is then evaluated with 1/d. Therefore parameterised setpoints in C00039/1 or C00039/2 for the positive and negative inching mode also refer to the circumferential speed or the line speed again and not to the motor speed.

4.2.4 Speed limitation

In the open-loop/closed-loop tension-controlled operation mode with material, the winder speed is determined by the circumferential speed and the current diameter. A speed limitation must take effect in the event of a web break or an overcompensation of a disturbance and prevent that the reel is accelerated in the direction of n_{max}.

Speed limitation in the event of a web break

Whereas the calculated diameter values are increasing (rewinder) or decreasing (unwinder) continuously during normal operation, a web break causes a change in direction in the diameter calculation.

In this process, the diameter of the rewinder might run towards d_{min}, which, without speed limitation, would cause acceleration of the winder up to n_{max}.

The unwinder diameter will run towards d_{max} in the event of a web break. With the speed limitation implemented, the speed in rewinding direction is not evaluated via the speed offset parameterised (C00660/2, C00660/4), but it will increase with $1/d_{max}$.

Speed limitation for disturbance overcompensation on an unwinder

In the case of an unwinder, an overcompensation of the friction or acceleration may result in acceleration in the unwinding direction. During normal operation, an unwinder always works in generator mode (speed in unwinding direction, torque in rewinding direction).

When the winder is accelerated, and during friction compensation, a change in the effective torque direction may occur. Then the winder works in motor mode. In this operating status, an overcompensation could lead to acceleration in the unwinding direction, in which case also the speed limitation function would take effect:

$$n_{set} = \frac{LineSpeedSetPoint + LineSpeedOffset}{d_{act}}$$

Note!

On an unwinder, an overcompensation of friction and/or acceleration should be avoided at all costs! The speed limitation, too, cannot prevent the generation of material slack. This results in the diameter approaching d_{min} and the winder rotating even faster (positive feedback).

For the speed limitation in torque-controlled operation, the speed offset is converted separately into a rotational speed offset by multiplication by 1/d. This value is added to / subtracted from the speed setpoint to calculate the upper and lower speed limits.

4.2.5 Calculation of the diameter

The current diameter is calculated by dividing the line speed by the motor speed. The following relationship can be established between the diameter, circumferential speed, and winding shaft speed:

$$d[mm] = \frac{v_{line}\left[\frac{m}{min}\right]}{n_{Wickler}\left[\frac{1}{min}\right] * \pi} * 1000$$

For further information, please see the description for the *L_CalcDiameter* block.

Setting the diameter value (transferring the sensor signal)

At the start of a winding procedure, the start diameter must be defined. It can be set or gathered from the signal of a diameter sensor.

- Parameterising the start diameter: C00472/1
- Loading the start diameter: C00470/5 = TRUE

An external diameter value (e.g. from an ultrasonic sensor) can be transferred via the same setting function. The sensor signal can also be loaded permanently.

Holding the diameter

For some operating states of the winder, in which the line speed signal does not correspond to the circumferential speed of the reel, the current diameter cannot be calculated from the line speed and the motor speed. In this case, the calculation of new values must be prevented and the diameter must be held at the old value. This is done automatically if:

- the line speed is smaller than the minimum line speed from C00472/5;
- control bit 11 "Hold diameter" is TRUE;
- tension-controlled (open loop) operation is not enabled;
- the motor speed is lower than C00024.

4.2.6 Tension control open loop and winding characteristic

In the case of some winding materials it may be required to adapt the tensile force with the diameter becoming smaller. Otherwise the reel may drift to the sides, meaning it telescopes.

The diameter-dependent winding characteristic or tensile force characteristic is defined via a characteristic function in the *L_Curve_3* block.

The tensile force setpoint is defined in a scaled fashion relating to a reference force. The setpoint is guided via a ramp generator. The acceleration/deceleration time can be set in C01040/1.

For further information, please see the description for the *L_Curve_3* block.

4.2.7 PI controller for the tensile force control

By enabling the *L_ProcessCtrl_1* process controller, optionally closed-loop control to an actual tensile force value can be carried out. Enable is executed with C00470/3 = TRUE. Use parameter C00472/7 to define the influence that the PI controller is to have on the motor control.

In order to avoid discontinuities in the tensile force profile during the enabling process, the actual value is loaded into the ramp generator first before the tensile force setpoint is loaded. When the tension-controlled (open loop)/closed-loop controlled operation "TensionMode" has been activated, the setpoint for the controller is identical to the actual value at first. Then, controlled via the ramp generator, a switch-over to the current tensile force setpoint takes place.

The actual tensile force value must be adapted using the settings of the analog input and the *L_ConvW_3* block, so that 100 % = 16384 is shown at *L_ConvW.wOut* when a reference force (nominal tension in the Excel tool) is applied.

The PID controller that is used for tension control offers various possibilities of conditioning the setpoints and actual values.

For further information, please see the description for the *L_ProcessCtrl* block.

4.3 Disturbance compensation functions

4.3.1 Acceleration compensation

The acceleration in the line speed setpoint constitutes a disturbance in the winding process, because the torque that is "consumed" for the process of acceleration is missing in the tensile force. Therefore the acceleration torque must be calculated and pilot-controlled as an additional torque.

In this process, scaled values are used for calculation. The additional scaling to the acceleration time is to improve the resolution of the acceleration torque.

In practice, a line speed signal that does not ideally increase is to be expected. Use parameter C00253 to set the resolution of the signal that is differentiated.

Only the number of higher-order bits that is set is taken into consideration for the differentiation. Furthermore the signal is smoothed via a PT1 functionality subsequently. The time constant can be set via parameter C00251.

The gain of the DT1 element and the scaling of the *L_ConvW_2* are used for taking the acceleration time *t_acc* into consideration.

The mass inertia of the drive results from the current diameter by a characteristic in which the moment of inertia scaled to the maximum moment of inertia *J_max* as a function of the diameter scaled to *d_max* is stored. Then the moment of inertia is evaluated with *M_acc / M_max* in order to achieve an improved scaling of the acceleration.

The moment of inertia is multiplied with parameter C00741/8 in order to provide for different material widths with all other settings remaining consistent.

Note!

The speed is resolved with 16 bits in the Inverter Drive 8400, 100 % = 2^14 = 16384 corresponding to the rated speed (C00011).

The rated speed is to be selected so that it is obtained with a minimum diameter at a line speed of 100 %. In the case of slow acceleration processes, the limited speed resolution produces quantisation steps of the same size as the change in setpoint values. Therefore, in particular for slow acceleration processes (>10 s), the feedforward control value is to be checked and deactivated, if necessary.

A greater filter time constant in the PT1 filter of the acceleration (C00250) can bring about an improvement.

If possible, the line speed signal should be derived from a stable setpoint instead of from an actual speed value, in order to obtain a utilisable acceleration feedforward control.

4.3.2 Friction compensation

The reference speed of the characteristic for compensating the friction is derived from the line speed signal. Therefore the compensation is a setpoint-based process, which is usually advantageous compared to using the actual motor speed as reference, since the setpoint is more stable. Furthermore the reference speed can also be filtered in order to increase the stability of the compensation process.

Effective direction of friction compensation

In the case of a rewinder, the friction compensation increases the total torque setpoint, whereas for the unwinder, the friction torque must be subtracted from the total torque setpoint.

In the Lenze setting, this effective direction is automatically switched over depending on the sign of the line speed and the winding direction set in C00470/1. The friction torque is inverted if, due to the negative sign of the line speed ,the winder is working so that it opposes the winding direction parameterised in C00470/1. Depending on the application, it may be reasonable to switch off this automatic feature, for instance if no rewinding/unwinding/half-duplex operation is required, or if the rewinding/unwinding/half-duplex operation is defined externally by a control.

If the automatic feature takes effect in half-duplex operation, the effective direction of the friction compensation may be incorrect at a standstill (line speed = 0), since the sign can only be determined at a line speed <> 0. In these cases it is necessary to externally define the desired effective direction via a Boolean signal (control, visualisation, DigIn). This requires interfering in the signal flow of the friction compensation.

Note!

If the friction is also be compensated at a line speed = 0, the current motor speed must be used as reference speed for the characteristic. This case for instance occurs if, when the winder is started, a web tension has to be established first as long as the line is still at a standstill, and, in addition, the tensile force setpoint torque is smaller than the friction torques.

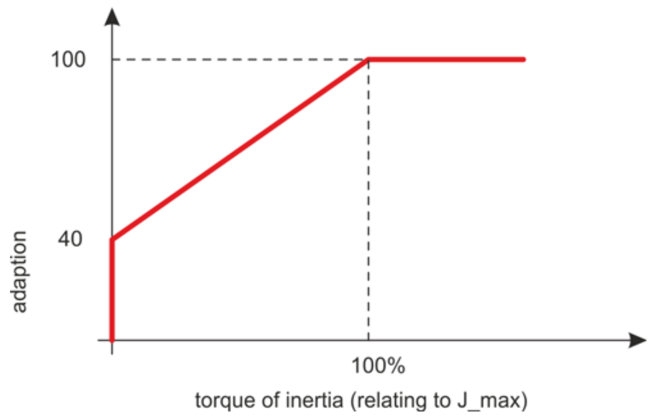
4.4 Adaptation of the speed controller gain

If we consider the motor and the reel as a rigid one-mass system, the optimum gain of the speed controller is directly proportional to the moment of inertia.

Since the moment of inertia usually changes significantly during the winding process, for a good control response it may be necessary to continuously adapt the gain of the speed controller to the moment of inertia.

The adaptation of the speed controller gain can only be used reasonably if the acceleration compensation is configured (J characteristic stored) so that the calculation of the current moment of inertia supplies correct values.

In the default setting, the speed controller is adapted linearly in the range from 40 ... 100 % of the moment of inertia. Below this range, the gain is held constant.



Note!

In tension-controlled (open loop)/closed-loop controlled operation, the speed controller does not interfere. However, the adaptation of the gain may be reasonable to work with an adapted gain in Inching mode (32) and at quick stop (QSP).

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Drives GmbH
Postfach 10 13 52, D-31763 Hameln
Breslauer Straße 3, D-32699 Extertal
Germany
HR Lemgo B 6478
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Accessories

POWERLINK

E84AYCEC

Inverter Drives 8400 ________________________

Communication manual EN

13564917

Lenze

Contents

1 About this documentation

Contents

This documentation exclusively describes the E84AYCEC communication module (POWERLINK).

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **hardware manual for the Inverter Drives 8400**.

The features and functions of the communication module POWERLINK are described in detail.

Examples illustrate typical applications.

This documentation also contains the following:

- Safety instructions that must be observed
- The basic technical data of the communication module
- Information about the versions of the Lenze standard devices to be used
- Notes on troubleshooting and fault elimination

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe any software provided by other manufacturers. No liability can be accepted for corresponding data provided in this documentation. For information on how to use the software, please refer to the host (PLC, *Managing Node* documents.

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information about POWERLINK can be found on the website of the "Ethernet POWERLINK Standardization Group":

www.ethernet-powerlink.org

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the field devices and the software version of the installed engineering tools (»Engineer«), the screenshots in this documentation may differ from the screen representation.

Target group

This documentation addresses to persons who configure, install, commission, and maintain the networking and remote maintenance of a machine.

Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.Lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

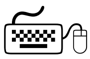
Extension module	Type designation	From hardware version	From software version
Communication module POWERLINK	E84AYCEC	VA	01.00

1.1 Document history

Version			Description
4.0	02/2019	TD23	General updates
3.0	09/2012	TD17	General updates Description of code C13060 (0x1006 cycle time) corrected.
2.0	02/2011	TD17	General revision
1.0	11/2010	TD17	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal	Normal spelling	Example: 1234
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Text		
Program name	» «	PC software Example: Lenze »Engineer«
Window	*italics*	The *message window*... / The *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Hyperlink	Underlined	Optically highlighted reference to another topic. Can be activated with a mouse-click in this documentation.
Icons		
Page reference	(🕮 7)	Optically highlighted reference to another page. Can be activated with a mouse-click in this documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Inverter	Lenze inverter of the "Inverter Drives 8400" product series
Standard device	
»Engineer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
Code	Parameter which serves to parameterise and monitor the inverter. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). This term is also referred to as "subindex" in common parlance.
Lenze setting	Default settings of the device, preconfigured ex works.
Basic setting	
HW	Hardware
SW	Software
EPL	Abbreviation for "POWERLINK"
(EPL) nodes	POWERLINK nodes (*Managing Node*, *Controlled Nodes*)
CN	Controlled node (EPL slave)
MN	Managing node (EPL master) The *Managing Node* accepts the control function for the data communication of the decentralised field devices. Typically, the *Managing Node* is the communication interface of a PLC.
Node ID	EPL node address

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

> **Pictograph and signal word!**
>
> (characterise the type and severity of danger)
>
> **Note**
>
> (describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Stop!	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

- Lenze drive and automation components ...
  - must only be used as directed.
    ▶ Application as directed (📖 12)
  - must never be commissioned if they display signs of damage.
  - must never be technically modified.
  - must never be commissioned if they are not fully mounted.
  - must never be operated without required covers.
  - during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.
- The following applies to Lenze drive components ...
  - only use the accessories approved.
  - Only use original manufacturer spare parts.
- Observe all specifications contained in the enclosed documentation and related documentation.
  - This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
    ▶ Product features (📖 13)
  - The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.
- All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...
  - are familiar with installing, mounting, commissioning, and operating the product.
  - who have the corresponding qualifications for their work.
  - who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables corresponding to the given specifications.
  ▶ Ethernet cable specification (📖 32)

Documentation for the standard device, control system, system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

- If the Inverter Drives 8400 are operated on a phase-earthed mains with a rated mains voltage of ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.
  ▶ Protective insulation (📖 16)

Device protection

- The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
  ▶ Installation (📖 21)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

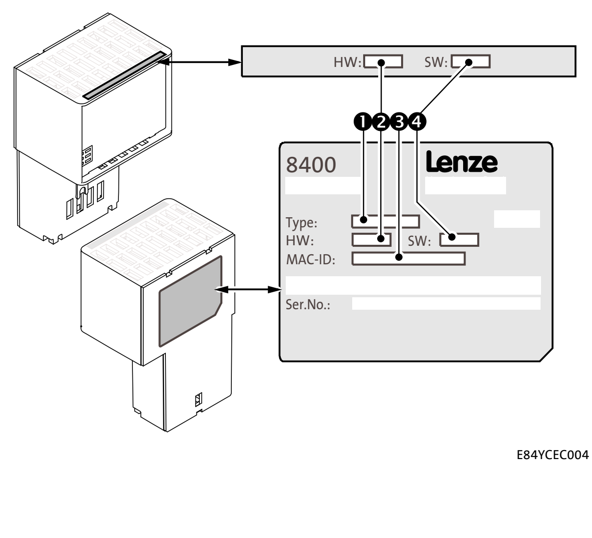
Product series	Type designation	From software version
Inverter Drives 8400 StateLine	E84AxSCxxxx	5.00
Inverter Drives 8400 HighLine	E84AxHCxxxx	5.00
Inverter Drives 8400 TopLine	E84AxTCxxxx	1.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in POWERLINK networks.

Any other use shall be deemed inappropriate!

3.2 Identification

Type designation and hardware and software version of the communication module are indicated on the nameplate:



1 Type designation (type)

E84	Product series
A	Version
Y	Module identification: Extension module
C	Module type: Communication module
EC	POWERLINK CN
V/S	V: Coated version S: Standard version

2 Hardware version (HW)

3 MAC address (MAC-ID)

00-0A-86-xx-yy-zz:
Hardware address of the communication module for unique identification in the network

4 Software version (SW)

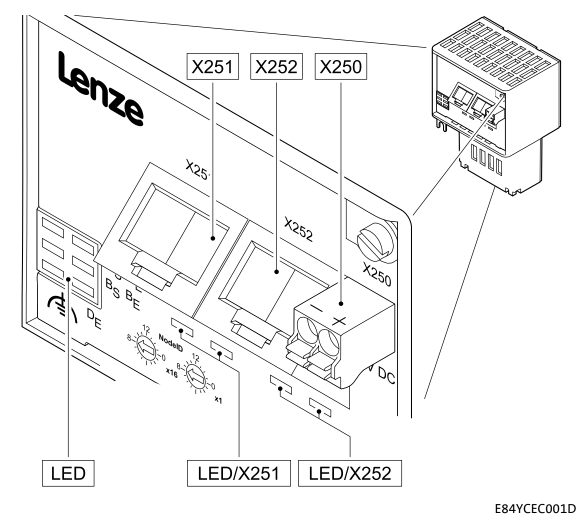
[3-1] Identification data

3.3 Product features

- Interface module for the POWERLINK communication system, for attachment to the expansion slots of Inverter Drives 8400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- Real-time Ethernet with the Ethernet POWERLINK V2 communication profile for motion and general applications
- A line topology can be implemented by means of the integrated 2-port hub.
- Supported functions:
  - POWERLINK CN
- Use of max. 6 PDO crosslinks for the managing node or the controlled node to create systems with "distributed intelligence"
- Very short CN response times for optimal network performance
- Parallel Ethernet communication between the Lenze »Engineer« and the Inverter Drive 8400 if an EPL router is used.
- Access to all Lenze parameters

3.4 Connections and interfaces

- 2 RJ45 sockets for the POWERLINK connection
- 2-pole plug connector for the external voltage supply of the communication module.
- 2 rotary coding switches for setting the node address (node ID)
- Front LEDs for diagnosing the ...
  - voltage supply of the communication module;
  - connection to the standard device;
  - POWERLINK connection;
  - POWERLINK activity.



X250	External voltage supply of the communication module • 2-pin plug connector with spring connection ▸ External voltage supply (📖 34)
X251	POWERLINK port 1
X252	POWERLINK port 2 • RJ45 sockets according to IEC/EN 60603-7 • each with 2 LED status displays for diagnostics ▸ POWERLINK connection (📖 30) ▸ Status displays at the RJ45 sockets (X251, X252) (📖 55)
x1 **x16**	Rotary coding switch for setting the node address (node ID) ▸ Node address setting (📖 37)
MS **ME** **BS** **BE** **EN**	5 LED status displays for diagnostics ▸ Module status displays (📖 53) ▸ Fieldbus status displays (📖 54)

[3-2] E84AYCEC communication module (POWERLINK)

4 Technical data

4.1 General data and operating conditions

Range	Values	
Order designation	E84AYCEC	
Communication profile	POWERLINK	
Communication medium	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e	
Interface	RJ45: Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)	
Network topology	Tree, star, and line	
Type of node	Controlled node	
Node addresses (node IDs)	1 ... 239	
Transmission mode	Half duplex	
Baud rate	100 Mbps	
Voltage supply	External supply via separate power supply unit • +: U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %), I = 140 mA • -: reference potential for external voltage supply	
Conformities, approvals	CE	2004/108/EC, EMC Directive 2006/95/EC, Low-Voltage Directive
	EAC	Eurasian conformity TR CU: Technical Regulations of Customs Union
		TP TC 004/2011 (TR CU 004/2011) About the safety of low voltage equipment
		TP TC 020/2011 (TR CU 020/2011) Electromagnetic compatibility of technical means
	UL	UL 508C / C22.2 No. 14, Power Conversion Equipment (File-No. E132659)

Hardware manual for Inverter Drives 8400

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 Protective insulation

Danger!

Dangerous voltage

If the Inverter Drives 8400 are operated on a phase-earthed mains with a rated mains voltage of ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

- Death or severe injuries

Protective measures:

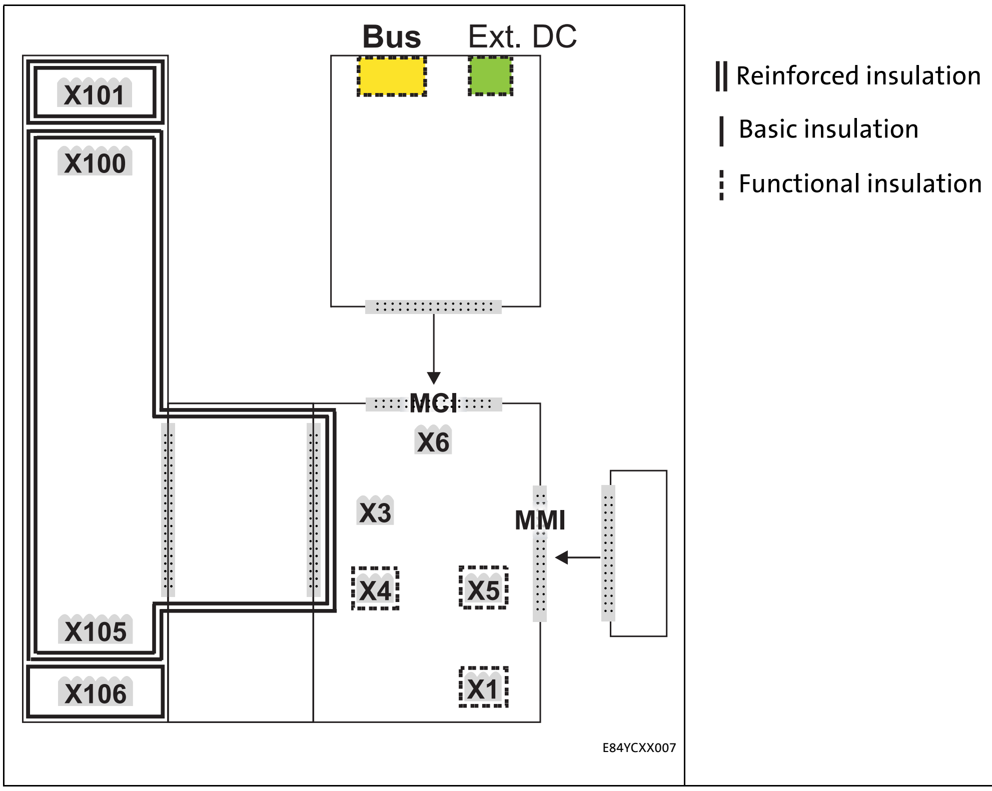
- If protection against accidental contact is required for the control terminals of the inverter and the connections of the device modules plugged, ...
  - a double isolating distance must exist.
  - the components to be connected must be provided with the second isolating distance.

Note!

The existing protective insulation in the Inverter Drives 8400 is implemented according to EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Inverter Drive 8400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection
X100	Mains/DC-bus connection
X101	Relay contact
X105	Motor/brake resistor
X106	Motor PTC
X1	System bus (CANopen)
X3	Analog inputs/outputs
X4	Digital outputs
X5	Digital inputs
X6	Diagnostics
MCI	Slot for the communication module
MMI	Slot for the memory module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MCI and the X100 mains terminal?

- The separate potential area with the better protective insulation is decisive.
  - The separate potential area of the bus terminal of the device module has a "functional insulation".
  - The separate potential area of the mains terminal has a "reinforced insulation".
- Result: The insulation between the X100 mains terminal and the bus terminal is a "reinforced insulation".

4.3 POWERLINK communication data

Range	Values
Min. cycle time	400 µs
Total cycle times	0.4 / 0.5 / 1.0 / 2.0 / 3.0 ... 20.0 ms
Buffer size	• Tx-iso: max. 1490 bytes • Rx-iso: max. 1490 bytes (max. 32 bytes of PDO user data)
Delay time	• Controlled node (T_{Preq} - T_{Res}): approx. 2.6 µs • Controlled node (T_{SoA} - T_{ASnd}): approx. 2.6 µs
Frame size	Max. asynchronous telegram size (MTU): 1490 bytes
SDO communication method	UDP/IP or ASND
Number of RPDOs	Max. 6 channels
RPDO user data per application (all RPDOs)	Max. 16 objects (max. 32 bytes)
Number of TPDOs	1 channel (data access of all nodes by broadcasting)
TPDO user data per application	Max. 16 objects (32 bytes)
CN operating modes	Support of ... • Multiplex CNs • Optional CNs

Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in an POWERLINK network depend on the ...

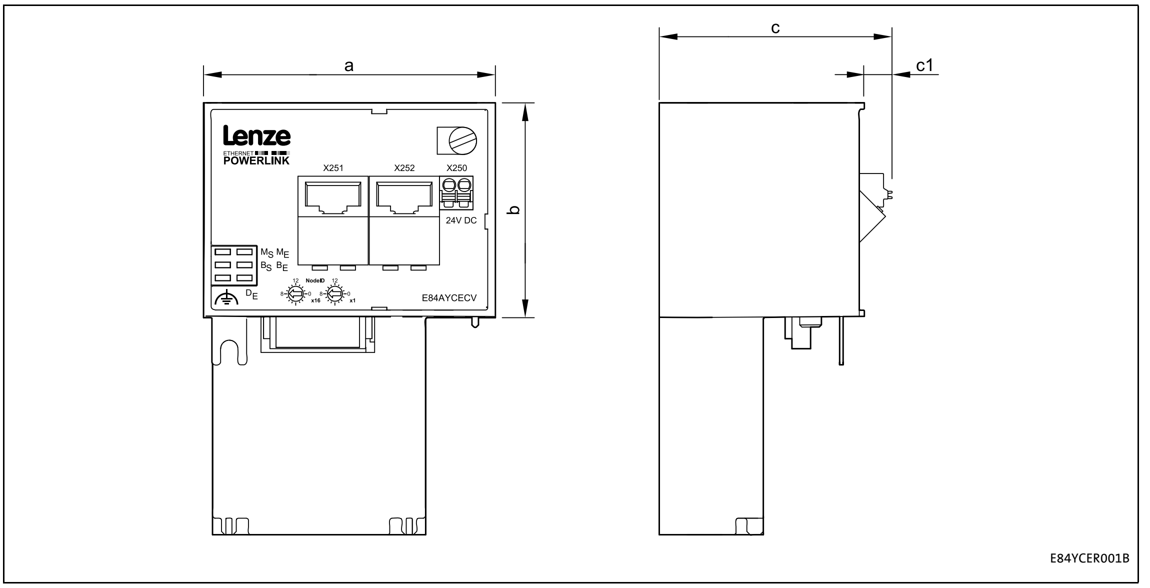
- processing time in the inverter;
- frame runtime (baud rate / frame length);
- the nesting depth of the network.

Processing time inside the inverter

Data	Processing time	
Process data	Approx. 2 ms + 0 ... 1 ms + 1 ... x ms	Update cycle Processing time in the module Runtime of the application task of the technology application used (tolerance)
Parameter data	Approx. 30 ms + a tolerance of 20 ms (typically) • For some codes, the processing time may be longer (see software manual/»Engineer« online help for the Inverter Drive 8400).	

There are no interdependencies between parameter data and process data.

4.4 Dimensions



[4-2] Dimensions

Type	Dimensions [mm]			
	a	b	c	c1
E84AYCEC	67	50	57	8

5 Installation

Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Fieldbus communication is not possible or faulty.

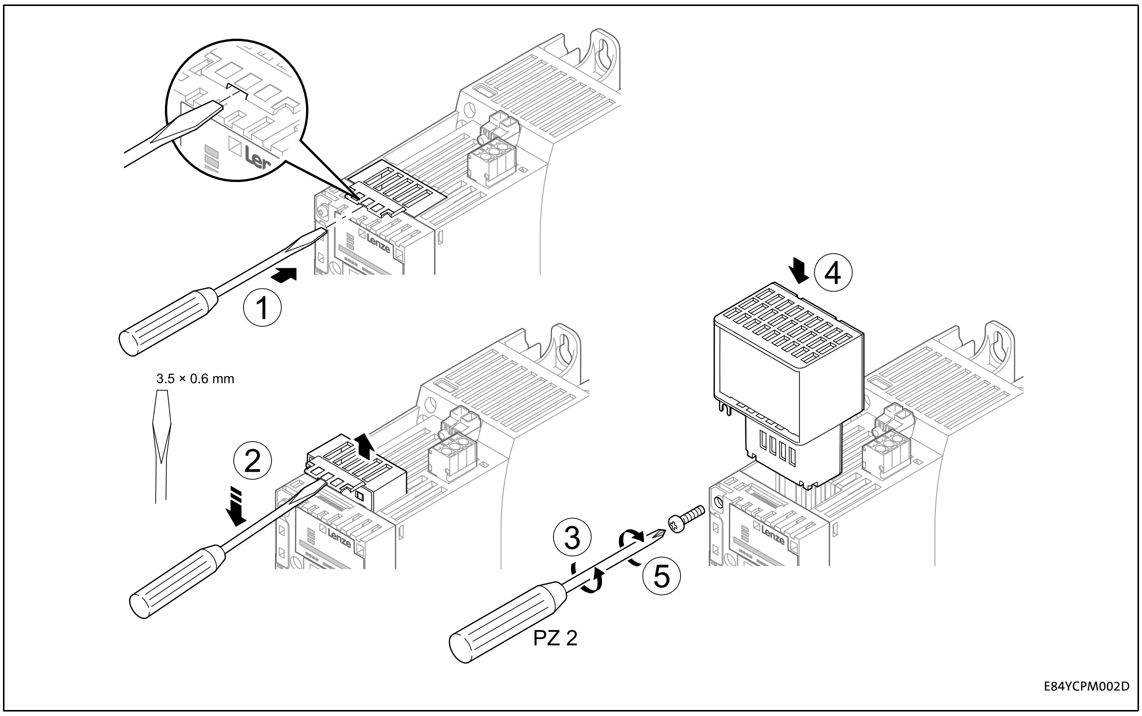
Protective measures

- Before touching the module, be sure that you are free of electrostatic charge.

5.1 Mechanical installation

The communication module can be plugged in or unplugged from the MCI slot when the inverter is switched on. When the module is plugged in, it is detected automatically, and a function and version plausibility check is executed.

5.1.1 Mounting for standard devices of 0.25 kW and 0.37 kW

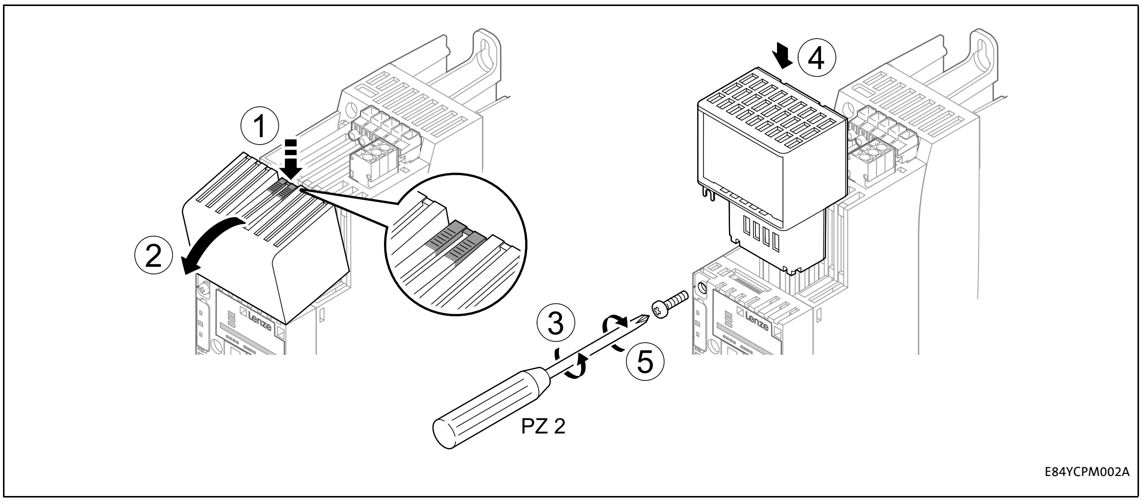


[5-1] Mounting for standard devices of 0.25 kW and 0.37 kW

Mounting steps

1. Use a screwdriver to lever out the cover of the MCI slot of the standard device and remove it (1, 2).
2. Loosen the securing screw for the communication module at the standard device (3).
3. Insert the communication module into the MCI slot of the standard device (4).
4. Tighten the securing screw again (5).

5.1.2 Mounting for standard devices of 0.55 kW or more

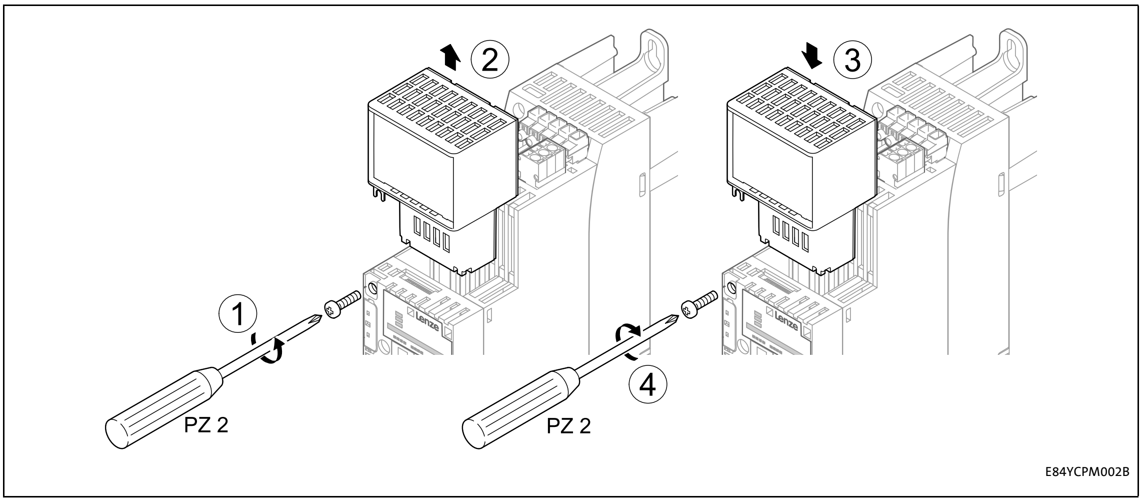


[5-2] Mounting for standard devices of 0.55 kW or more

Mounting steps

1. Slightly press on the marked spot on the top of the MCI slot cover of the standard device (1).
2. Tilt the cover forward and remove it from the standard device (2).
3. Loosen the securing screw for the communication module at the standard device (3).
4. Insert the communication module into the MCI slot of the standard device (4).
5. Tighten the securing screw again (5).

5.1.3 Exchanging the communication module



[5-3] Exchanging the communication module

Mounting steps

1. Loosen the securing screw for the communication module at the standard device (1).
2. Pull the communication module out of the MCI slot of the standard device (2).
3. Insert the new communication module into the MCI slot of the standard device (3).
4. Tighten the securing screw again (4).

5.2 Electrical installation

Documentation for the standard device, control system, system/machine
Observe the notes and wiring instructions contained in this documentation.

5.2.1 Wiring according to EMC guidelines

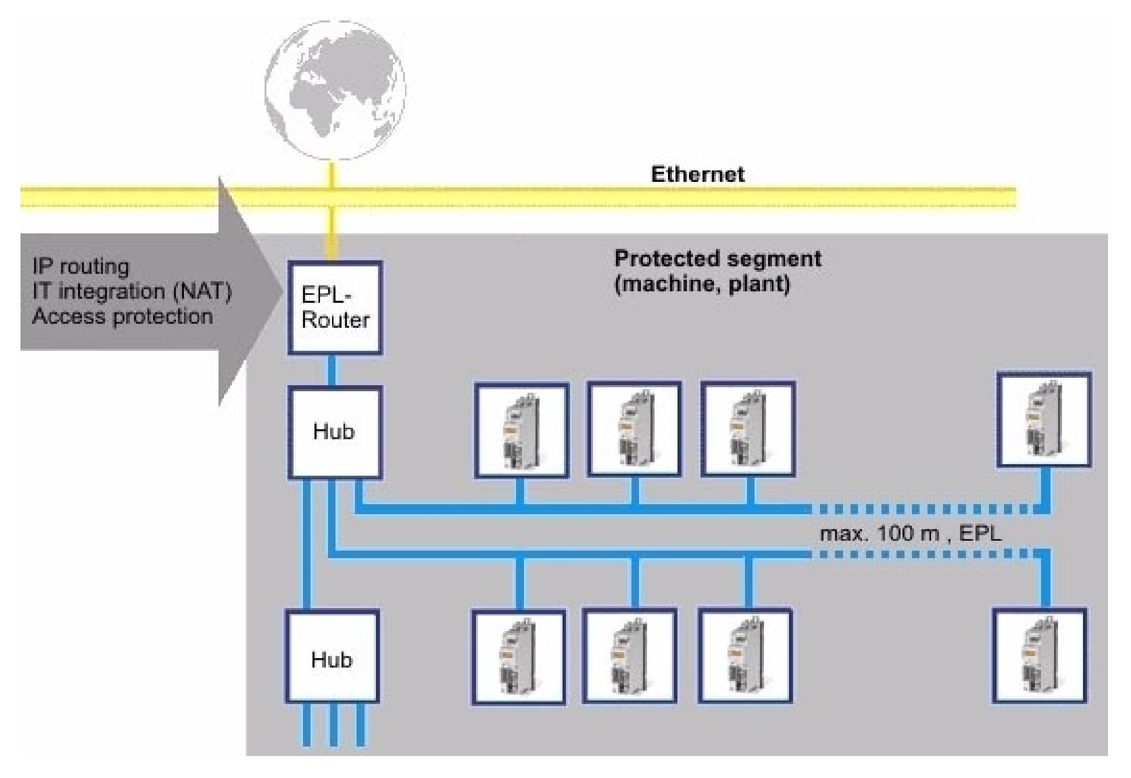
In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. Remove the plastic sheath of the cable at a length of 2 cm.
2. Fasten the cable shield to the shield support of the standard device.

5.2.2 Network topology



[5-4] Network topology for POWERLINK

Note!

The use of class I hubs and switches inside the EPL network segment is not permitted.

Inside the segment only Ethernet hubs may be used as infrastructure elements. The hubs must meet the requirements on class II repeaters acc. to IEEE 802.3u.

For this purpose Lenze offers the dual hub integrated into the communication module and the separate eight-fold hub, type E94AZCEH.

Class I hubs and switches are not permissible since they have considerably longer delay times for the frame forwarding and a bigger jitter. Both sizes reduce the real-time capability and dynamics.

The cable length between both nodes is limited to 100 m.

The topology rules (IEEE 802.3u) required for controlling the collisions may be violated in the EPL network segment since according to the EPL access order frame collisions are prevented. This enables a structure of lines and any hybrid forms between star and line topology.

Recommended topology

For an easy configuration and due to many possible topology variants we recommend to create networks according to the following rules:

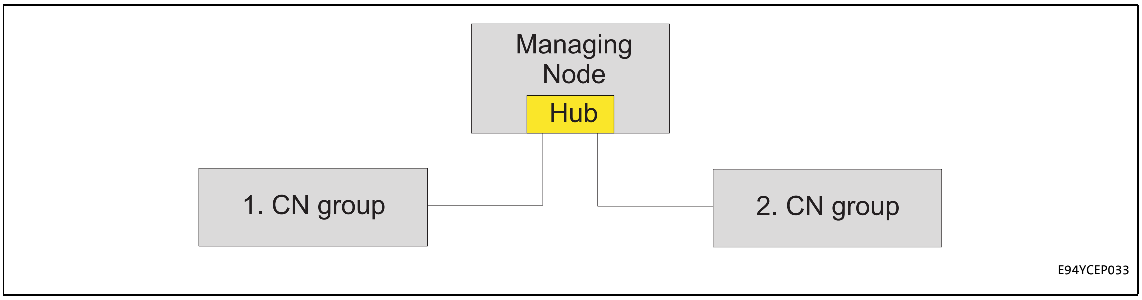
1. Create CN groups with up to 10 nodes.
2. Connect CN groups in star shape to the managing node.

3. Connect CN groups to the managing node via one external hub each.
   - For more than 2 CN groups, use an external 8-port hub (e.g. Lenze hub E94AZCEH).
   - For max. 7 CN groups one hub is sufficient.
   - For more than 7 CN groups, use further hubs.
   - The groups can be distributed on the hubs just as you like.
   - Observe the restrictions of the used managing node.

Note!

E94AYCEP communication module als Managing Node

With maximally 2 CN groups and the use of an E94AYCEP communication module as *Managing Node*, connect the CN group(s) directly to the POWERLINK terminals X211 and X212 of the E94AYCEP communication module.



[5-5] Star topology for 1 to 2 CN groups

5.2.3 Ethernet POWERLINK

Note!

Standard Ethernet nodes are not permitted in the POWERLINK network segment.

In order to use the real-time capability of the POWERLINK technology, POWERLINK nodes must be interconnected in a separate network segment.

In accordance with the EPL rules, only the managing node controls the network access of EPL nodes. The managing node is the only node that transmits autonomously. The slave nodes (i.e. all controlled nodes) only transmit when they are entitled to transmit by the managing node.

Non-EPL nodes (e.g. PCs) typically violate these rules by sending frames independently of the managing node. These frames interfere with the cyclic frame exchange of the EPL nodes and impede the real-time capability of POWERLINK.

Connection to the standard Ethernet network

The connection to an external standard Ethernet network is carried out via an POWERLINK router or an POWERLINK gateway.

These infrastructure components separate the network traffic in the POWERLINK network segment from the one in the standard Ethernet. The handling of the frames depend on their direction:

- Standard Ethernet ---> EPL network segment:
  Only frames that are addressed to nodes in the EPL network segment are forwarded. The forwarding takes place in the asynchronous area of the EPL cycle.
- EPL network segment ---> standard Ethernet:
  Only asynchronous frames that are not addressed to nodes in the EPL network segment are forwarded.

Tip!

Detailed information on the function and setting of the router or gateway can be found in the documentation of the component manufacturer.

5.2.4 Basic Ethernet Mode

Note!

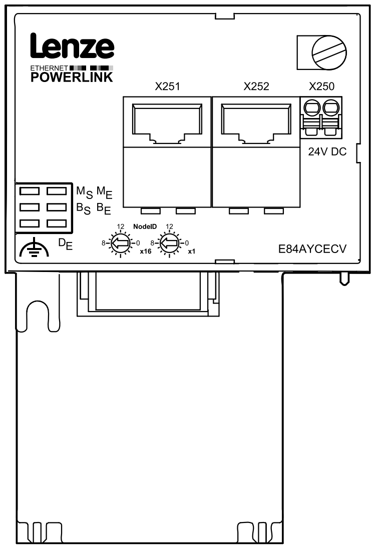
Operation in the "Basic Ethernet Mode" does not permit any real-time communication.

The communication module can be operated in the "Basic Ethernet Mode" for a basic parameter setting provided that the following applies:

1. Operation of the module with network address ≤ 239.
   - IP address: 192.168.100.[*EPL address*]
   - see Node address setting (📖 37)
2. No operation with real-time EPL must be carried out.
3. No integration of a managing node (EPL address = 240) into the standard Ethernet network.

5.2.5 POWERLINK connection

POWERLINK is connected via the RJ45 sockets **X251** (P1) and **X252** (P2).



E84YCEC001E

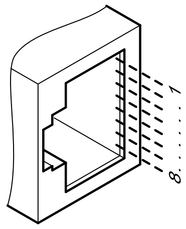
[5-6] POWERLINK connections X251 (P1) and X252 (P2)

- If possible, do not wire more than 10 nodes in a row in a network line.
- In case of more than 10 nodes you should deactivate the automatic calculation of the hub level in the »Engineer« and enter it manually to increase the performance.
- With a cycle time of 1 ms, 25 nodes can be operated.
  - The number of nodes can be increased if the EPL cycle time is increased as well.
- You can use a standard Ethernet patch cable to connect the communication module to the POWERLINK fieldbus (see Ethernet cable specification (📖 32)).
- The installation and removal of the Ethernet cables is optimised for the use of connectors in accordance with the "Automation Initiative of German Domestic Automobile Manufacturers" (AIDA).

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment of the RJ45 sockets

RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Tip!

The POWERLINK interfaces feature an auto-MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite POWERLINK interface and irrespective of the type of cable used (standard patch cable or crossover cable).

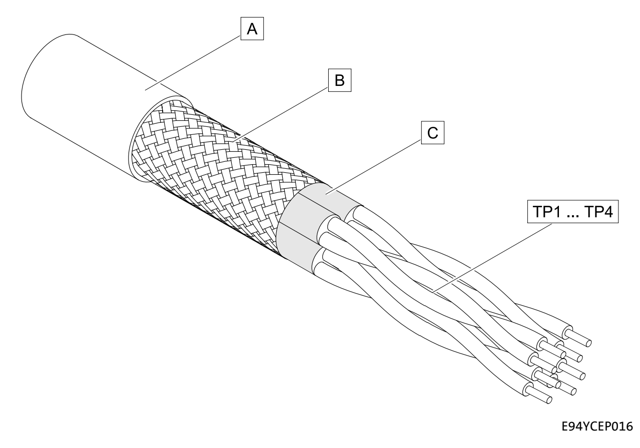
5.2.6 Ethernet cable specification

Note!

Only use cables that meet the listed specifications.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 … TP4 Twisted core pairs 1 ... 4
▸ Colour code of the Ethernet cable (🕮 33)

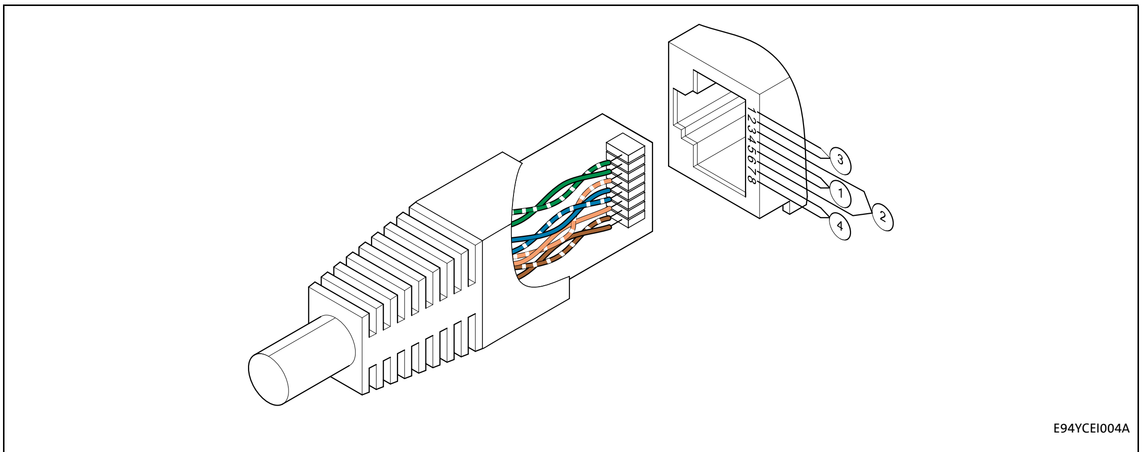
[5-7] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-8] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	White / Green	White / Orange
	2	Tx -	green	orange
2	3	Rx +	White / Orange	White / Green
1	4		blue	blue
	5		White / Blue	Blue / White
2	6	Rx -	orange	green
4	7		White / Brown	White / Brown
	8		brown	brown

5.2.7 External voltage supply

The communication module can be externally supplied with voltage via separate supply cables at the 2-pin plug connector **X250**.

Note!

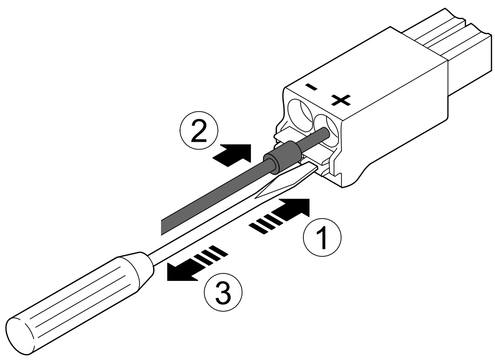
Always use a separate power supply unit safely separated according to EN 61800-5-1 ("SELV/PELV") in every control cabinet for external voltage supply.

- External voltage supply of the communication module is required if the communication via the bus should be maintained when the supply of the standard device fails.
- Access to parameters of a standard device disconnected from the mains is not possible.

Wiring of the X250 plug connector

Stop!

Only wire the plug connector if the standard device is disconnected from the mains.



[5-9] Wiring of the 2-pin plug connector with spring connection

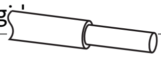
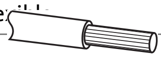
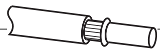
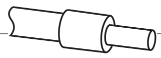
How to wire the plug connector with spring connection:

1. Press a screwdriver into the notch below the terminal and keep it pressed.
2. Place the supply cable in the terminal.
3. Remove the screwdriver from the notch.

Assignment of the X250 plug connector

Name	Description
+	U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %) I = 140 mA
-	Reference potential for the external voltage supply

Terminal data

Range	Values	
Electrical connection	2-pin plug connector with spring connection	
Possible connections	Rig	
		0.2 ... 1.5 mm² (AWG 24 ... 16)
	Fle	
		Without wire end ferrule 0.2 ... 1.5 mm² (AWG 24 ... 16)
		With wire end ferrule, without plastic sleeve 0.2 ... 1.5 mm² (AWG 24 ... 16)
		With wire end ferrule, with plastic sleeve 0.2 ... 1.5 mm² (AWG 24 ... 16)
Stripping length	10 mm	

6 Commissioning

During commissioning, system-related data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. For Lenze devices, this is done via the codes.

The codes of the inverter and communication are saved non-volatilely as a data set in the memory module.

In addition, there are codes for diagnosing and monitoring the stations.

▶ Parameter reference (📖 65)

6.1 Before initial switch-on

Stop!

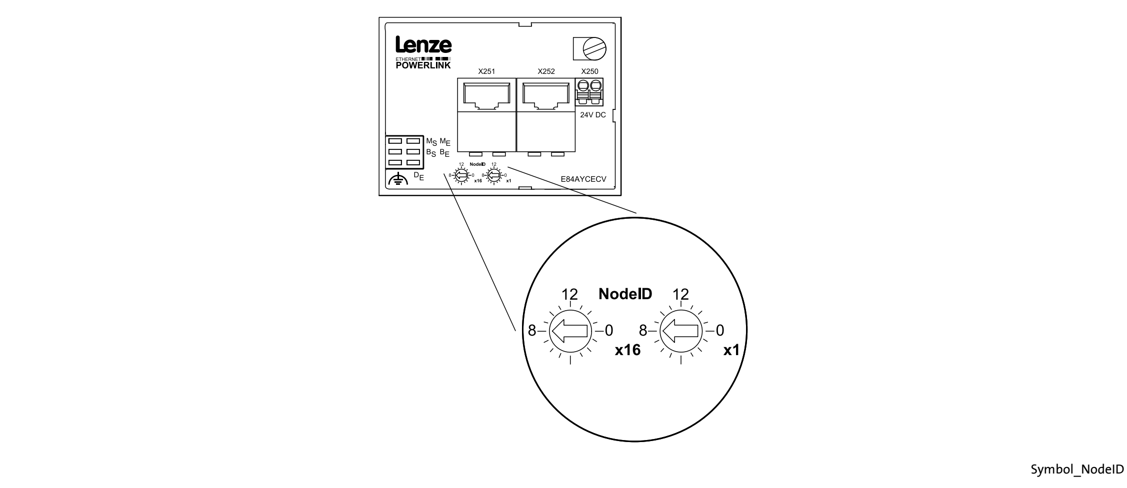
Before switching on the Inverter Drive 8400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Node address setting

Note!

- Each network node address must only be used once.
- The node address setting in the configuration software (e.g. »Engineer«) and the rotary encoder switches must be identical for each node.
- Adjustable address range: 1 ... 239
- Switch the voltage supply of the communication module off and on again to activate changed settings.

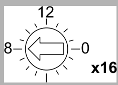
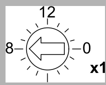
The node address is set via the two front panel rotary coding switches "Node ID".

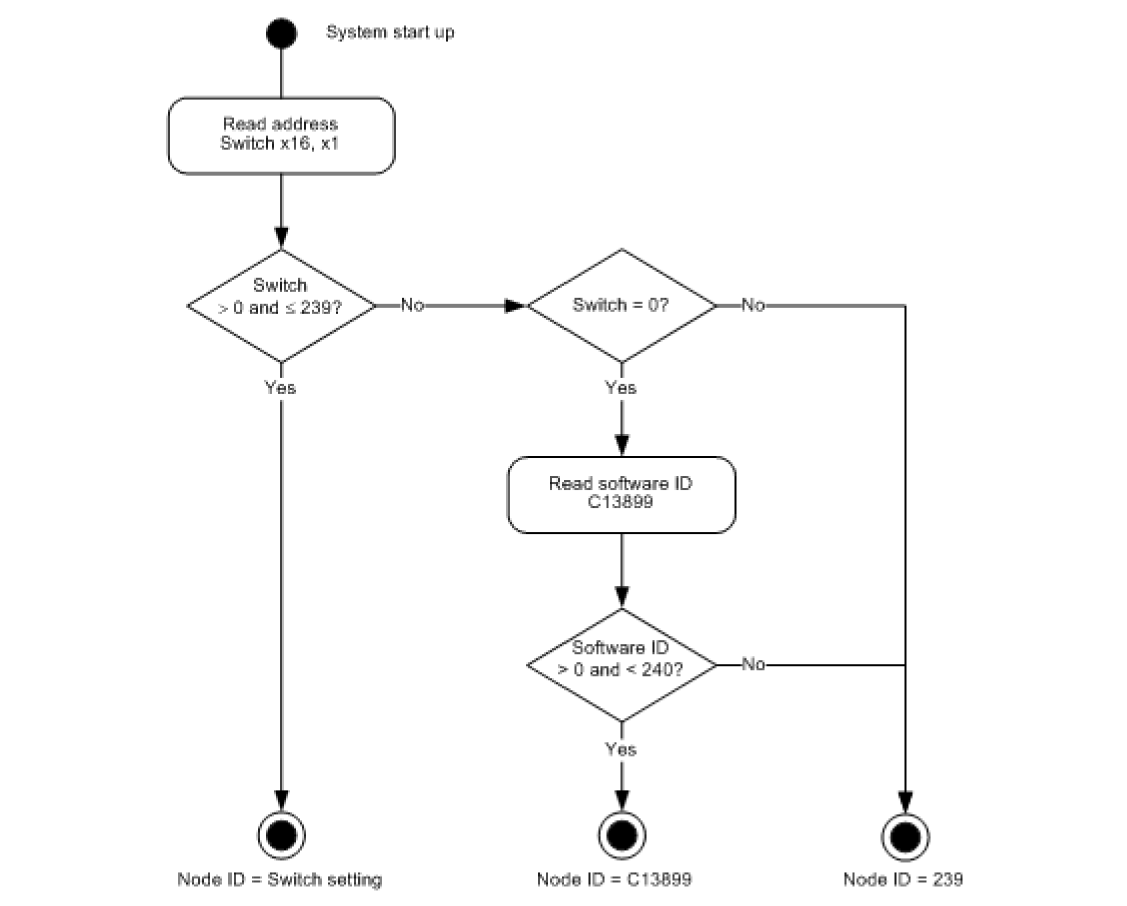


[6-1] Rotary coding switches "Node ID"

- The labelling of the rotary coding switches corresponds to the values for determining the node address.
- Valid addresses for the communication module in the EPL network:
  - 1 ... 100: Controlled node with Lenze module E94AYCEP as managing node
  - 1 ... 239: Controlled node with managing node of another manufacturer
- The corresponding Ethernet IP address of the communication module results from the address setting of the rotary coding switches:
  - IP address: 192.168.100.[*EPL address*]

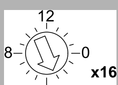
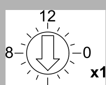
Possible settings of the rotary coding switches

Position of the rotary coding switches				Resulting node address
				
0	× 16	0	× 1	Value from code C13899
0 ... 14		1 ... 15		1 ... 239
15		0 ... 15		239



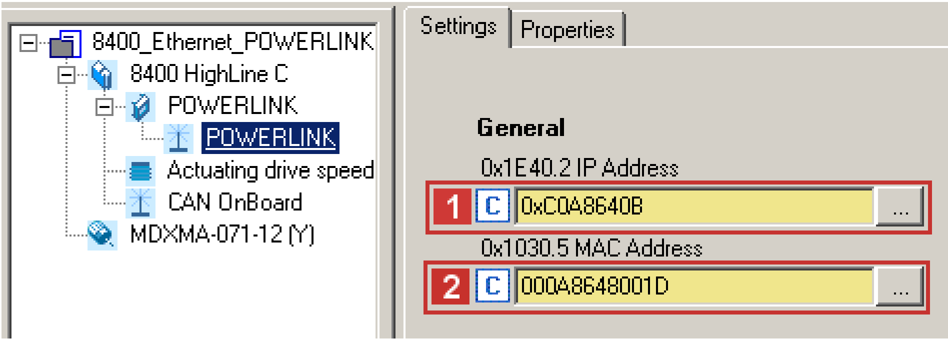
[6-2] Internal check of the rotary coding switch setting after the system start

Example: Setting of the node address "52"

Position of the rotary coding switches				Resulting node address
				
3	× 16	**4**	× 1	(3 × 16) + (4 × 1) = **52**

6.3 Setting and displays in the »Engineer«

The 1 **IP address** (C13000) and the 2 **MAC address** (C13004) are displayed on the **Settings** tab.



6.4 Initial switch-on

Documentation for the standard device

Observe the safety instructions and information on residual hazards.

Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Activating changed setting

Sequence for activating changed settings:

- Execute device command "11: Save all parameter sets" via standard device code **C00002**.
- Switch the voltage supply of the communication module off and on again.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

The restart protection is activated in the Lenze setting of the Inverter Drives 8400.

The restart behaviour of the inverter can be set via **C00142** ("auto-start option"):

- **C00142 = 9** (Lenze setting)
  - The inverter remains inhibited (even if the fault is no longer active).
  - Bit 0 (inhibit at power-on) and bit 3 (inhibit in the case of undervoltage) are set.
  - The drive restarts in a controlled mode through explicit controller enable: LOW-HIGH edge at digital input X4/RFR.
- **C00142 = 8** (Enabled)
  - In order to enable the device directly when switching it on, set bit 0 to zero (FALSE).
  - An uncontrolled restart of the drive is possible.

Initial switch-on and diagnosing

1. Switch on the controller and check whether it is ready for operation using the diagnostic LEDs of the communication module.
   - Red diagnostic LEDs must not be on.
   - The following signalling should be visible:

LED	Colour	Status	Meaning	
MS	green	On	**M**odule **S**tatus	▶ Module status displays (🕮 53)
EN	-	Off	**D**evice **E**rror	
X251	green	On	LED is lit if a connection to the node has been established. ▶ Status displays at the RJ45 sockets (X251, X252) (🕮 55)	
X252	green	On		

2. If not already done, load the EPL configuration into the inverter.
3. Repeat steps 1 and 2 for all EPL nodes.
4. Start network.

   The network starts automatically if you set the managing node **last**.

   Otherwise there are two options:
   - Switch off the network nodes and switch them on together or
   - execute a fault reset on the managing node (EPL address 240).
5. Check the EPL network again using the diagnostic LEDs of the communication module.

   The following signalling should be visible:

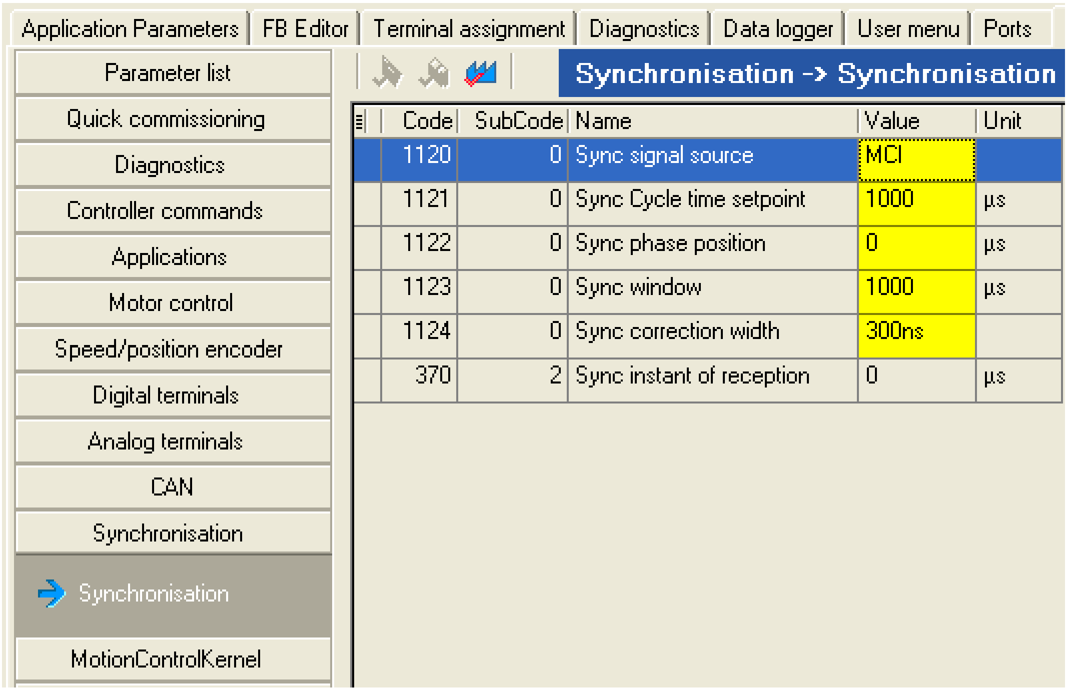
LED	Colour	Status	Meaning	
MS	green	On	**M**odule **S**tatus	▶ Module status displays (🕮 53)
ME	-	Off	**M**odule **E**rror	
BE	-	Off	**B**us **E**rror	
EN	-	Off	**D**evice **E**rror	
BS	green	On	**B**us **S**tatus	▶ Fieldbus status displays (🕮 54)
X251	green	On	LED is lit if a connection to the node has been established. ▶ Status displays at the RJ45 sockets (X251, X252) (🕮 55)	
X252	green	On		

6.5 Drive synchronisation

Examples of the use of drive synchronisation via POWERLINK:

- Implementation of the "electrical shaft"
- Specification of cyclic position information
- Exactly simultaneous data processing of several drives
- Optimisation of the processing times of cyclic process data (PDO).

Settings in the »Engineer«



- Set "MCI" as synchronisation source in code **C01120** of the inverter.
- The cycle time setpoint **C01121** and the phase position **C01122** are automatically calculated from the communication module. Default settings of these parameters are ignored.
- It is not necessary to adjust the sync window **C01123** and the sync correction width **C01124**.

Software manual/»Engineer« online help for the Inverter Drive 8400
Here you can find detailed information on drive synchronisation and the corresponding parameters.

6.6 Optimisation of networks

SDO bandwidth

Note!

The channel bandwidth should only be increased if the network is below capacity limit.

The SDO channel width (C13075) is the size of the asynchronous channel used for parameter setting and diagnostics. A higher value improves the transmission of large amounts of data (e.g. parameter downloads) and at the same time reduces the number of possible nodes.

Maximum time for device search

During the device search, the managing node has to wait until all controlled nodes have been found.

- Unless all controlled nodes are available in the defined EPL cycle time, the "EPL_BOOTUP_1" error message is generated. The managing node remains in this status.
- If the managing node has found all controlled nodes, it starts the network.

Due to machine or system-specific switch-on sequences, it may be required to adapt the following EPL objects:

- CN object **0x1F99**: NMT_CNBasicEthernetTimeout_U32 (C13078)
- MN object **0x1F89**: NMT_BootTime_REC

Note!

In order to avoid a too quick change to the "Basic Ethernet Mode" (📖 29), the value of the **0x1F99** object (NMT_CNBasicEthernetTimeout_U32) must be higher than the value of the **0x1F89** object (NMT_BootTime_REC) which is the case by default.

▶ Index table (📖 82)

7 Process data transfer

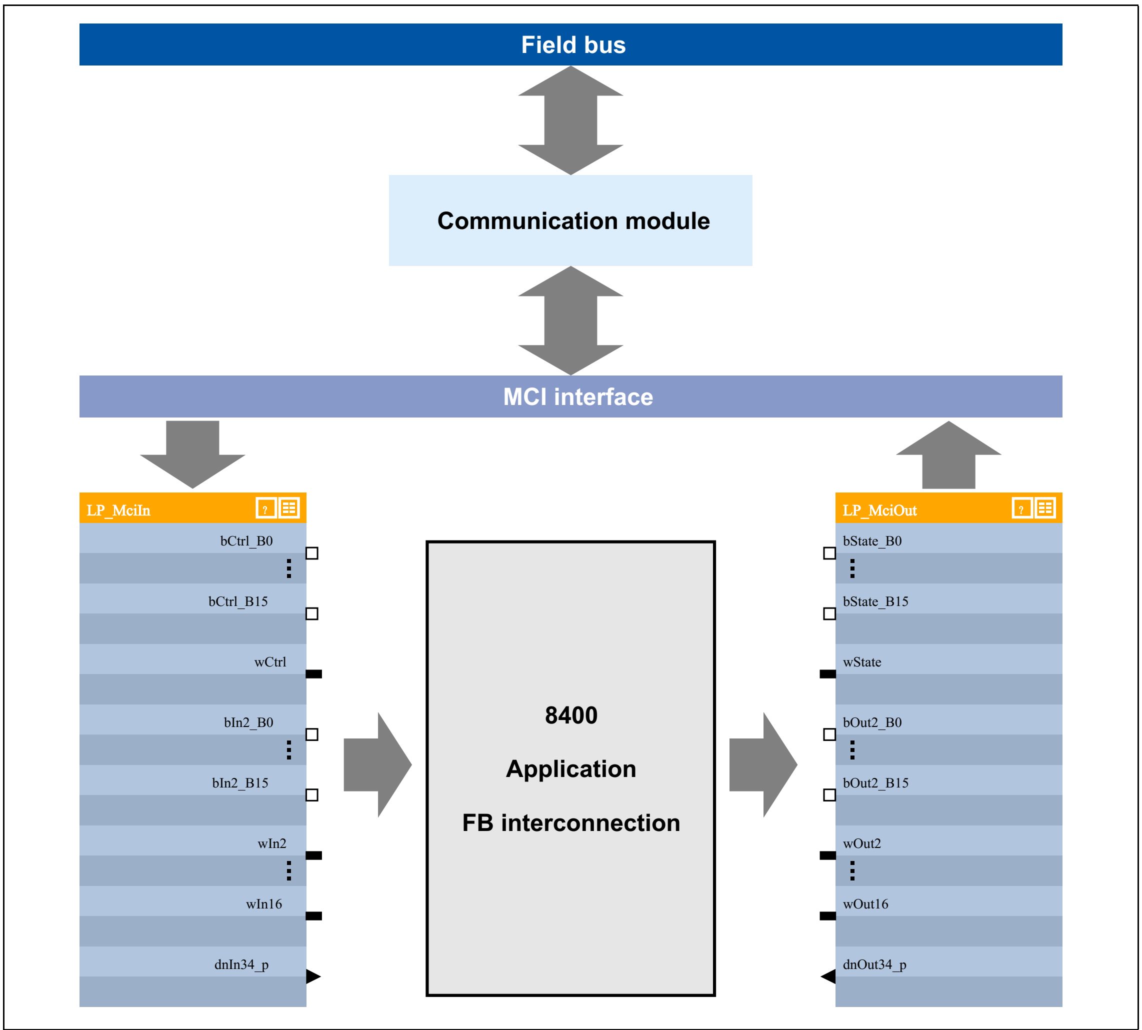
POWERLINK transmits process data, parameter data, configuration data and diagnostic data between the managing node and the controlled nodes. Depending on their time-critical behaviour, the data are transmitted via different communication channels.

- Process data are transmitted via the process data channel.
- Process data serve to control the Inverter Drive 8400.
- The transmission of process data is time-critical.
- Process data are cyclically transferred between the managing node and the controlled nodes (permanent exchange of current input and output data).
- The managing node can directly access the process data. In the PLC, for instance, the data are directly assigned to the I/O area.
- Up to 16 process data words (16 bit/word) can be exchanged per direction.
- Process data are not saved in the inverter.
- Process data are e.g. setpoints, actual values, control words, and status words.

7.1 Access to process data / PDO mapping

Process data (MCI-PDOs) are transferred via the MCI interface.

- A maximum of 16 words are exchanged per direction.
- The process data are accessed via the port blocks **LP_MciIn** and **LP_MciOut**. These port blocks are also referred to as process data channels.
- The port block **LP_MciIn** maps the received MCI-PDOs.
- The port block **LP_MciOut** maps the MCI-PDOs to be sent.
- The port/function block interconnection for the process data objects (PDO) is carried out with the Lenze »Engineer«.



[7-1] External and internal data transfer between the bus system, inverter, and application

Software manual / »Engineer« online help for the Inverter Drive 8400
Here you can find detailed information on the port/function block interconnection in the »Engineer« and on the port blocks.

Note!

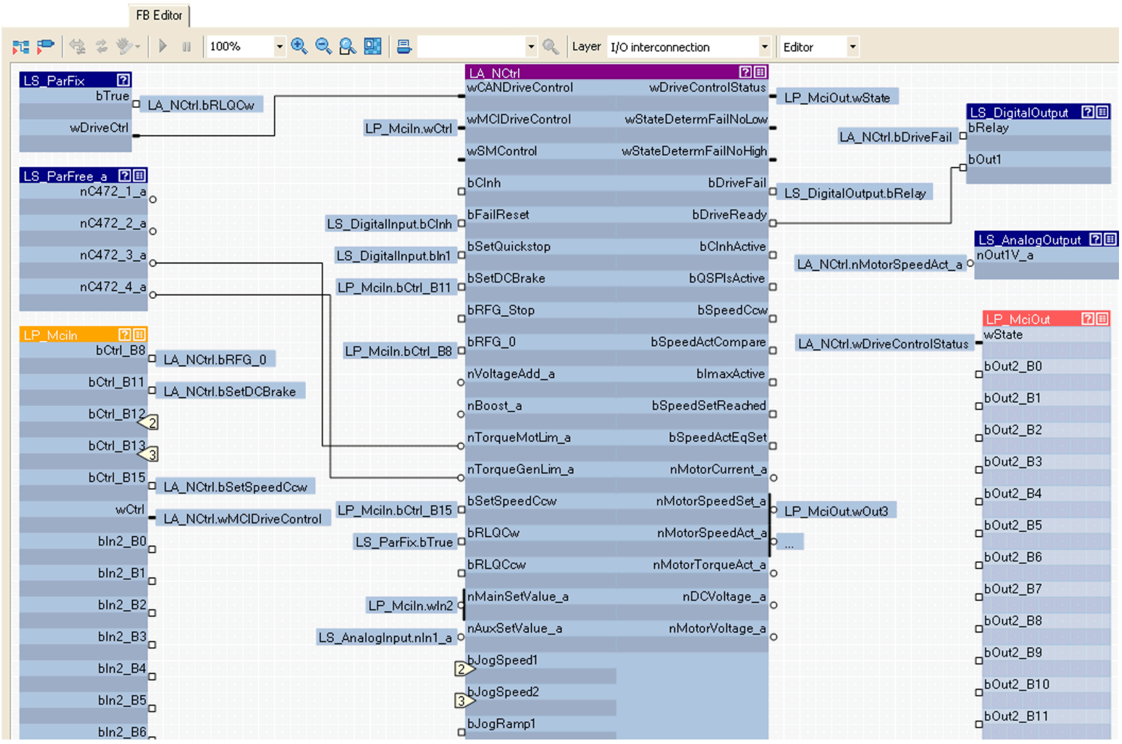
The »Engineer« screenshots shown on the following pages are only examples of the setting sequence and the resulting displays.

The data in the display fields may differ from the ones of your project.

7.2 Preconfigured port interconnection of the process data objects (PDO)

The preconfigured port interconnection of the process data objects can be activated by setting standard device code **C00007 = "40: MCI"**.

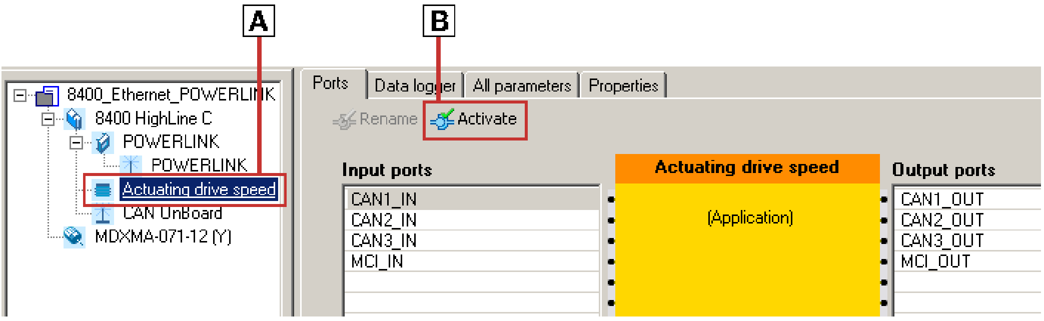
The function block editor (FB Editor) serves to display the **LP_MciIn** and **LP_MciOut** port blocks with the preconfigured interconnections:



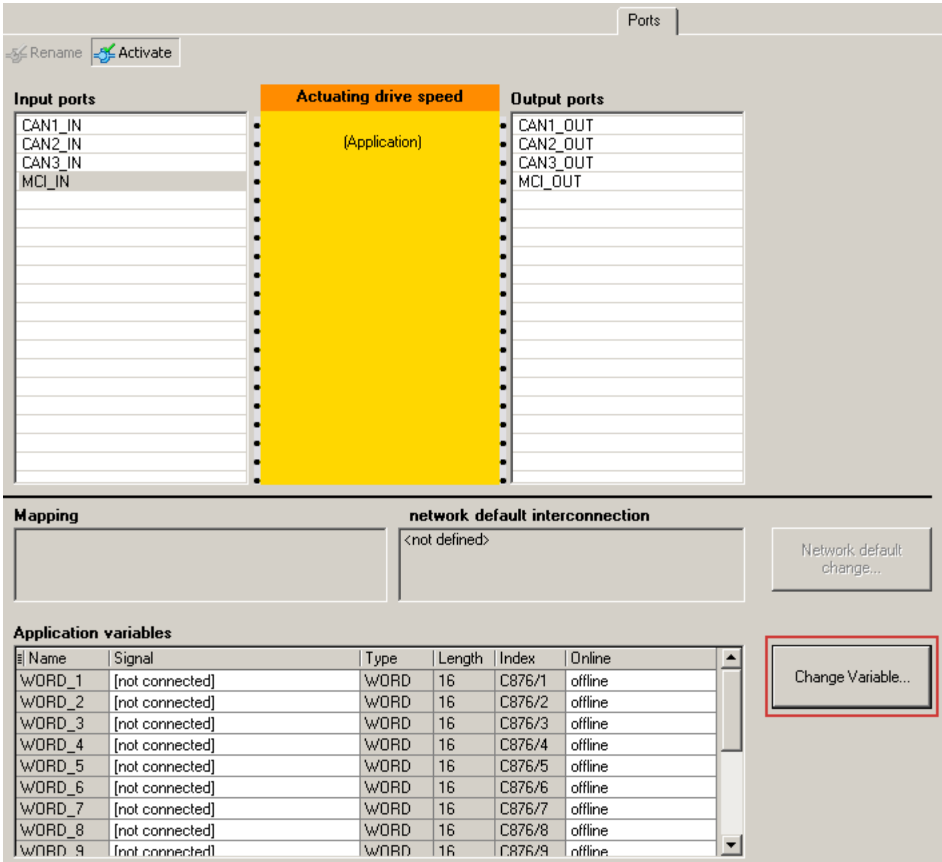
7.3 Freely configuring the port interconnection of the process data objects (PDO)

How to configure the port interconnection freely in the »Engineer«:

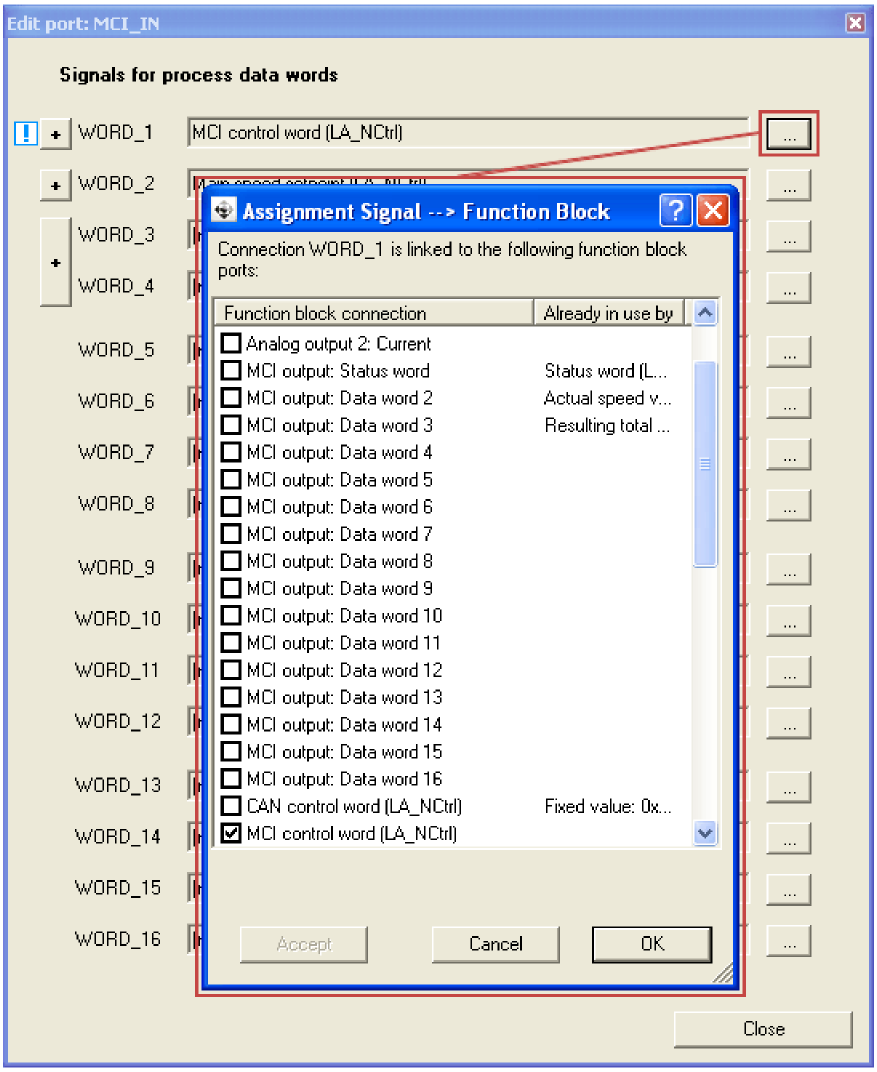
1. Go to the project view of the »Engineer« and select the application (A).
2. Select the port blocks **MCI_IN** or **MCI_OUT** on the **Ports** tab with a mouse-click and activate them with the **Activate** button (B).



3. Click the **Change Variable ...** button.

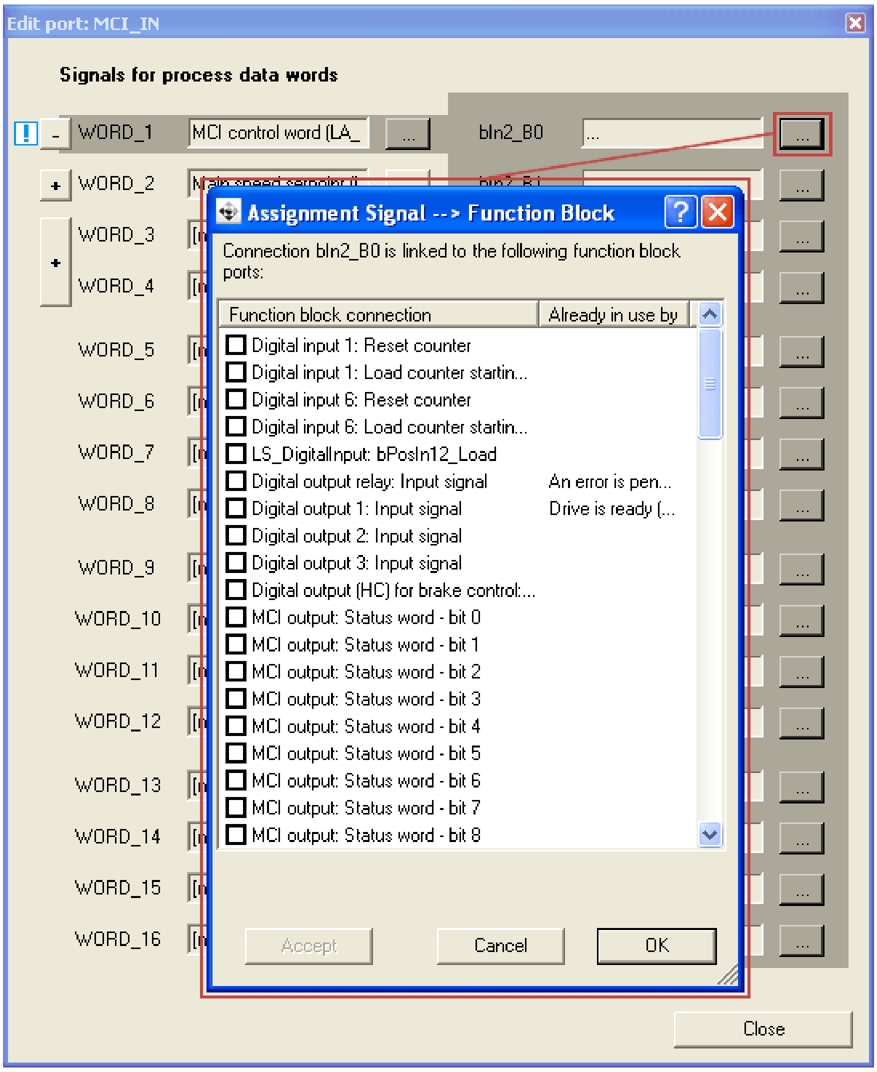


4. Via the [...] button, you can assign signals to the process data words in the *Assignment Signal --> Function Block* dialog window.
   → Select the signals and then confirm the selection with **OK**.



Furthermore you can assign signals to the individual control and status bits of the WORD_1 ... WORD_4 process data words via the + and ... buttons.

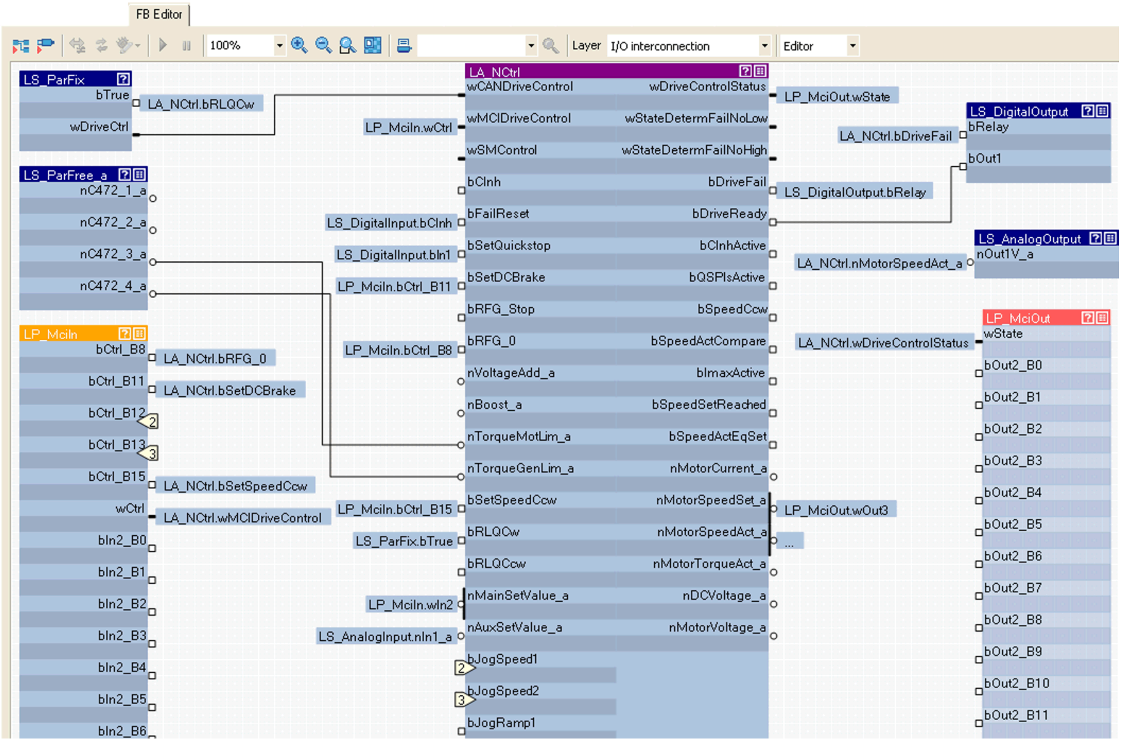
→ Select signals and confirm the selection by clicking the **OK** button.



Tip!

When the **LP_MciIn** and **LP_MciOut** port blocks are activated (see 1.), they will be visible in the FB Editor. Here you can also assign signals to the process data words.

In order to improve the clarity, unassigned signals can be hidden for each block.



8 Monitoring

Interruption of POWERLINK communication

An interruption of POWERLINK communication in the OPERATIONAL state, e.g. by cable break or failure of the managing node, is detected by the controlled node. The response to this interruption of communication depends on the following settings:

1. The watchdog monitoring time defined in the managing node is transferred to the controlled node when the POWERLINK communication is initialised.

   If the controlled node being in the OPERATIONAL state does not receive valid process data, the process data are handled according to the setting in C13885. (The last data sent by the managing node can either be used or set to zero.)

   If communication fails, the controlled node state changes to PRE-OPERATIONAL (see C13861) and the red LED "BE" is activated (see Fieldbus status displays (📖 54)).

   By default, there is no response in the controlled node.
2. To cause a response by the controlled node, you additionally have to ...
   - set a response of the controlled node (C13880/1).

9 Diagnostics

The LEDs on the front panel of the communication module serve to diagnose faults.

9.1 LED status displays

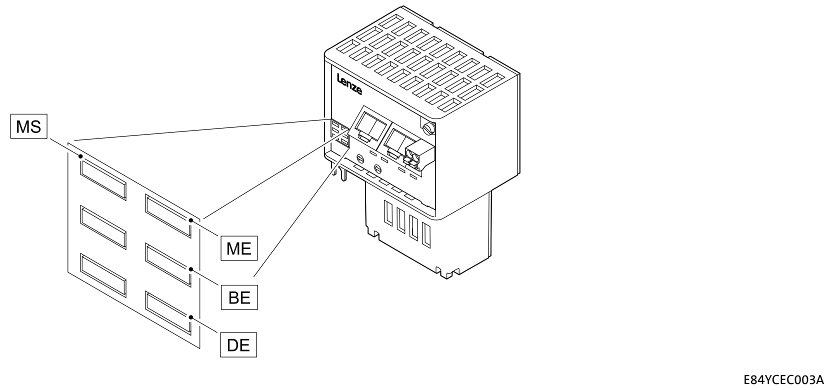
Note!

LED status displays for trouble-free operation:
- The LEDs **MS** (📖 53) and **BS** (📖 54) are permanently lit.
- The green LED at the RJ45 sockets X251 and X252 is permanently lit or blinking (📖 55).

The following status displays are distinguished:
- Module status displays (📖 53)
- Fieldbus status displays (📖 54)
- Status displays at the RJ45 sockets (X251, X252) (📖 55)

9.1.1 Module status displays

The LEDs **MS**, **ME, BE** and **DE** indicate the module status.

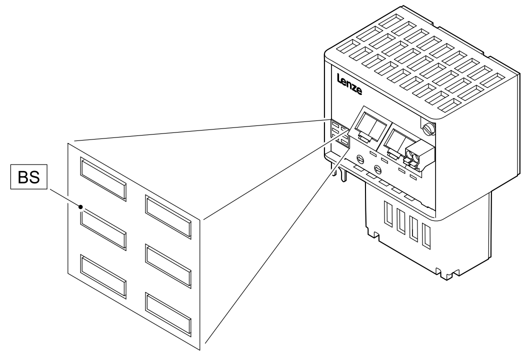


MS	**M**odule **S**tatus
ME	**M**odule **E**rror
BE	**B**us **E**rror
EN	**D**evice **E**rror

LED	Colour	Status	Description
MS	green	On	The communication module is supplied with voltage and is connected to the standard device.
		blinking	 The communication module is supplied with voltage, but has no connection to the standard device. (The standard device is either switched off, in the initialisation phase, or not available).
ME	red	On	An error has occurred in the communication module.
BE	red	On	A fieldbus error has occurred.
EN	red	On	The communication module is not accepted by the standard device. See notes given in the documentation for the standard device.

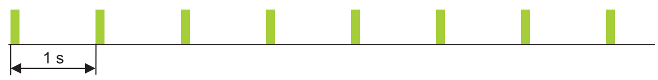
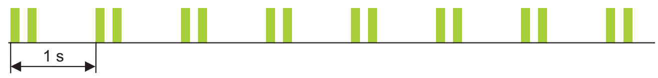
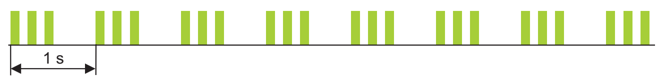
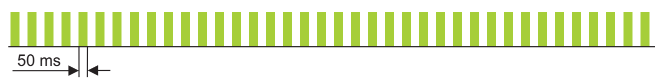
9.1.2 Fieldbus status displays

The **BS** LED indicates the fieldbus status.



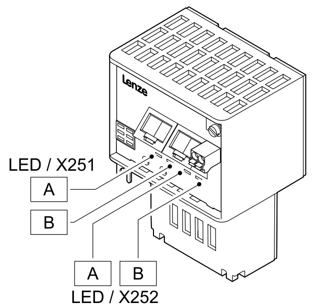
E84YCEC003B

BS	**B**us **S**tatus

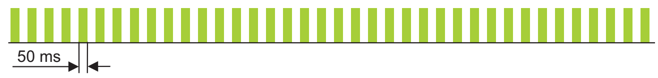
LED	Colour	Status	Description
BS	green	Off	The communication module is not active on the fieldbus or is in the INIT state.
		blinking	 EPL network is in the initialisation phase. EPL status: NMT_CS_PREOPERATIONAL_1
		blinking	 EPL network is in the initialisation phase with cyclic traffic. EPL status: NMT_CS_PREOPERATIONAL_2
		blinking	 EPL node is waiting for the start signal. EPL status: NMT_CS_READY_TO_OPERATE
		blinking	 The EPL node has not found a managing node and is in Basic Ethernet Mode (📖 29). EPL status: NMT_CS_BASIC_ETHERNET
		blinking	 EPL node is in the "Stopped" status (waiting for disconnection). EPL status: NMT_CS_STOPPED
		On	 EPL node is in the operating phase. EPL status: NMT_CS_OPERATIONAL

9.1.3 Status displays at the RJ45 sockets (X251, X252)

The LEDs at the RJ45 sockets **X251** and **X252** indicate the POWERLINK connection status.



E84YCER005

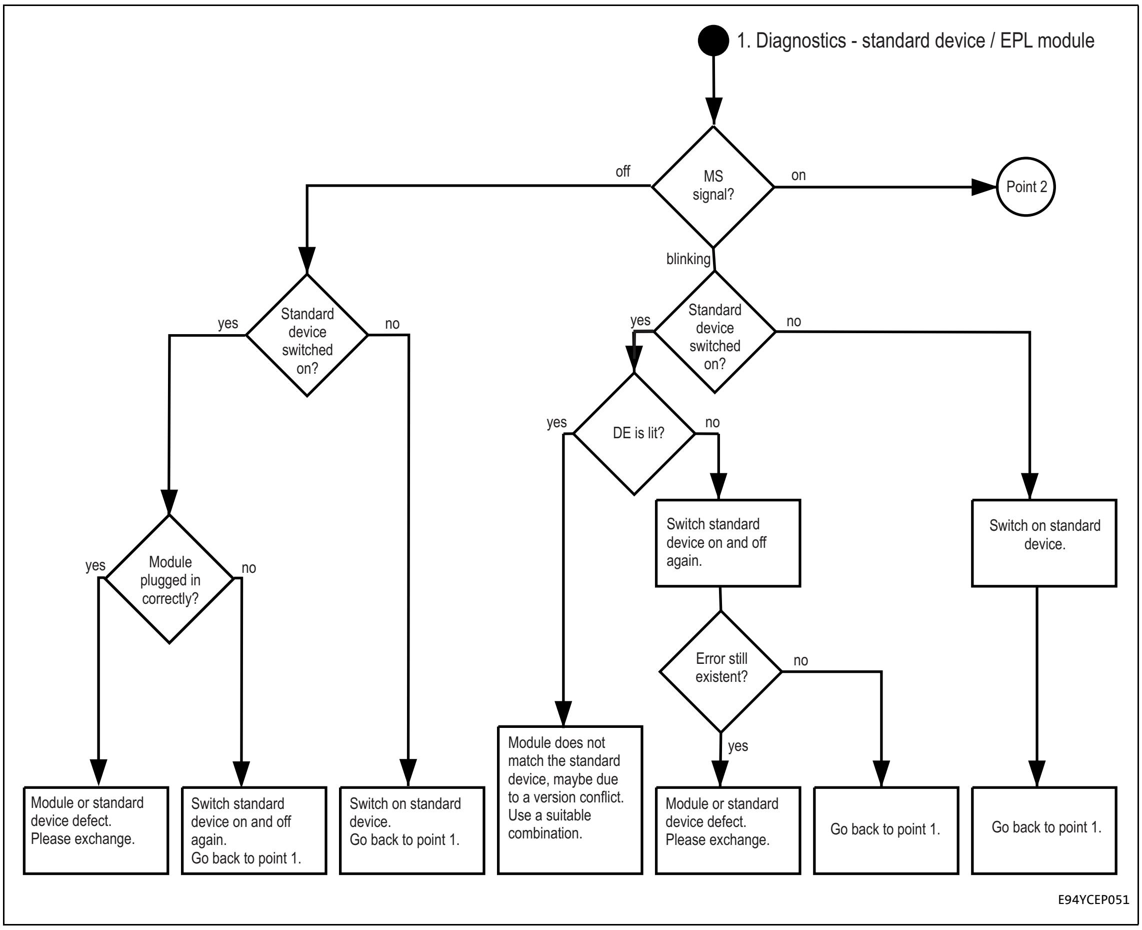
LED	Colour	Status	Description
A	green	Off	No POWERLINK connection
		On	 POWERLINK connection has been established.
		Flickers	 Data communication of POWERLINK connection is active.
B	red	On	 POWERLINK collision has occurred.

9.2 Troubleshooting with module signalling

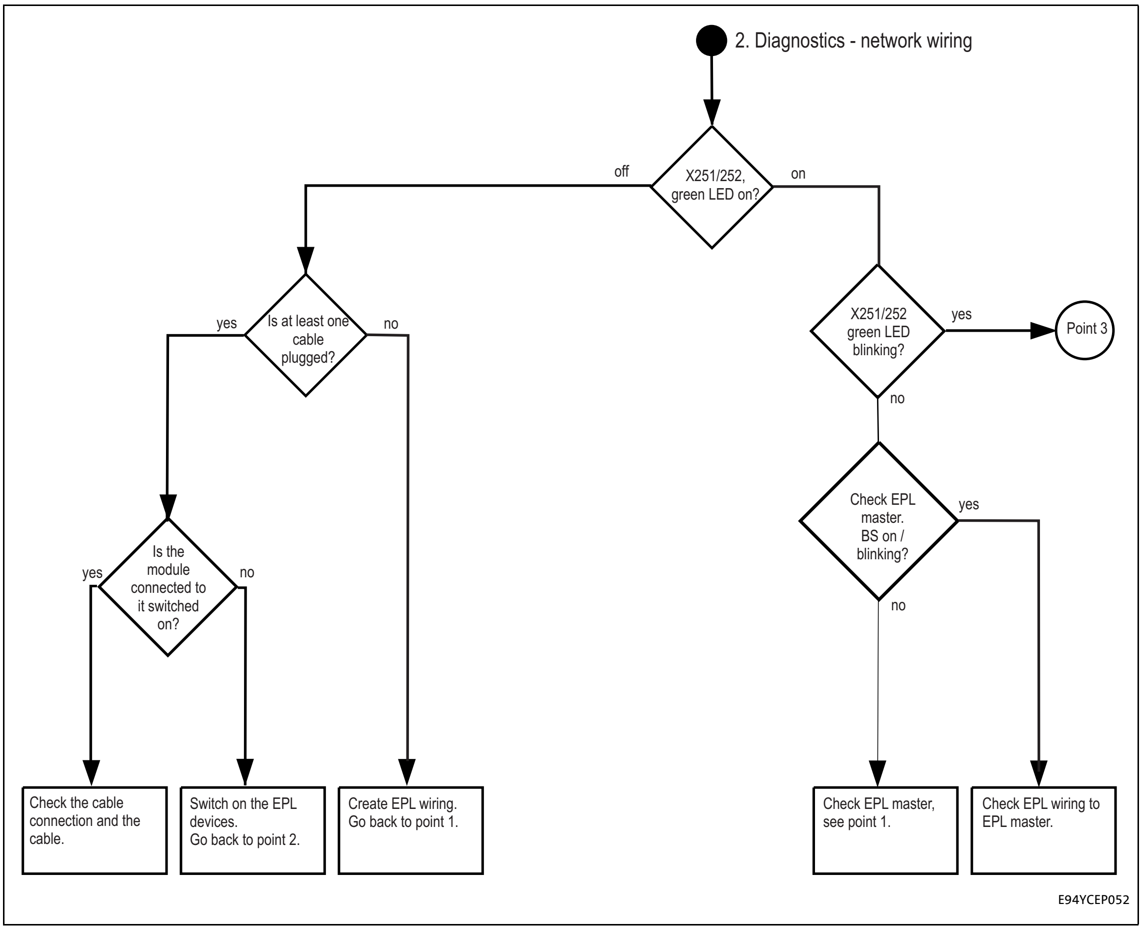
In the following, troubleshooting via the module signals will be described.

Please particularly observe ...

- the basic readiness for operation between the standard device and the communication module;
- the network wiring.



[9-1] Diagnostics - standard device / POWERLINK module

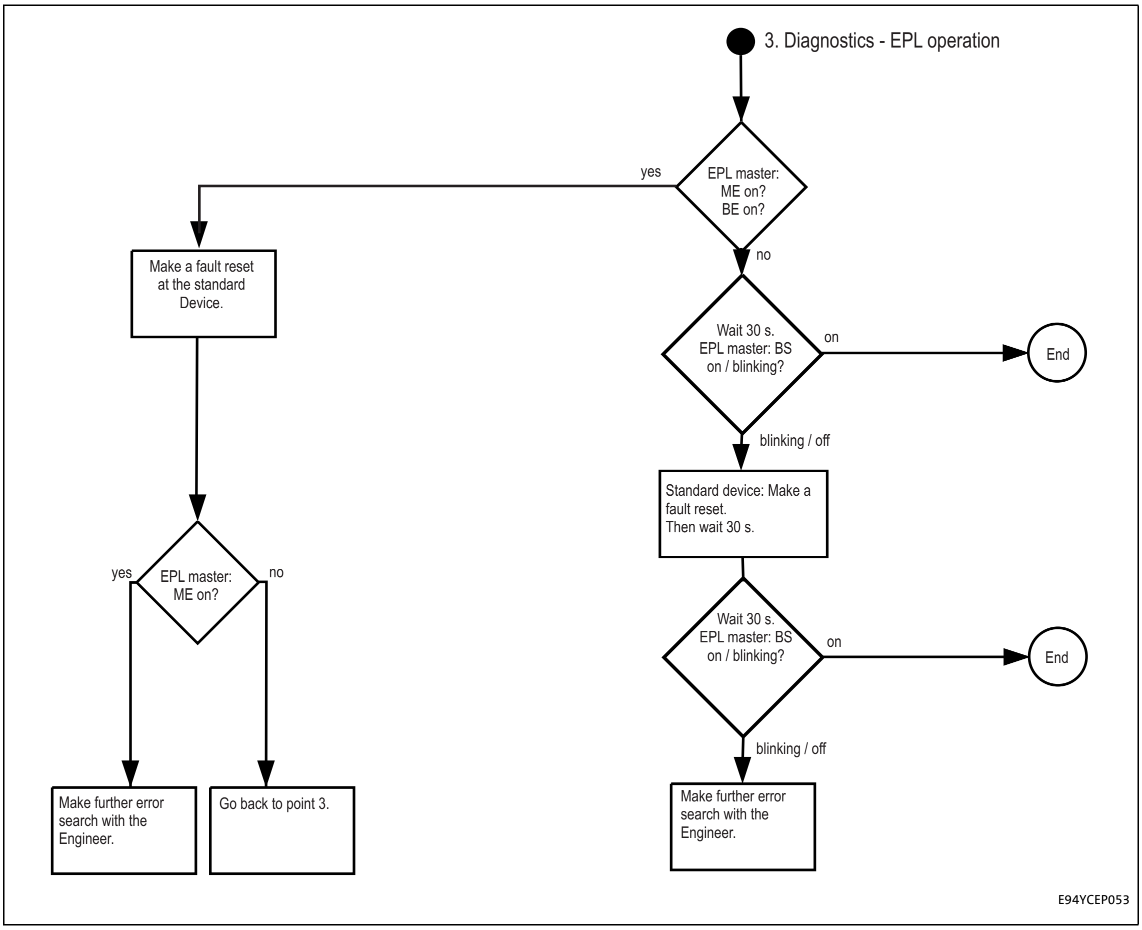


[9-2] Diagnostics - network wiring

Note!

The **BS** (*Bus State*) LED refers to the Lenze communication module E94AYCEP as managing node.

This status display applies analogously to the managing nodes of other manufacturers.



[9-3] Diagnostics - POWERLINK operation

Note!

The LEDs **ME** (*Module Error*), **BE** (*Bus Error*) and **BS** (*Bus State*) refer to the Lenze communication module E94AYCEP as managing node.

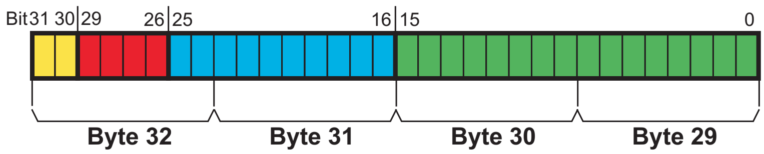
These status displays apply analogously to the managing nodes of other manufacturers.

9.3 Diagnostic data

- Pending diagnostic data are signalled from the controlled node to the managing node by means of an emergency message.
- Code C13887 serves to suppress sending emergency messages to the managing node. You can select which type of error is to be suppressed.
- Errors and warnings in the Inverter Drive 8400 and the plugged communication module are sent to the managing node as extended diagnostic messages.
- The diagnostic data are visible via the PLC Engineering software.

Bytes	Meaning	Value [hex]
1 ... 6	Diagnostic block header	0x0010 001C 0100
7 ... 8	Alarm type	0x0001 (diagnosis)
9 ... 12	API (Application Programming Interface)	0x0000 0000
13, 14	Slot number	0x0001 / 0x0002
15, 16	Subslot number	0x0001
17 ... 20	Module ID	ID according to module
21 ... 24	Submodule number	ID according to module
25, 26	Alarm specification	0xB000
27, 28	User structure ID	0x0001
29 ... 32	Error code	

Error code of Inverter Drive 8400



[9-4] Error code of Inverter Drive 8400

- The error code of the Inverter Drive 8400 can be found in bytes 29 ... 32 of the diagnostic message.
- In the logbook and in code **C00165** of the standard device, the error number is shown in the following syntax in order to facilitate the readability:
  [error type].[error subject area no.].[error ID]

Example

Diagnostic message of the error "EPL: State OPERATIONAL lost [0x01bc8131]":

Bit assignment (Bytes 29 ... 32)	Information	Values from the example
Bit31 30 0 0	Reserved	0
Bit 29 26 0 X X X	Error type / error response	Setting via C13880/1: • 0: No error response • 1: No Response • 3: Quick stop • 4: Warning locked • 6: Information
Bit25 16 0 1 1 0 1 1 1 1 0 0	Error subject area	**0x01bc** (444_{dec})
Bit 15 0 1 0 0 0 0 0 0 1 0 0 1 1 0 0 0 1	Error ID	**0x8131** (33073_{dec})

Software manual/»Engineer« online help for Inverter Drive 8400

Detailed information on the error codes is provided here.

10 Error messages

[ALinkKeywords]
Error messages

This chapter supplements the error list in the software manual and the »Engineer« online help for the Inverter Drive 8400 by the error messages of the communication module.

Software manual/»Engineer« online help for the Inverter Drive 8400

Here you will find general information on diagnostics & fault analysis and on error messages.

[Search-Keywords]
Error messages

10.1 Short overview of the POWERLINK error messages

The following table contains all error messages of the communication module in numerical order of the error numbers. Moreover the preset error response and - if applicable - the parameters for setting the error response are specified.

Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error no. [hex]	Subject area no. [dec]	Error no. [dec]	Error text	Error type (Error response)	Adjustable in
0x01bc3100	444	12544	EPL: Lost connection to 8400 target	1: No Response	-
0x01bc5531	444	21809	EPL: NV memory: No access	1: No Response	-
0x01bc5532	444	21810	EPL: NV memory: Read error	1: No Response	-
0x01bc5533	444	21811	EPL: NV memory: Write error	1: No Response	-
0x01bc6010	444	24592	EPL: Restart after watchdog reset	1: No Response	-
0x01bc6011	444	24593	EPL: Watchdog reset	1: No Response	-
0x01bc6100	444	24832	EPL: Software error	1: No Response	-
0x01bc6101	444	24833	EPL: Fatal software error	1: No Response	-
0x01bc6110	444	24848	EPL: Invalid PDO mapping	1: No Response	-
0x01bc641f	444	25631	EPL: Invalid parameter set	1: No Response	-
0x01bc6420	444	25632	EPL: Default setting loaded	1: No Response	-
0x01bc6430	444	25648	EPL: Invalid module configuration	1: No Response	-
0x01bc8131	444	33073	EPL: OPERATIONAL status left	1: No Response	C13880/1
0x01bc8261	444	33377	EPL: Invalid address selected	1: No Response	-
0x01bc8265	444	33381	EPL: Synchronisation of MN lost	0: None	C13880/2
0x01bc8266	444	33382	EPL: Frame error (CRC)	0: None	C13880/3

10.2 Possible causes and remedies

In this chapter, all error messages of the communication module are listed in numerical order of the error numbers. Possible causes and remedies and responses to the error messages are described in detail.

▶ Short overview of the POWERLINK error messages (📖 61)

EPL: Lost connection to 8400 target [0x01bc3100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
MCI communication to Inverter Drive 8400 is interrupted. • Inverter Drive 8400 is switched off. • The communication module is not connected correctly at the MCI slot of the Inverter Drive 8400.	• Switch on Inverter Drive 8400. • Check screwed connection of the communication module at the MCI slot of the Inverter Drive 8400. • Send the communication module and Inverter Drive 8400 together with a description of the fault to Lenze.

EPL: NV Memory: No access [0x01bc5531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set in memory module via standard device was not successful.	Download application again (including module).

EPL: NV Memory: Read error [0x01bc5532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Parameter in the memory module could not be read.	Download application again (including module).

EPL: NV Memory: Write error [0x01bc5533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Parameter in the memory module could not be written.	Download application again (including module).

EPL: Restart after watchdog reset [0x01bc6010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send the communication module and a description of the fault to Lenze.

EPL: Watchdog reset [0x01bc6011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send the communication module and a description of the fault to Lenze.

EPL: Software error [0x01bc6100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send the communication module and a description of the fault to Lenze.

EPL: Fatal software error [0x01bc6101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send the communication module and a description of the fault to Lenze.

EPL: PDO Mapping invalid [0x01bc6110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Invalid mapping configuration.	Correct the mapping configuration.

EPL: Invalid parameter set [0x01bc641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Download application again (including module).

EPL: Factory settings loaded [0x01bc6420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set in memory module via standard device was not successful.	Download application again (including module).

EPL: Invalid module configuration [0x01bc6430]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Module configuration is faulty.	Check and correct module configuration.

EPL: State OPERATIONAL lost [0x01bc8131]

Response (Lenze setting printed in bold)	**Setting:** C13880/1
☒ No reaction ☐ System fault ☒ **Fault** ☒ Trouble ☒ Quick stop by trouble ☒ WarningLocked ☐ Warning ☒ Information	

Cause	Remedy
Data exchange via POWERLINK has been terminated. • Also see the chapter "Interruption of POWERLINK communication" (📖 51).	• Check the network cable (plug) and replace it if necessary. • Connect the network cable to the Ethernet POWERLINK terminals X251 or X252 and continue to check the status of the managing node.

EPL: Invalid address selected [0x01bc8261]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
An invalid IP address has been assigned by the managing node via POWERLINK or set as node ID.	• Ensure that a valid IP address is assigned by the managing node via POWERLINK. • Set valid node ID. ▸ Node address setting (📖 37)

EPL: Synchronisation lost from MN [0x01bc8265]

Response (Lenze setting printed in bold)	**Setting:** C13880/2
☒ No reaction ☐ System fault ☐ Fault ☒ **Trouble** ☒ Quick stop by trouble ☒ Warning locked ☐ Warning ☒ Information	

Cause	Remedy
In the controlled node, the synchronisation cycle of the managing node has failed. The controlled node changes automatically to the NMT_CS_PREOPERATIONAL_1 state and waits for a new run-up by the managing node.	• Check network cable and components (failure of managing node, router). • Restart managing node if required.

EPL: Telegram error detected (CRC) [0x01bc8266]

Response (Lenze setting printed in bold)	**Setting:** C13880/3
☒ No reaction ☐ System fault ☐ Fault ☒ **Trouble** ☒ Quick stop by trouble ☒ Warning locked ☐ Warning ☒ Information	

Cause	Remedy
Faulty Ethernet frames (CRC error) have been detected. Possible causes: • A device in the network is not EPL-compliant. • EMC interference is too strong.	• Check if a non-EPL-compliant device is in the network (e.g. diagnostics PC). • Reduce EMC interference on the network or use an additional shield connection.

11 Parameter reference

[ALinkKeywords] Parameters

This chapter supplements the parameter list and the table of attributes for the Inverter Drive 8400 contained in the software manual and in the »Engineer« online help by the parameters of the communication module E84AYCEC (POWERLINK).

Software manual/»Engineer« online help for Inverter Drive 8400

Here you will find general information on parameters.

[Search-Keywords] Hlp_Para

11.1 Parameters of the communication module

This chapter lists the parameters of the E84AYCEC communication module (POWERLINK) in numerically ascending order.

C13000

Parameter | Name: **C13000 | 0x1E40.2 IP Address**			Data type: UNSIGNED_32 Index: 11575_d = $2D37_h$
The code displays the IP address of the communication module. • The IP address is derived from the EPL node address (Node ID): **192.168.100.[Node ID]** • The node ID is derived from the coding switch position or from C13899. ▶ Node address setting (📖 37)			
Display range (min. value | unit | max. value)			
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)	
Subcodes			**Info**
C13000/1			IP address • Lenze setting: 0xC0A86401 = 192.168.100.1
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13001

Parameter | Name: **C13001 | 0x1E40.3 Subnet Mask**			Data type: UNSIGNED_32 Index: 11574_d = $2D36_h$
The code declares the IP subnet mask which restricts the directly addressable IP address range (i.e. without a gateway in the EPL segment of the routers). The value 255.255.255.0 (0xFFFFFF00) is always assigned to the subnet mask in one segment.			
Display range (min. value | unit | max. value)			
255.0.0.0 (0xFF000000)		255.255.255.0 (0xFFFFFF00)	
Subcodes			**Info**
C13001/1			EPL IP Subnet Mask • Lenze setting: 0xFFFFFF00 = 255.255.255.0
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13003

Parameter | Name: **C13003 | 0x1E40.5 IP Address Router**			Data type: UNSIGNED_32 Index: 11572_d = $2D34_h$
The code declares the IP address of the router which connects the EPL segment to the higher-level network. The standard entry corresponds to the standard router address of the EPL specification: **→ 192.168.100.254** Permissible entries replace the lowest-order byte of the standard entry with the EPL address of the node which has the function of a router.			
Setting range (min. value | unit | max. value)			
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)	
Subcodes	**Lenze setting**		**Info**
C13003/1	3232261374		IP address router • Lenze setting: 0xC0A864FE = 192.168.100.254
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13004

Parameter | Name: **C13004 | 0x1030.5 MAC Address**	Data type: OCTET_STRING Index: 11571_d = $2D33_h$
The code indicates the physical address (MAC address) of the Ethernet interface of the communication module. When the communication module is produced, the MAC address is assigned unequivocally worldwide and provides addressing on the lowest level.	
Subcodes	**Info**
C13004/1	MAC Address (octet string[6]) 00-0A-86-xx-yy-zz • 00-0A-86 = Lenze • xx-yy-zz = consecutive number
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13028

Parameter | Name: **C13028 | 0x1F81 Node Assignment CN**		Data type: UNSIGNED_32 Index: 11547_d = $2D1B_h$
The code declares the controlled nodes 1 ... 100 and their properties.		
Value is bit-coded:		
Bit 0 ... 31	see below	
Subcodes	**Lenze setting**	**Info**
C13028/1	0	
...	..	
C13028/100	0	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

The describing bit field has the following structure:

Bit	Value	Description
0 (LSB)	0	EPL node with this ID does not exist.
	1	EPL node with this ID exists.
1	0	EPL node with this ID is not a controlled node.
	1	EPL node with this ID is a controlled node.
2	0	On detection of a booting controlled node, the application will not be informed.
	1	On detection of a booting controlled node, the application will be informed and the controlled node will be started.
3	0	Optional controlled node
	1	Obligatory controlled node
4	0	The managing node is allowed to send reset commands.
	1	The managing node must not send any reset command.
5	0	Software version verification of the controlled node is not required.
	1	Software version verification of the controlled node is required.
6	0	Automatic application software update is not allowed.
	1	Automatic application software update is allowed.
7	-	Reserved / No function
8	0	Isochronously accessed controlled node
	1	Asynchronously accessed controlled node (bit 9 is irrelevant)
9	0	Continuously accessed controlled node
	1	Multiplex EPL nodes are supported.
10	0	No "POWERLINK to standard Ethernet" router function.
	1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	0	No "POWERLINK to fieldbus" router function.
	1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	0	The managing node does not send any PRes frames
	1	The managing node sends PRes frames
13 ... 30	-	Reserved / No function
31 (MSB)	0	Bits 0 ... 30 inhibited
	1	Bits 0 ... 30 enabled

C13029

Parameter | Name: **C13029 | 0x1F81 Node Assignment**		Data type: UNSIGNED_32 Index: 11546_d = $2D1A_h$
The code declares managing node, diagnostic device, and router, and describes their properties.		
Value is bit-coded:		
Bit 0 ... 31	see below	
Subcodes	**Lenze setting**	**Info**
C13029/1	2147483661	EPL node assignment managing node Lenze setting (hex): 0x8000000D
C13029/2	0	EPL node assignment diagnostic device Lenze setting (hex): 0x00000000
C13029/3	2147483655	EPL node assignment router Lenze setting (hex): 0x80000007
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

The describing bit field has the following structure:

Bit	Value	Description
0 (LSB)	0	EPL node with this ID does not exist.
	1	EPL node with this ID exists.
1	0	EPL node with this ID is not a controlled node.
	1	EPL node with this ID is a controlled node.
2	0	On detection of a booting controlled node, the application will not be informed.
	1	On detection of a booting controlled node, the application will be informed and the controlled node will be started.
3	0	Optional controlled node
	1	Obligatory controlled node
4	0	The managing node is allowed to send reset commands.
	1	The managing node must not send any reset command.
5	0	Software version verification of the controlled node is not required.
	1	Software version verification of the controlled node is required.
6	0	Automatic application software update is not allowed.
	1	Automatic application software update is allowed.
7	-	Reserved / No function
8	0	Isochronously accessed controlled node
	1	Asynchronously accessed controlled node (bit 9 is irrelevant)
9	0	Continuously accessed controlled node
	1	Multiplex EPL nodes are supported.
10	0	No "POWERLINK to standard Ethernet" router function.
	1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	0	No "POWERLINK to fieldbus" router function.
	1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	0	The managing node does not send any PRes frames
	1	The managing node sends PRes frames
13 ... 30	-	Reserved / No function
31 (MSB)	0	Bits 0 ... 30 inhibited
	1	Bits 0 ... 30 enabled

C13040

Parameter | Name: **C13040 | 0x1F82 Feature Flags**		Data type: UNSIGNED_32 Index: 11535_d = $2D0F_h$
The code displays the Ethernet POWERLINK features implemented by the EPL node. Lenze standard value: 0x00000207		
Value is bit-coded:		
Bit 0 ... 31	see below	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

The describing bit field has the following structure:

Bit	Value	Description
0 (LSB)	0	Asynchronous access
	1	Isochronous access
1	0	No SDO by UDP/IP
	1	SDO by UDP/IP
2	0	No SDO by EPL "ASnd"
	1	SDO by EPL "ASnd"
3	0	No SDO integrated in PDO
	1	SDO integrated in PDO
4	0	No "NMT Info Services"
	1	"NMT Info Services" supported
5	0	No extended "NMT State Commands"
	1	Extended "NMT State Commands" supported
6	0	No dynamic PDO mapping
	1	Dynamic PDO mapping supported
7	-	Reserved / No function
8	0	No configuration manager function
	1	Configuration manager function
9	0	Only isochronous cyclic access permitted.
	1	Isochronous multiplexed access possible.
10	0	No address assignment via software
	1	Address assignment via software
11	-	Reserved / No function
12	0	No "POWERLINK to standard Ethernet" router function.
	1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
13	0	No "POWERLINK to fieldbus" router function.
	1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
14 ... 31 (MSB)	-	Reserved / No function

C13060

Parameter | Name: C13060 | 0x1006 Cycle Time	Data type: UNSIGNED_32 Index: 11515_d = $2CFB_h$
The code defines the length of the EPL cycle in µs. • In the configured state, this code must have an identical value in all EPL nodes. • The selected value must correspond to the actual bus cycle time so that the internal monitoring functions work correctly.	

Selection list (Lenze setting printed in bold)	
400	400
500	500
600	600
800	800
1000	**1000**
2000	2000
3000	3000
4000	4000
5000	5000
6000	6000
7000	7000
8000	8000
9000	9000
10000	10000
11000	11000
12000	12000
13000	13000
14000	14000
15000	15000
16000	16000
17000	17000
18000	18000
19000	19000
20000	20000

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13066

Parameter | Name: **C13066 | 0x1F8D PResPayloadLimit RPDO CN**			Data type: UNSIGNED_16 Index: 11509_d = $2CF5_h$
The code defines the reserved user data length of the PRes telegrams for controlled nodes 1 ... 100. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C13028. The subcodes describe the PRes telegrams received. The value must be within the range of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the EPL nodes must be identical to the corresponding C13072 entries. C13066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	
Subcodes	**Lenze setting**		**Info**
C13066/1	0 byte		
...	...		
C13066/100	0 byte		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13067

Parameter | Name: **C13067 | 0x1F8D PResPayloadLimit RPDO**			Data type: UNSIGNED_16 Index: 11508_d = $2CF4_h$
The code defines the reserved user data length of the PRes frames for managing node, diagnostic device or router. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C13028. The subcode describes the PRes telegrams received. The value must be within the limits of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the nodes must be identical to the corresponding C13072 entries. C13067 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	
Subcodes	**Lenze setting**		**Info**
C13067/1	0 byte		Managing node
C13067/2	0 byte		Diagnostics device
C13067/3	0 byte		router
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13071

Parameter | Name: **C13071 | 0x1F98.4 PReqPayloadLimit RPDO**			Data type: UNSIGNED_16 Index: 11504_d = $2CF0_h$
The code defines the maximum data size to be received by the controlled node via PReq for the current network configuration. C13071 is an upper limit value for the entire size of the PDO mapping for the PReq telegram. In the configured state, the value must be identical to the entry for a response valid for the EPL node.			
Setting range (min. value | unit | max. value)			**Lenze setting**
36	Byte	1490	**36 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13072

Parameter | Name: C13072 | 0x1F98.5 PResPayloadLimit TPDO			Data type: UNSIGNED_16 Index: 11503_d = $2CEF_h$
The code defines the maximum data size to be sent by the EPL node for the current network configuration. The PDO mapping is allowed to assign data with a total size greater than or equal to C13072. **In the configured state, C13072 must be identical to the entry in C13066 valid for the EPL node.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
36	Byte	1490	**36 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13074

Parameter | Name: C13074 | 0x1F98.7 Multiplex Cycle Counter			Data type: UNSIGNED_8 Index: 11501_d = $2CED_h$
This code serves to define the maximum number of the multiplexed cycles. If, for instance, the value "3" is entered, the multiplexed cycle is repeated every three cycles. Within one multiplexed cycle, the nodes are queried according to the value in code C13079. If, for instance, the value "2" is entered for a node in C13079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	255	**0 cycles**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13075

Parameter | Name: C13075 | 0x1F98.8 SDO Channel Width (AsyncMTU)			Data type: UNSIGNED_16 Index: 11500_d = $2CEC_h$
The code defines the maximum user data size of asynchronous frames. Protocol-specific headers for EPL, UDP/IP and others as well as service-specific headers are to be interpreted as part of the user data. **In the configured state, the C13075 values of all EPL nodes must be identical.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
300	Byte	1500	**300 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13076

Parameter | Name: C13076 | 0x1F98.9 Prescaler_U16			Data type: UNSIGNED_16 Index: 11499_d = $2CEB_h$
This code configures the change rate of the SoC PS flag.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	1000	**2 Cycles**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13078

Parameter | Name: C13078 | 0x1F99 BasicEthTimeout			Data type: UNSIGNED_32 Index: 11497_d = $2CE9_h$
The code defines the time interval needed for the booting controlled node to wait for the managing node. If the controlled node detects a managing node within the interval, the controlled node changes to NMT_CS_PREOPERATIONAL_1, if not it changes to "Basic Ethernet Mode".			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	4294967295	**5000000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13079

Parameter | Name: **C13079 | 0x1F9B Multiplex Cycle CN**			Data type: UNSIGNED_8 Index: 11496_d = $2CE8_h$
This code indicates in which multiplexed cycle the node is queried. The value entered must not exceed the value in code C13074. If, for instance, the value "3" is entered in code C13074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C13079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	
Subcodes	**Lenze setting**		**Info**
C13079/1	0 cycles		
...	...		
C13079/100	0 cycles		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13102

Parameter | Name: **C13102 | 0x1F9E NMT Reset Command**	Data type: UNSIGNED_8 Index: 11473_d = $2CD1_h$
A reset command to a single EPL node in the network can result in cycle and monitoring errors. The code initiates a reset of the EPL node. When the reset has been executed, the code is automatically set to "NoCommand / NMTInvalidService".	

Selection list (Lenze setting printed in bold)	
40	ResetNode
41	ResetCommunication
42	ResetConfiguration
255	**NoCommand**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13136

Parameter | Name: **C13136 | SoC Cycle Counter**	Data type: UNSIGNED_32 Index: 11439_d = $2CAF_h$
The subcodes of the code display a counter for EPL cycles. • The SoC cycle counter can be used for activity monitoring. • The counter is started at "0" each time the EPL node is switched on. The overflow is at "4294967295".	

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13852

Parameter | Name: **C13852 | PDO words from MN**	Data type: UNSIGNED_16 Index: 10723_d = $29E3_h$
The code indicates the PDO words received (from the managing node).	

Subcodes	**Info**
C13852/1	wCtrl
C13852/2	wIn2
...	...
C13852/16	wIn16

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13853

Parameter | Name: **C13853 | PDO words to MN**	Data type: UNSIGNED_16 Index: 10722_d = $29E2_h$
The code indicates the PDO words sent (to the managing node).	

Subcodes	**Info**
C13853/1	wState
C13853/2	wOut2
...	...
C13853/16	wOut16

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13859

Parameter | Name: **C13859 | 0x1A00.0 Number of mapped Tx PDO-1**				Data type: UNSIGNED_8 Index: 10716_d = $29DC_h$
Number of sent PDOs via the PDO channel 1 (Tx)				
Display range (min. value | unit | max. value)				
0		16		
Subcodes			**Info**	
C13859/1			0x1A00.0 Number of mapped Tx PDO-1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

C13860

Parameter | Name: **C13860 | 0x160x.0 Number of mapped Rx PDO**				Data type: UNSIGNED_8 Index: 10715_d = $29DB_h$
Number of received PDOs via PDO channels 1 ... 6 (Rx)				
Display range (min. value | unit | max. value)				
0		16		
Subcodes			**Info**	
C13860/1			0x1600.0 Number of mapped Rx PDO-1	
...			...	
C13860/6			0x1605.0 Number of mapped Rx PDO-6	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

C13861

Parameter | Name: **C13861 | 0x1F8C NMT Communication Status**			Data type: UNSIGNED_8 Index: 10714_d = $29DA_h$
The code displays the current NMT state (according to the Ethernet POWERLINK specification) of the EPL node.			
Selection list (read only)		**Info**	
0	Off	NMT_GS_OFF	
25	Initialising	NMT_GS_INITIALISING	
41	ResetAppl	NMT_GS_RESET_APPLICATION	
57	ResetComm	NMT_GS_RESET_CONFIGURATION	
121	ResetConfig	NMT_GS_RESET_CONFIGURATION	
28	NotActive	NMT_CS_NOT_ACTIVE	
29	PreOp1	NMT_CS_PRE_OPERATIONAL_2	
93	PreOp2	NMT_CS_PRE_OPERATIONAL_2	
109	ReadyToOp	NMT_CS_READY_TO_OPERATE	
253	Operational	NMT_CS_OPERATIONAL	
77	Stopped	NMT_CS_STOPPED	
30	BasicEthernet	NMT_CS_BASIC_ETHERNET	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13864

Parameter | Name: C13864 | 0x1F93.1 Node ID			Data type: UNSIGNED_8 Index: 10711_d = $29D7_h$
The code displays the currently valid EPL node address (Node ID). • The node ID must be unequivocal in the EPL network segment. • See also C13899.			
Display range (min. value | unit | max. value)			
1		239	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13865

Parameter | Name: C13865 | 0x1F93.2 Node ID by HW			Data type: UNSIGNED_8 Index: 10710_d = $29D6_h$
The code indicates whether the EPL node address (node ID) has been set via rotary encoder switches at the communication module or via the software. • 0 (FALSE): Node ID has been set via software (C13899). • 1 (TRUE): node ID has been set via rotary encoder switches at the communication module. ▸ Node address setting (📖 37)			
Display range (min. value | unit | max. value)			
0		1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13879

Parameter | Name: C13879 | Bus error		Data type: UNSIGNED_32 Index: 10696_d = $29C8_h$
The code displays the bus error state which is signalled by the LED "BE" (Bus Error). • Bit 0 = 0 (0x00000000) no bus error • Bit 0 = 1 (0x00000001) bus error active		
Value is bit-coded:		
Bit 0	Bus Error if TRUE	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13880

Parameter | Name: C13880 | error response		Data type: UNSIGNED_8 Index: 10695_d = $29C7_h$
The subcodes determine the reaction when the OPERATIONAL state is left, when SoC is lost and when a CRC error occurs.		
Selection list		
0	No response	
1	Error	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
Subcodes	**Lenze setting**	**Info**
C13880/1	0: No Response	Error reaction at OPERATIONAL loss
C13880/2	0: No Response	Error reaction at SoC loss
C13880/3	0: No Response	Error reaction at CRC error
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13884

Parameter | Name: **C13884 | CU synchronisation active**		Data type: BITFIELD_8 Index: 10691_d = $29C3_h$
The code displays if the EPL node is synchronised. • Bit 0 = 0 (0x00000000) The EPL node is not synchronised. • Bit 0 = 1 (0x00000001) The EPL node is synchronised.		
Value is bit-coded:		
Bit 0	CU Synchronisation is locked if TRUE	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13885

Parameter | Name: **C13885 | Response to RPDO monitoring**		Data type: UNSIGNED_8 Index: 10690_d = $29C2_h$
The code controls the error response to failing PDO communication with a node. The failure is detected because expected PDO data are missing.		
Selection list (Lenze setting printed in bold)		
0	Maintain PDO	
1	**Delete PDO**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13887

Parameter | Name: **C13887 | Suppress emergency message upon**		Data type: BITFIELD_8 Index: 10688_d = $29C0_h$
This code serves to suppress sending alarm messages to the managing node. You can explicitly suppress a certain type of error. Furthermore, all errors are entered into the logbook. • A change can only be effective immediately if no error number with the error type selected here is active in **C00165**.		
Value is bit-coded:		
Bit 0	Error	
Bit 1	Fault	
Bit 2	Quick stop by trouble	
Bit 3	Warning Locked	
Bit 4	Warning	
Bit 5	Information	
Bit 6	Reserved	
Bit 7	Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13898

Parameter | Name: **C13898 | 0x1F9A Host Name**	Data type: VISIBLE_STRING Index: 10677_d = $29B5_h$
The code defines a DNS-compatible device name. The length is limited to 20 characters. **The device name must be non-ambiguous within the network domain.** Naming convention: The device name ... • starts with a letter; • ends with a letter or a digit. The device name consists of ... • letters (A ... Z, a ... z); • digits (0 ... 9); • hyphen (-).	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13899

Parameter | Name: **C13899 | Node ID SW**			Data type: UNSIGNED_8 Index: 10676_d = $29B4_h$
This code serves to set the node ID by means of the software. The value "0" is not a valid node ID. It serves to activate the setting via the rotary coding switches. ▸ Node address setting (📖 37)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		239	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13900

Parameter | Name: **C13900 | Firmware Type**	Data type: VISIBLE_STRING Index: 10675_d = $29B3_h$
Display of the Lenze firmware type of the communication module (product designation).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware Date**	Data type: VISIBLE_STRING Index: 10674_d = $29B2_h$
Display of the creation date of the communication module firmware	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware Version**	Data type: VISIBLE_STRING Index: 10673_d = $29B1_h$
Display of the version number of the communication module firmware	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13910

Parameter | Name: **C13910 | Last Module Error**		Data type: UNSIGNED_32 Index: 10665_d = $29A9_h$
Display of the error code of the error last occurred in the communication module		
Selection list (Lenze setting printed in bold)		
0	**No error**	
297566817	Warning locked INVALID_ADDR	
29122816	Software error	
96239921	Fault OPER_LOST	
230457649	Quick stop OPER_LOST	
297566513	Warning locked OPER_LOST	
431784241	Information OPER_LOST	
96240229	Fault CN_SoC_LOSS	
230457957	Quick stop CN_SoC_LOSS	
297566821	Warning locked CN_SoC_LOSS	
431784549	Information CN_SoC_LOSS	
96240230	Fault COM_CRC	
230457958	Quick stop COM_CRC	
297566822	Warning locked COM_CRC	
431784550	Information COM_CRC	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13915

Parameter | Name: **C13915 | CustomerObject[16]**		Data type: OCTET_STRING Index: 10660_d = $29A4_h$
The code contains an octet string[16] for customised use.		
Subcodes	**Lenze setting**	**Info**
C13915/1	00000000000000000000000000000000	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13920

Parameter | Name: **C13920 | Current address switch**			Data type: UNSIGNED_8 Index: 10655_d = $299F_h$
The code indicates the current switch position for the communication module (node ID). Not all switch positions are useful. The values for node IDs assigned by means of switches usually are between 1 and 239. ▸ Node address setting (📖 37)			
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

11.2 Table of attributes

The table of attributes contains information required for communicating with the Inverter Drive 8400 via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ No decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing is only possible when the controller is inhibited (CINH)	

Table of attributes

Code	Name	Lenze index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13000	0x1E40.2 IP Address	11575	2D37	A	1	UNSIGNED_32		☑		
C13001	0x1E40.3 Subnet Mask	11574	2D36	A	1	UNSIGNED_32		☑		
C13003	0x1E40.5 IP Address Router	11572	2D34	A	1	UNSIGNED_32		☑	☑	
C13004	0x1030.5 MAC Address	11571	2D33	A	1	OCTET_STRING		☑		
C13028	0x1F81 Node Assignment CN	11547	2D1B	A	100	UNSIGNED_32		☑	☑	
C13029	0x1F81 Node Assignment	11546	2D1A	A	3	UNSIGNED_32		☑	☑	
C13040	0x1F82 Feature Flags	11535	2D0F	E	1	UNSIGNED_32		☑		
C13060	0x1006 Cycle Time	11515	2CFB	E	1	UNSIGNED_32	1	☑	☑	
C13066	0x1F8D PResPayloadLimit RPDO CN	11509	2CF5	A	100	UNSIGNED_16	1	☑	☑	
C13067	0x1F8D PResPayloadLimit RPDO	11508	2CF4	A	3	UNSIGNED_16	1	☑	☑	
C13071	0x1F98.4 PReqPayloadLimit RPDO	11504	2CF0	E	1	UNSIGNED_16	1	☑	☑	
C13072	0x1F98.5 PResPayloadLimit TPDO	11503	2CEF	E	1	UNSIGNED_16	1	☑	☑	
C13074	0x1F98.7 Multiplex Cycle Counter	11501	2CED	E	1	UNSIGNED_8	1	☑	☑	
C13075	0x1F98.8 SDO Channel Width (Asyn-cMTU)	11500	2CEC	E	1	UNSIGNED_16	1	☑	☑	
C13076	0x1F98.9 Prescaler_U16	11499	2CEB	E	1	UNSIGNED_16	1	☑	☑	
C13078	0x1F99 BasicEthTimeout	11497	2CE9	E	1	UNSIGNED_32	1	☑	☑	
C13079	0x1F9B Multiplex Cycle CN	11496	2CE8	A	100	UNSIGNED_8	1	☑	☑	
C13102	0x1F9E NMT Reset Command	11473	2CD1	E	1	UNSIGNED_8	1	☑	☑	
C13136	SoC Cycle Counter	11439	2CAF	E	1	UNSIGNED_32	1	☑		
C13852	PDO word from MN	10723	29E3	A	16	UNSIGNED_16		☑		
C13853	PDO words to MN	10722	29E2	A	16	UNSIGNED_16		☑		
C13859	0x1A00.0 Number of mapped Tx PDO-1	10716	29DC	A	1	UNSIGNED_8	1	☑		
C13860	0x160x.0 Number of mapped Rx PDO	10715	29DB	A	6	UNSIGNED_8	1	☑		
C13861	0x1F8C NMT Communication Status	10714	29DA	E	1	UNSIGNED_8	1	☑		
C13864	0x1F93.1 Node ID	10711	29D7	E	1	UNSIGNED_8	1	☑		
C13865	0x1F93.2 Node ID by HW	10710	29D6	E	1	UNSIGNED_8	1	☑		
C13879	Bus error	10696	29C8	E	1	UNSIGNED_32		☑		
C13880	Error response	10695	29C7	A	3	UNSIGNED_8	1	☑	☑	
C13884	CU synchronisation active	10691	29C3	E	1	BITFIELD_8		☑		
C13885	Error reaction on RPDO check	10690	29C2	E	1	UNSIGNED_8	1	☑	☑	
C13887	Suppress emergency message upon	10688	29C0	E	1	BITFIELD_8		☑	☑	
C13898	0x1F9A Host Name	10677	29B5	E	1	VISIBLE_STRING		☑	☑	
C13899	Node ID SW	10676	29B4	E	1	UNSIGNED_8	1	☑	☑	
C13900	Firmware Type	10675	29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware Date	10674	29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware version	10673	29B1	E	1	VISIBLE_STRING		☑		
C13910	Last Module Error	10665	29A9	E	1	UNSIGNED_32	1	☑	☑	
C13915	CustomerObject[16]	10660	29A4	A	1	OCTET_STRING		☑	☑	
C13920	Current address switch	10655	299F	E	1	UNSIGNED_8	1	☑		

12 Index table

The following objects specified by the POWERLINK communication profile (DS 301) are supported:

EPL index	Index name	Subindex name	Type	Attr.	Reference / description	Value
0x1000	NMT_DeviceType_U32	-	U32	R	-	0
0x1001	ERR_ErrorRegister_U8	-	I8	R	-	
0x1003.0	ERR_History_ADOM	NumberOfEntries	U8	RW	-	
0x1003.1 ...254	ERR_History_ADOM	ErrorEntry_DOM	DOMAIN	R	-	
0x1006	NMT_CycleLen_U32	-	U32	RW	C13060	2000
0x1008	NMT_ManufactDevName_VS	-	VS5	R	-	Lenze
0x1009	NMT_ManufactHwVers_VS	-	VS2	R	Hardware version, variable	
0x100A	NMT_ManufactSwVers_VS	-	VS6	R	Software version, variable	
0x1016.0	NMT_ConsumerHeartbeatTime_AU32	NumberOfEntries	U8	RW	-	0 ... 4
0x1016.1 ...4	NMT_ConsumerHeartbeatTime_AU32	HeartbeatDescription	U32	RW	Value in [ms]	
0x1018.0	NMT_IdentityObject_REC	NumberOfEntries	U8	R	-	1
0x1018.1	NMT_IdentityObject_REC	VendorId_U32	U32	R	-	59
0x1030.0	NMT_InterfaceGroup_0h_REC	NumberOfEntries	U8	R	-	9
0x1030.1	NMT_InterfaceGroup_0h_REC	InterfaceIndex_U16	U16	R	Interface no. of the Ethernet interface	1
0x1030.2	NMT_InterfaceGroup_0h_REC	InterfaceDescription_VSTR	VS3	R	Description of the Ethernet interface	EPL
0x1030.3	NMT_InterfaceGroup_0h_REC	InterfaceType_U8	U8	R	Interface type	6
0x1030.4	NMT_InterfaceGroup_0h_REC	InterfaceMtu_U32	U32	R	Maximum frame size [byte]	1490
0x1030.5	NMT_InterfaceGroup_0h_REC	InterfacePhysAddress_OSTR	OS6	R	C13004/1	
0x1030.6	NMT_InterfaceGroup_0h_REC	InterfaceName_VSTR	VS16	RW	Symbolic name of the Ethernet interface	ETH0
0x1030.7	NMT_InterfaceGroup_0h_REC	InterfaceOperState_U8	U8	R	Operation status of the Ethernet interface	0: down
0x1030.8	NMT_InterfaceGroup_0h_REC	InterfaceAdminState_U8	U8	RW	Administration status of the Ethernet interface	1: up
0x1030.9	NMT_InterfaceGroup_0h_REC	Valid_BOOL	BOOL	RW	Release of the interface description	1: TRUE
0x1300	SDO_SequLayerTimeout_U32	-	U32	RW	Value in [ms]	15000
0x1301	SDO_CmdLayerTimeout_U32	-	U32	RW	Value in [ms]	30000
0x1400.0	PDO_RxCommParam_00h_REC	NumberOfEntries	U8	R	-	2
0x1400.1	PDO_RxCommParam_00h_REC	NodeID_U8	U8	RW	-	
0x1400.2	PDO_RxCommParam_00h_REC	MappingVersion_U8	U8	R	Version of PDO mapping	0
0x1401.0	PDO_RxCommParam_01h_REC	NumberOfEntries	U8	R	-	2
0x1401.1	PDO_RxCommParam_01h_REC	NodeID_U8	U8	RW	-	
0x1401.2	PDO_RxCommParam_01h_REC	MappingVersion_U8	U8	R	Version of PDO mapping	0
0x1402.0	PDO_RxCommParam_02h_REC	NumberOfEntries	U8	R	-	2
0x1402.1	PDO_RxCommParam_02h_REC	NodeID_U8	U8	RW	-	
0x1402.2	PDO_RxCommParam_02h_REC	MappingVersion_U8	U8	R	Version of PDO mapping	0
0x1403.0	PDO_RxCommParam_03h_REC	NumberOfEntries	U8	R	-	2
0x1403.1	PDO_RxCommParam_03h_REC	NodeID_U8	U8	RW	-	
0x1403.2	PDO_RxCommParam_03h_REC	MappingVersion_U8	U8	R	Version of PDO mapping	0
0x1404.0	PDO_RxCommParam_04h_REC	NumberOfEntries	U8	R	-	2
0x1404.1	PDO_RxCommParam_04h_REC	NodeID_U8	U8	RW	-	
0x1404.2	PDO_RxCommParam_04h_REC	MappingVersion_U8	U8	R	Version of PDO mapping	0
0x1405.0	PDO_RxCommParam_05h_REC	NumberOfEntries	U8	R	-	2
0x1405.1	PDO_RxCommParam_05h_REC	NodeID_U8	U8	RW	-	
0x1405.2	PDO_RxCommParam_05h_REC	MappingVersion_U8	U8	R	Version of PDO mapping	0

EPL index	Index name	Subindex name	Type	Attr.	Reference / description	Value
0x1600.0	PDO_RxMappParam_00h_REC	NumberOfEntries	U8	RW	C13860/1	
0x1600.1 ...16	PDO_RxMappParam_00h_REC	ObjectMapping_U64	U64	RW	-	
0x1601.0	PDO_RxMappParam_01h_REC	NumberOfEntries	U8	RW	C13860/2	
0x1601.1 ...16	PDO_RxMappParam_01h_REC	ObjectMapping_U64	U64	RW	-	
0x1602.0	PDO_RxMappParam_02h_REC	NumberOfEntries	U8	RW	C13860/3	
0x1602.1 ...16	PDO_RxMappParam_02h_REC	ObjectMapping_U64	U64	RW	-	
0x1603.0	PDO_RxMappParam_03h_REC	NumberOfEntries	U8	RW	C13860/4	
0x1603.1 ...16	PDO_RxMappParam_03h_REC	ObjectMapping_U64	U64	RW	-	
0x1604.0	PDO_RxMappParam_04h_REC	NumberOfEntries	U8	RW	C13860/5	
0x1604.1 ...16	PDO_RxMappParam_04h_REC	ObjectMapping_U64	U64	RW	-	
0x1605.0	PDO_RxMappParam_05h_REC	NumberOfEntries	U8	RW	C13860/6	
0x1605.1 ...16	PDO_RxMappParam_05h_REC	ObjectMapping_U64	U64	RW	-	
0x1800.0	PDO_TxCommParam_00h_REC	NumberOfEntries	U8	R	-	2
0x1800.1	PDO_TxCommParam_00h_REC	NodeID_U8	U8	R	Node ID of the receiver	0 (pseudo node ID for PRes-Tx)
0x1800.2	PDO_TxCommParam_00h_REC	MappingVersion_U8	U8	R	Version of PDO mapping	0
0x1A00.0	PDO_TxMappParam_00h_REC	NumberOfEntries	U8	RW	C13859/1	
0x1A00.1 ...16	PDO_TxMappParam_00h_REC	ObjectMapping_U64	U64	RW	-	
0x1C0A.0	DLL_CNCollision_REC	NumberOfEntries	U8	R	-	1
0x1C0A.1	DLL_CNCollision_REC	CumulativeCnt_U32	U32	R	-	
0x1C0B.0	DLL_CNLossSoC_REC	NumberOfEntries	U8	R	-	3
0x1C0B.1	DLL_CNLossSoC_REC	CumulativeCnt_U32	U32	R	-	
0x1C0B.2	DLL_CNLossSoC_REC	ThresholdCnt_U32	U32	R	-	
0x1C0B.3	DLL_CNLossSoC_REC	Threshold_U32	U32	RW	-	
0x1C0F.0	DLL_CNCRCError_REC	NumberOfEntries	U8	R	-	1
0x1C0F.1	DLL_CNCRCError_REC	CumulativeCnt_U32	U32	R	-	
0x1C0F.2	DLL_CNCRCError_REC	ThresholdCnt_U32	U32	R	-	
0x1C0F.3	DLL_CNCRCError_REC	Threshold_U32	U32	RW	-	
0x1C10	DLL_CNLossOfLinkCum_U32	-	U32	R	-	0
0x1E40.0	NWL_IpAddrTable_0h_REC	NumberOfEntries	U8	R	-	5
0x1E40.1	NWL_IpAddrTable_0h_REC	IfIndex_U16	U16	R	-	0
0x1E40.2	NWL_IpAddrTable_0h_REC	Addr_IPAD	U32	R	C13000/1	
0x1E40.3	NWL_IpAddrTable_0h_REC	NetMask_IPAD	U32	R	C13001/1	
0x1E40.4	NWL_IpAddrTable_0h_REC	ReasmMaxSize_U16	U16	R	Maximum frame size which can be reconstructed from frames arriving in fragments [byte].	1490
0x1E40.5	NWL_IpAddrTable_0h_REC	DefaultGateway_IPAD	U32	RW	C13003/1	
0x1E4A.0	NWL_IpGroup_REC	NumberOfEntries	U8	R	-	3
0x1E4A.1	NWL_IpGroup_REC	Forwarding_BOOL	BOOL	R	Release for IP router function	0: FALSE
0x1E4A.2	NWL_IpGroup_REC	DefaultTTL_U16	U16	RW	TimeToLive value for transmitted IP frames Limits the range with regard to router stations passed.	64
0x1E4A.3	NWL_IpGroup_REC	ForwardingDatagrams_ U32	U32	R	Counter for routed frames	-
0x1F81.0	NMT_NodeAssignment_AU32	NumberOfEntries	U8	R	-	254
0x1F81.1 ...100	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	C13028/1...100	
0x1F81.240	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	C13029/1	
0x1F81.253	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	C13029/2	

EPL index	Index name	Subindex name	Type	Attr.	Reference / description	Value
0x1F81.254	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	C13029/3	
0x1F82	NMT_FeatureFlags_U32	-	U32	R	C13040	0x20
0x1F83	NMT_EPLVers_U8	-	U8	R	EPL version	
0x1F8C	NMT_CurrState_U8	-	U8	R	C13861	
0x1F8D.0	NMT_MNPResPayloadList_AU32	NumberOfEntries	U8	R	-	254
0x1F8D.1 ...100	NMT_MNPResPayloadList_AU32	PResPayload	U16	RW	C13066/1...100	
0x1F8D.240	NMT_MNPResPayloadList_AU32	PResPayload	U16	RW	C13067/1	
0x1F8D.253	NMT_MNPResPayloadList_AU32	PResPayload	U16	RW	C13067/2	
0x1F8D.254	NMT_MNPResPayloadList_AU32	PResPayload	U16	RW	C13067/3	
0x1F93.1	NMT_EPLNodeID_REC	NodeID_U8	U8	R	C13864	
0x1F93.2	NMT_EPLNodeID_REC	NodeIDByHW_BOOL	BOOL	R	-	
0x1F98.0	NMT_CycleTiming_REC	NumberOfEntries	U8	R	-	9
0x1F98.1	NMT_CycleTiming_REC	IsochrTxMaxPayload_U16	U16	R	Size of the isochronous transmit memory	1490
0x1F98.2	NMT_CycleTiming_REC	IsochrRxMaxPayload_U16	U16	R	Size of the isochronous receive memory	1490
0x1F98.3	NMT_CycleTiming_REC	PResMaxLatency_U32	U32	R	Isochronous response delay [ns]	2000
0x1F98.4	NMT_CycleTiming_REC	PReqActPayload_U16	U16	RW	C13071	
0x1F98.5	NMT_CycleTiming_REC	PResActPayload_U16	U16	RW	C13072	
0x1F98.6	NMT_CycleTiming_REC	ASndMaxLatency_U32	U32	R	Asynchronous response delay [ns]	80000
0x1F98.7	NMT_CycleTiming_REC	MultipleCycleCnt_U8	U8	RW	C13074	0
0x1F98.8	NMT_CycleTiming_REC	AsyncMTUSize_U16	U16	RW	C13075	
0x1F98.9	NMT_CycleTiming_REC	Prescaler_U16	U16	RW	C13076	
0x1F99	NMT_CNBasicEthernetTimeout_U32	-	U32	RW	C13078	
0x1F9A	NMT_HostName_VSTR	-	VS32	RW	C13898	
0x1F9B.1 ...100	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	C13079	0 ... value of 0x1F98.7
0x1F9E	NMT_ResetCmd_U8	-	U8	W	C13102	

13 Index

0-9

A

B

C

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Drives GmbH
Postfach 10 13 52, D-31763 Hameln
Breslauer Straße 3, D-32699 Extertal
Germany
HR Lemgo B 6478
+49 5154 82-0
+49 5154 82-2800
sales.de@lenze.com
www.lenze.com

Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service.de@lenze.com

Accessories

EtherNet/IP™

E84AYCEO

Inverter Drives 8400 ________________________

Communication Manual EN

13443009

Lenze

Contents

1 About this documentation

Contents

This documentation only contains descriptions of the E84AYCEO (EtherNet/IP™) communication module.

> **Note!**
>
> This documentation supplements the **mounting instructions** supplied with the communication module and the **hardware manual "Inverter Drives 8400"**.
>
> **The hardware manual contains safety instructions that must be observed!**

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe the software of another manufacturer. No guarantee can be given for corresponding information in this documentation. Information on the use of the software can be found in the documents for the host (PLC, scanner).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

> **Tip!**
>
> Detailed information on EtherNet/IP can be found on the website of the user organisation ODVA (Open DeviceNet Vendor Association):
>
> www.odva.org

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.

Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.Lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
Communication module EtherNet/IP	E84AYCEO	VA	01.01

From software version 01.02, the "AC Drive Profile" of the Inverter Drives 8400 is supported from version V13.00.

Screenshots/application examples

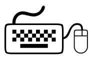
All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and software version of the installed engineering tools (»Engineer«, »RSLogix 5000«), the screenshots in this documentation may differ from the screen representation.

1.1 Document history

Version			Description
1.0	08/2012	TD17	First edition
2.0	12/2012	TD17	• Revision for software version 01.02 • New layout
3.0	10/2013	TD17	Revised chapters: ▸ I/O data transfer (implicit messages) (📖 56) ▸ Parameter data transfer (explicit messages) (📖 84) ▸ Implemented CIP™ objects (📖 124)

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Blue text colour	All information that applies to from a certain software version of the drive onwards are marked accordingly in this documentation. Example: This function extension is available from software version V3.0 onwards!
Program name	» «	The Lenze PC software »Engineer«...
Window	*italics*	The *Message window*... / The dialog box *Options*...
Variable names		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button... / the **Copy** command... / the **Characteristics** tab... / the **Name** input field...
Sequence of menu commands		If several commands are required to execute one function, the single commands are separated by an arrow: Select the **File** → **Open** command to...
Hyperlink	underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 9)	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
ACD	Address Conflict Detection
Adapter	EtherNet/IP slave
Inverter / Drive	Lenze inverters of the "Inverter Drives 8400" series
Standard device	
ARP	Address Resolution Protocol
Use BOOTP	Bootstrap Protocol
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3"). In everyday language, the term is also referred to as "subindex".
»Engineer«	Software from Lenze which supports you throughout the whole machine life cycle - from planning to maintenance.
EtherNet/IP	EtherNet/IP™ (EtherNet Industrial Protocol) is a fieldbus system based on Ethernet which uses the Common Industrial Protocol™ (CIP™) for data exchange. EtherNet/IP™ and Common Industrial Protocol™ (CIP™) are trademarks and patented technologies, licensed by the user organisation ODVA (Open DeviceNet Vendor Association), USA.
Use DHCP	Dynamic Host Configuration Protocol
DSCP	Differentiated Services Codepoints
EDS	Electronic Data Sheet
Explicit messages	Explicit Messages are used to transfer parameter data.
HW	Hardware
IGMP	Internet Group Management Protocol
Implicit messages	Implicit Messages are used to transfer I/O data.
"Class 1" connection	I/O connection
"Class 3" connection	Explicit connection
Level 2	EtherNet/IP performance level 2: I/O Message Server including Explicit Message Server
Lenze setting	Default settings of the device, preconfigured ex works.
Basic setting	
PLC	Programmable Logic Controller
QoS	Quality of Service
RPI	Requested Package Interval: Interval requested between 2 telegrams for cyclic data transmission
»RSLogix 5000«	Programming and development software from Rockwell for hosts (scanners) in EtherNet/IP networks (e.g. Allen-Bradley Logix controllers).
Scanner	EtherNet/IP master or client
Host	
SW	Software
TTL	Time To Live: Validity time of data packets in the EtherNet/IP network
UCMM	Unconnected Message Manager

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Stop!	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

Lenze drive and automation components ...

- must only be used as directed.
- must never be commissioned if they display signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

The following applies to Lenze drive components ...

- only use the accessories approved.
- Only use original manufacturer spare parts.

Observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
  ▸ Product features (📖 14)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...

- are familiar with installing, mounting, commissioning, and operating the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables corresponding to the given specifications.
  ▸ Ethernet cable specification (📖 31)

Documentation for the standard device, host, system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If the Inverter Drives 8400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not ensured without implementing external measures.
▸ Protective insulation (📖 17)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 23)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

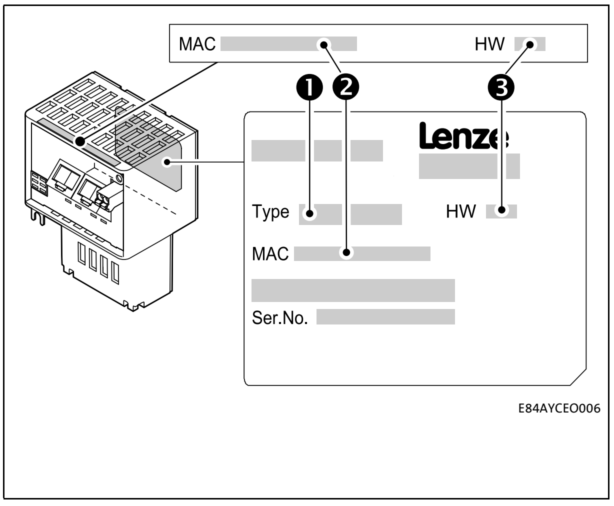
Product series	Type designation	From software version
Inverter Drives 8400 StateLine	E84AVSCxxxxx	11.00
Inverter Drives 8400 HighLine	E84AVHCxxxxx	11.00
Inverter Drives 8400 TopLine	E84AVTCxxxxx	02.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in EtherNet/IP networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation as well as the hardware and software version of the communication module are indicated on the nameplate:



[3-1] Identification data

1 Type designation (type)

E84	Product series
A	Version
Y	Module identification: Extension module
C	Module type: Communication module
EO	EtherNet/IP
V/S	V: Coated version
	S: Standard version

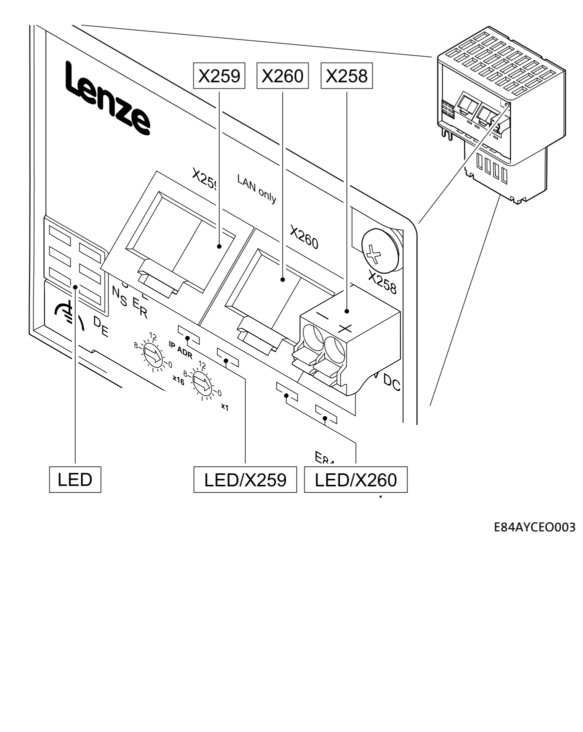
2 MAC address (MAC)
00-0A-86-xx-yy-zz:
Hardware address of the communication module for unique identification in the network

3 Hardware version (HW)

3.3 Product features

- Interface module for the EtherNet/IP communication system, for attachment to the expansion slots of Inverter Drives 8400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- The Inverter Drive 8400 is always an adapter device:
  EtherNet/IP adapter with "Level 2" functionality
- 2-port interface with integrated switch functionality
- Access to all Lenze parameters (configurable via TCP/IP using the Lenze »Engineer«)
- Up to 3 TCP/IP socket connections for communication with the Lenze »Engineer«
- Support of "IP Config Pending" (activation of changed IP configuration by "power off/on" or "type 0 reset")
- Support of the redundancy protocol DLR (Device Level Ring) as "beacon-based ring node"
- Up to 16 I/O data words (32 bytes) are possible.
- Further CIP features:
  - Max. 8 CIP connections
  - 1 "exclusive owner" connection
  - I/O connection type: cyclic
  - Minimum I/O cycle time: 4 ms
  - Support of multicast messages, UCMM, ACD, BOOTP/DHCP, VLAN tagging/DSCP

3.4 Connections and interfaces



X258	External voltage supply of the communication module • 2-pin plug connector with spring connection ▶ External voltage supply (📖 33)
X259 **X260**	EtherNet/IP terminals • RJ45-sockets • each with 2 LED status displays for diagnostics ▶ Network topology (📖 27) ▶ EtherNet/IP connection (📖 29) ▶ Status indicators at the RJ45 sockets (X259, X260) (📖 94)
x1 **x16**	Rotary coding switches for node address setting (IP ADR) ▶ Setting node addresses by means of rotary coding switches (📖 40)
MS **ST** **NS** **ER** **DE**	5 LED status displays for diagnostics ▶ Module status displays (📖 90) ▶ CIP™ status displays (📖 91)

[3-2] E84AYCEO communication module (EtherNet/IP)

4 Technical data

4.1 General data and operating conditions of the EtherNet/IP

Range	Values
Order designation	E84AYCEO
Communication profile	EtherNet/IP
Communication medium	S/FTP (Screened Foiled Twisted Pair), ISO/IEC 11801 or EN 50173, CAT 5e
Interface for communication	RJ45 Standard Ethernet (according to IEEE 802.3), 100Base-TX (Fast Ethernet)
Network topology	Tree, star, and line
Type of node	Adapter (slave)
Number of nodes	Max. 254 in the subnetwork
Max. cable length	100 m
Vendor ID	587 (0x24B), Lenze ('Lenze AC Tech' in older Rockwell data)
Device type	2 (0x02), AC Drive
Product code	8400 (0x20D0)
Baud rate	• 10 Mbps • 100 Mbps
Transmission mode	Half duplex / full duplex
Switching method	Store-and-forward / cut-through
Switch latency	Approx. 125 µs at max. telegram length
Voltage supply	External supply via separate power supply unit • + : U = 24 V DC (20.4 ... 28.8 V), I_{max} = 140 mA • - : Reference potential for external voltage supply
Conformities , approvals	• CE • UL (see also hardware manual)

Hardware manual for Inverter Drives 8400

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 Protective insulation

Danger!

Dangerous voltage

If the Inverter Drives 8400 are operated on a phase-earthed mains with a rated mains voltage of ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injuries

Protective measures:

If protection against accidental contact is required for the control terminals of the inverter and the connections of the plugged device modules, ...

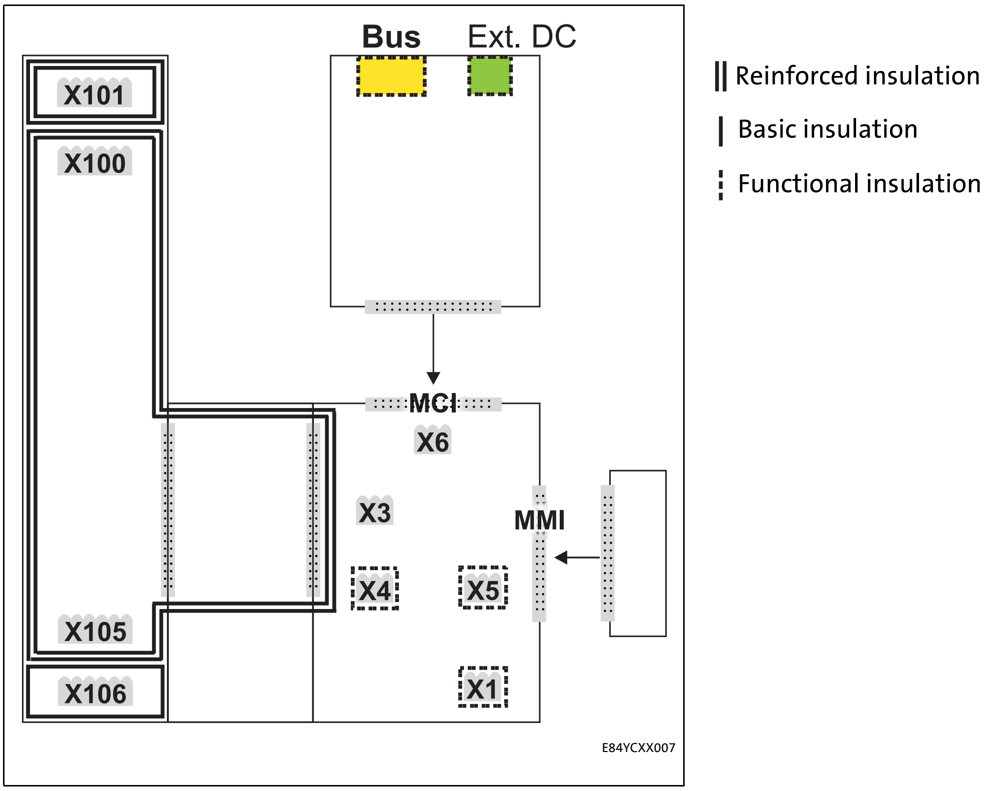
- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

Note!

The existing protective insulation in the Inverter Drives 8400 is implemented according to EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Inverter Drive 8400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection
X100	Mains / DC bus connection
X101	Relay contact
X105	Motor/brake resistor
X106	Motor PTC
X1	System bus (CANopen)
X3	Analog inputs/outputs
X4	Digital outputs
X5	Digital inputs
X6	Diagnostics
MCI	Slot for communication module
MMI	Slot for the memory module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MCI and the X100 mains terminal?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.3 Protocol data

Range	Values
I/O data words	1 ... 16 words (max. 32 bytes, 16 bits/word)
Supported CIP services	• Get_Attributes_All • Get_Attribute_Single • Set_Attribute_Single • Reset (types '0' and '1' only) • Forward_Open • Forward_Close • Get_Member

4.4 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in an EtherNet/IP network depend on the ...

- processing time in the inverter;
- telegram runtime (baud rate / telegram length);
- nesting depth of the network.

processing time within the inverter

Data	Processing time	
Process data (I/O data)	10 ms	Lenze standard update cycle (can be changed in the Rockwell engineering tool)
	+ 0 ... 1 ms	Processing time in the module
	+ 1 ... x ms	Runtime of the application task of the technology application used (tolerance)
Parameter data	Approx. 30 ms + a tolerance of 20 ms (typically) For some codes, the processing time may be longer (see software manual/ »Engineer« online help for Inverter Drives 8400).	

There are no interdependencies between parameter data and I/O data.

4.5 Internal switch latency

The integrated 2-port switch causes runtime delays. For "store-and-forward" and 100 Mbps, these runtime delays can be calculated as follows.

Runtime delay for an output data packet of the scanners incl. 32-bit "run/idle header" with 16-bit sequence counter:

Runtime delay = ((66 permanent bytes + I/O data in bytes) x 8 x 10 nsec) + 4 µsec

Runtime delay for an output data packet of an adapter without 32/bit "run/idle header":

Runtime delay = ((62 permanent bytes + I/O data in bytes) x 8 x 10 nsec) + 4 µsec

Example

Delay of an output data packet of the scanners with 8 output data words (16 bytes):

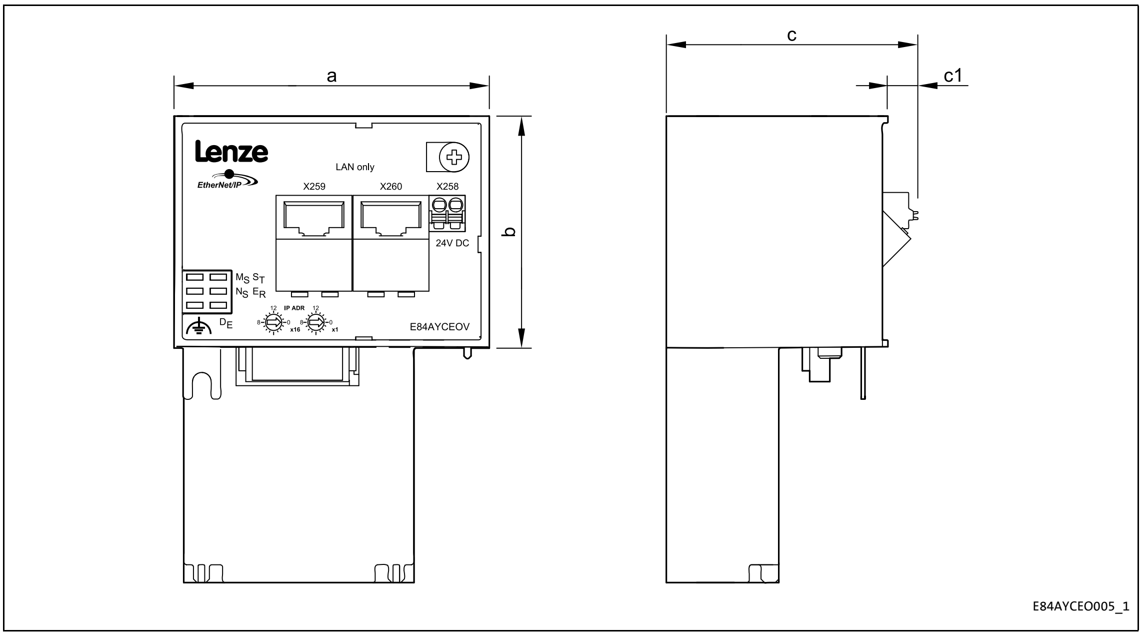
- ((66 permanent bytes + 16 bytes) x 8 x 10 nsec) + 4 µsec
- (82 bytes x 8 x 10 nsec) + 4 µsec
- 6.56 µsec + 4 µsec = **10.56 µsec**

Note!

The use of external switches can also lead to runtime delays. Depending on the system constellation, it may be useful to create a star topology or a line/mix topology.

▶ Network topology (📖 27)

4.6 Dimensions



[4-2] Dimensions

Type	Dimensions [mm]			
	a	b	c	c1
E84AYCEO	67	50	57	8

5 Installation

Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Fieldbus communication is not possible or faulty.

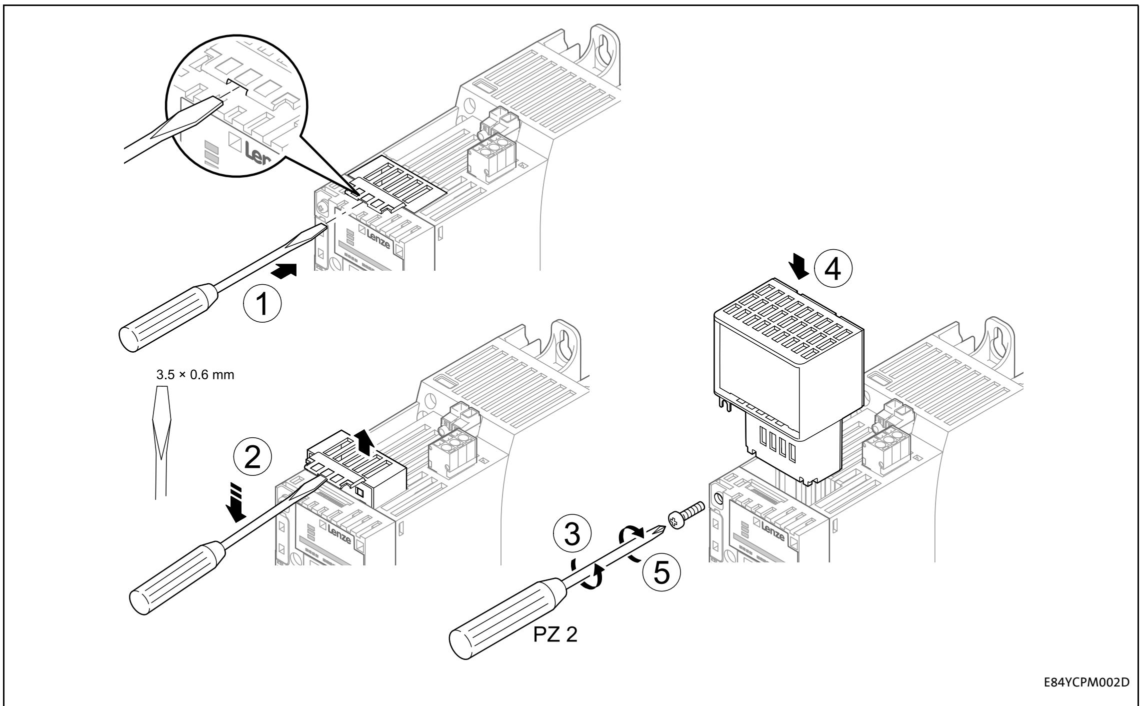
Protective measures

Before touching the module, be sure that you are free of electrostatic charge.

5.1 Mechanical installation

The communication module can be plugged in or unplugged from the MCI slot when the drive is switched on. When the module is plugged in, it is detected automatically, and a function and version plausibility check is executed.

5.1.1 Mounting for standard devices of 0.25 kW and 0.37 kW

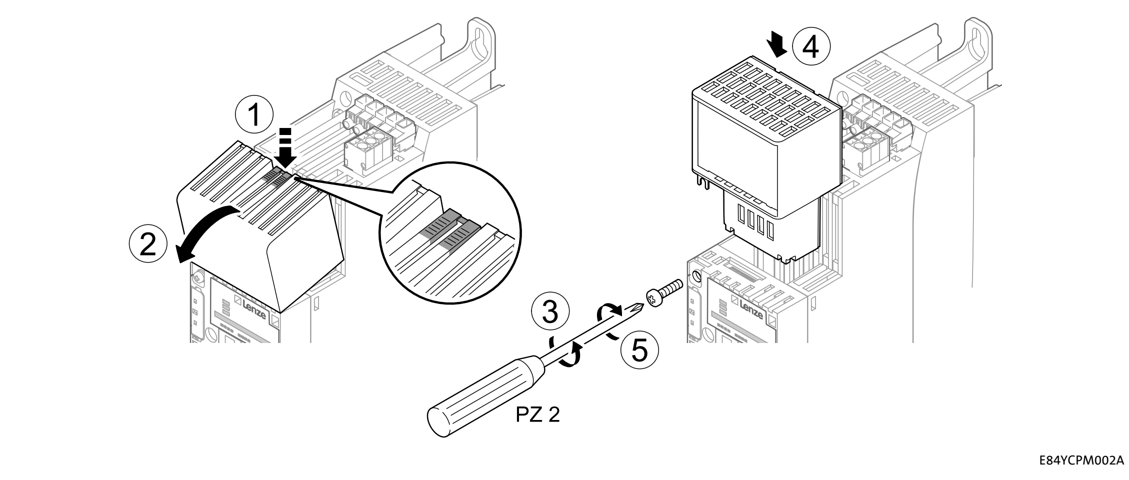


[5-1] Mounting for standard devices of 0.25 kW and 0.37 kW

Mounting steps

1. Use a screwdriver to lever out the cover of the MCI slot of the standard device and remove it (1, 2).
2. Loosen the securing screw for the communication module at the standard device (3).
3. Insert the communication module into the MCI slot of the standard device (4).
4. Tighten the securing screw again (5).

5.1.2 Mounting for standard devices of 0.55 kW or more

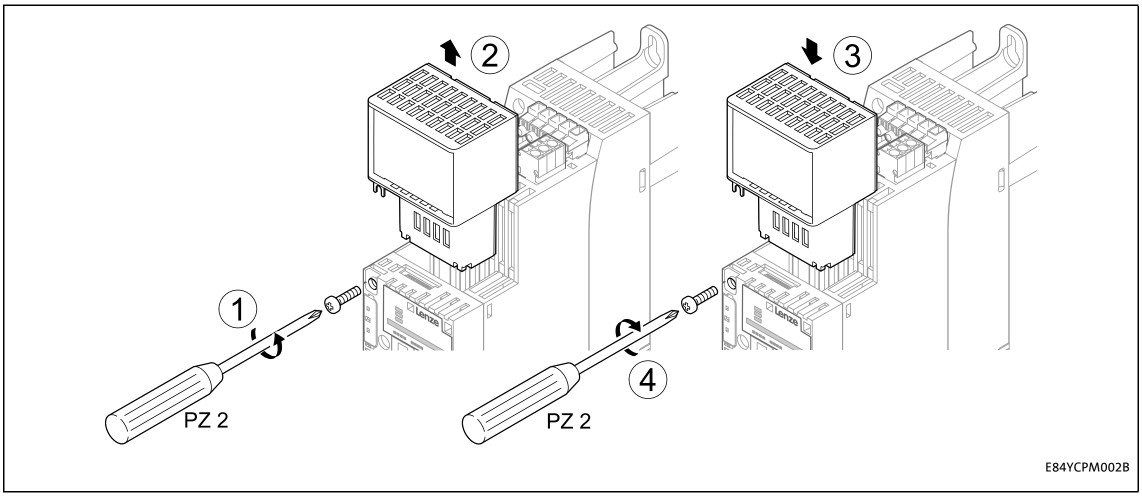


[5-2] Mounting for standard devices of 0.55 kW or more

Mounting steps

1. Slightly impress the pressure surface of the top side of the MCI slot cover of the standard device (1).
2. Tilt the cover forward and remove it from the standard device (2).
3. Loosen the securing screw for the communication module at the standard device (3).
4. Insert the communication module into the MCI slot of the standard device (4).
5. Tighten the securing screw again (5).

5.1.3 Replacing the communication module



[5-3] Replacing the communication module

Mounting steps

1. Loosen the securing screw for the communication module at the standard device (1).
2. Pull the communication module out of the MCI slot of the standard device (2).
3. Insert the new communication module into the MCI slot of the standard device (3).
4. Tighten the securing screw again (4).

5.2 Electrical installation

Documentation for the standard device, host, system/machine

Observe the notes and wiring instructions contained in this documentation.

5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. Remove the plastic sheath of the cable at a length of 2 cm.
2. Fasten the cable shield to the shield support of the standard device.

5.2.2 Network topology

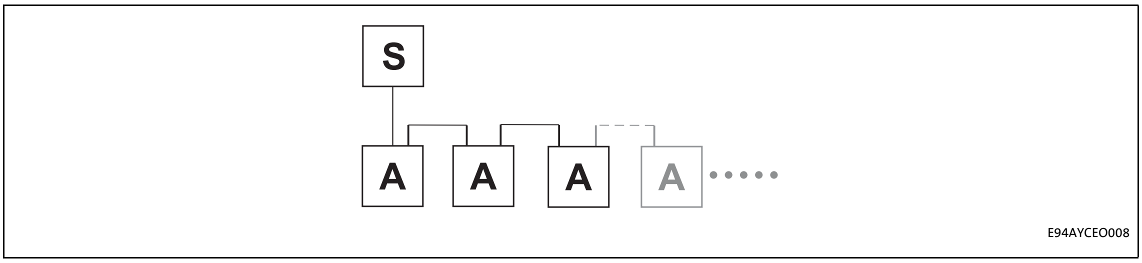
It is typical of EtherNet/IP to have a rather free topology the limiting factor of which is large message latencies due to e.g. switches connected in series.

▶ Internal switch latency (📖 21)

The combination of a line and a stub is useful for system wiring.

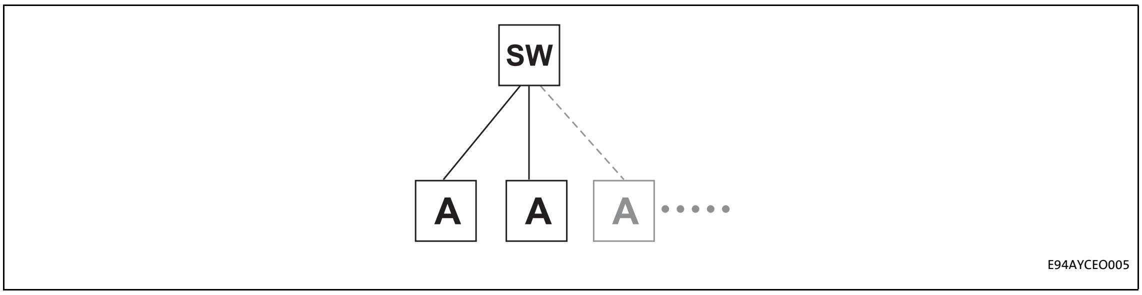
EtherNet/IP supports the following topologies:

- Line



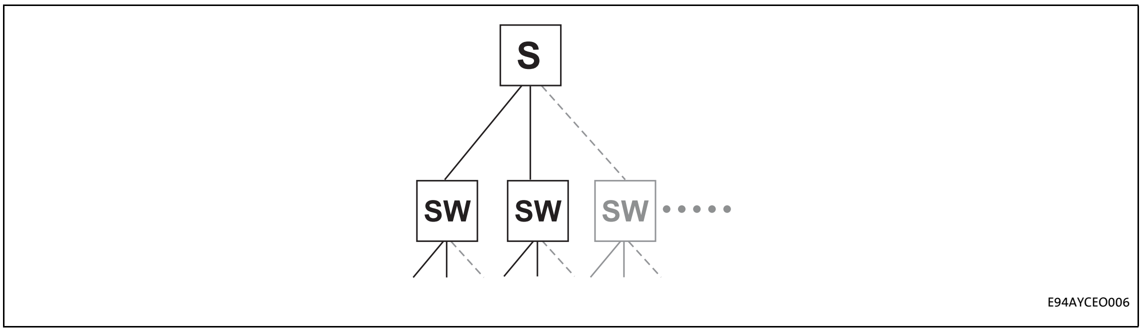
[5-4] Line topology (S = scanner, A = adapter)

- Switch / star



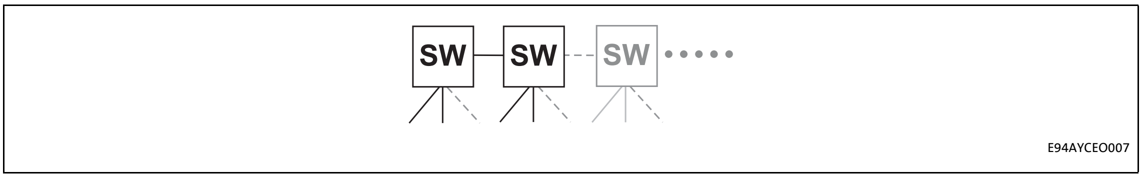
[5-5] Switch / star topology (SW = switch, A = adapter)

- Tree via switches



[5-6] Tree topology (S = scanner, SW = switch)

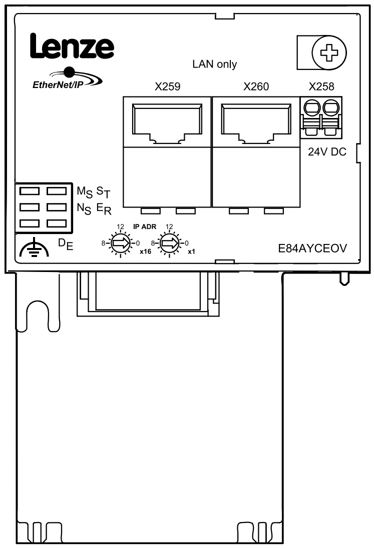
- Switch / switch



[5-7] Switch / switch topology (SW = switch)

5.2.3 EtherNet/IP connection

A connection to the EtherNet/IP network is established via RJ45 sockets **X259** and **X260**.



[5-8] EtherNet/IP terminals X259 and X260

For connection of the communication module to the EtherNet/IP fieldbus, a standard Ethernet patch cable is suitable.

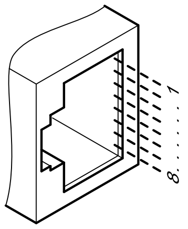
▶ Ethernet cable specification (📖 31)

The installation and removal of the Ethernet cables is optimised for the use of connectors in accordance with the "Automation Initiative of German Domestic Automobile Manufacturers" (AIDA).

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment of the RJ45 sockets

RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Note!

Dependent on the configuration of the Ethernet port of the device to be connected, we recommend the use of a cross-over cable.

Tip!

The EtherNet/IP interfaces feature an auto-MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite EtherNet/IP interface and irrespective of the type of cable used (standard patch cable or crossover cable).

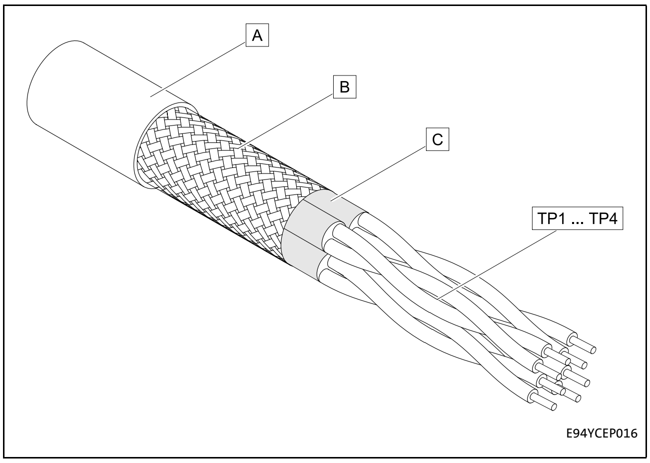
5.2.4 Ethernet cable specification

Note!

Only use cables that meet the listed specifications.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair), ISO/IEC 11801 or EN 50173, CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 ... TP4 Twisted core pairs 1 ... 4
▶ Colour code of the Ethernet cable (📖 32)

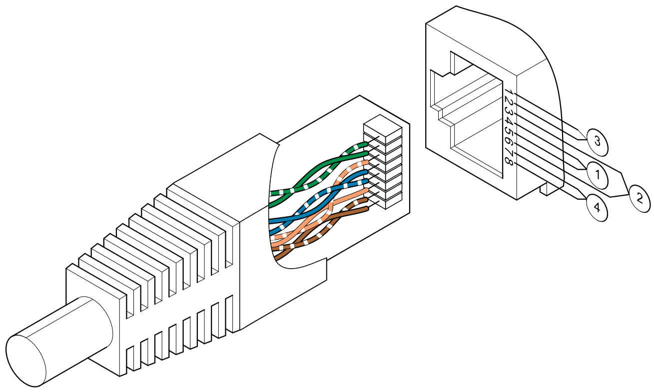
[5-9] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-10] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	white / green	white / orange
	2	Tx -	green	orange
2	3	Rx +	white / orange	white / green
1	4		blue	blue
	5		white / blue	blue / white
2	6	Rx -	orange	green
4	7		white / brown	white / brown
	8		brown	brown

5.2.5 External voltage supply

The communication module can be externally supplied with voltage via separate supply cables at the 2-pin plug connector **X258**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

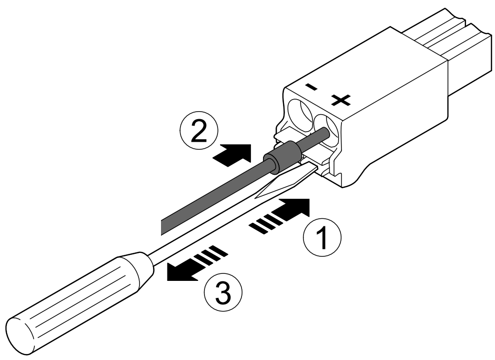
The external voltage supply of the communication module is necessary if the power supply for the basic device fails but communication via the fieldbus is to be maintained.

The parameters of a standard device separated from the mains cannot be accessed.

Wiring of the X258 plug connector

Stop!

Only wire the plug connector if the standard device is disconnected from the mains.



[5-11] Wiring of the 2-pin plug connector with spring connection

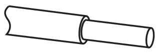
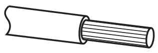
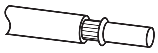
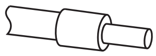
How to wire the plug connector with spring connection:

1. Press a screwdriver into the notch below the terminal and keep it pressed.
2. Place the supply cable in the terminal.
3. Remove the screwdriver from the notch.

Assignment of the X258 plug connector

Name	Description
+	U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %) I = 140 mA
-	Reference potential for the external voltage supply

Terminal data

Range	Values	
Electrical connection	2-pin plug connector with spring connection	
Possible connections	Rigid:	
		0.2 ... 1.5 mm^2 (AWG 24 ... 16)
	Flexible:	
		Without wire end ferrule 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, without plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, with plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
Stripping length	10 mm	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The codes of the inverter and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

▶ Parameter reference (🕮 106)

6.1 Before initial switch-on

Stop!

Before you switch on the Inverter Drive 8400 with the communication module for the first time, check all the wiring for completeness, short-circuits and earth faults.

6.2 Configuring the host system (scanner)

To be able to communicate with the communication module, the host (scanner) must be configured first.

The configuration of EtherNet/IP networks always requires an EtherNet/IP configuration software (e.g. »RSLogix 5000« from Rockwell) for the host system (scanner).

The configuration software is necessary for the programming of controller programs, EtherNet/IP configuration, real-time execution and diagnostics.

The basic parameters of the communication module are stored in the internal configuration memory and can be used for node detection by the scanner.

For node detection (fieldbus scan), the corresponding device descriptions of the Lenze device family are used.

Tip!

Here you will find information on configuring with the »RSLogix 5000« programming software from Rockwell:

- I/O configuration with »RSLogix 5000« version 19 or lower (68)
- I/O configuration with »RSLogix 5000« version 20 or higher (73)

6.2.1 EDS files

Depending on the EtherNet/IP scanner configuration software, the EDS files (Electronic Data Sheet) may be used for the configuration of the network profile, the communication with the participating devices and the automatic generation of tags. For this purpose, the EDS files have to be imported into the controller project of the EtherNet/IP configuration software.

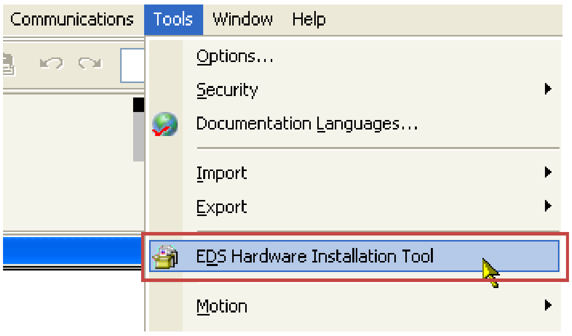
The EDS file required for the configuration can be found in the download area at:

www.Lenze.com

Tip!

From version 20 onwards, Rockwell's »RSLogix 5000« programming software features an "EDS Hardware Installation Tool" that can be used to ...

- install/import EDS files;
- create EDS files;
- carry out EDS uploads;
- delete EDS files from your controller project.



In »RSLogix 5000«, the dialog for the "EDS Hardware Installation Tool" is self-explanatory and not described further in this documentation.

6.2.2 Example: IP configuration of the Allen-Bradley 1769-L32E CompactLogix controller

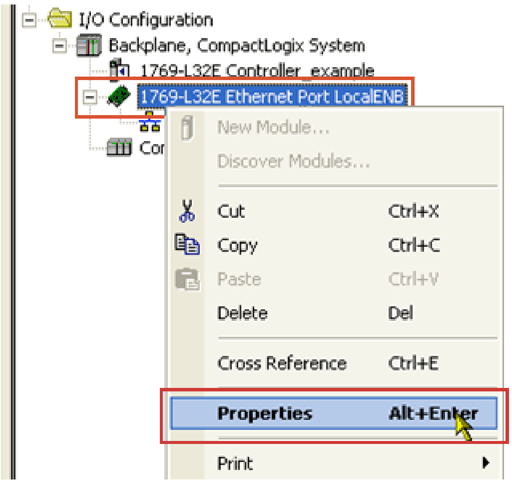
In this example, the Allen-Bradley CompactLogix control unit 1769-L32E with integrated EtherNet/IP interface is used for communication with the Inverter Drives 8400.

The »RSLogix 5000« programming software from Rockwell is used for the configuration.

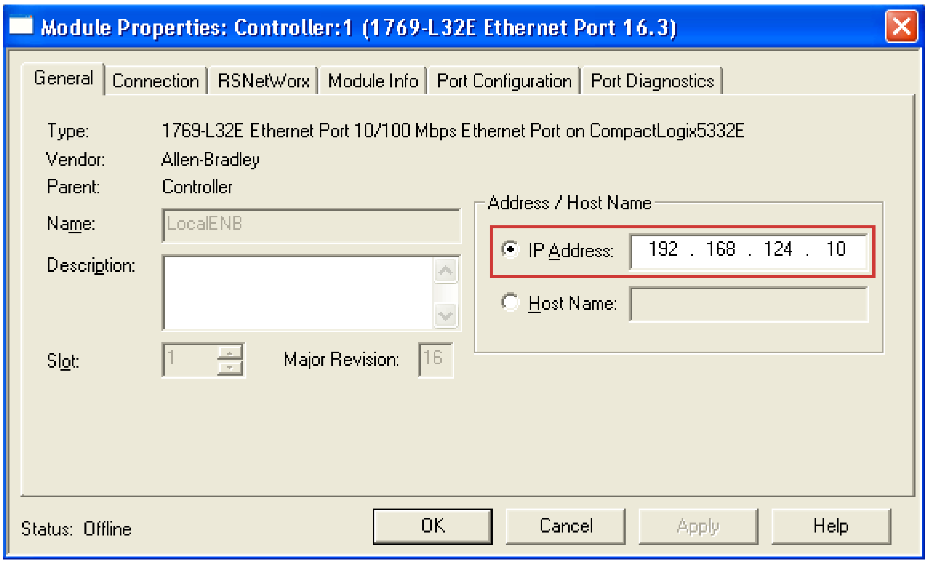
To establish communication via an EtherNet/IP network, add the controller and its scanner to the I/O configuration.

How to set the IP configuration of the 1769-L32E CompactLogix controller using the »RSLogix 5000« programming software:

1. Click on the **I/O Configuration** folder in the configuration tree.
2. Right click on "1769-L32E Ethernet Port LocalENB" and select "Properties" from the context menu.

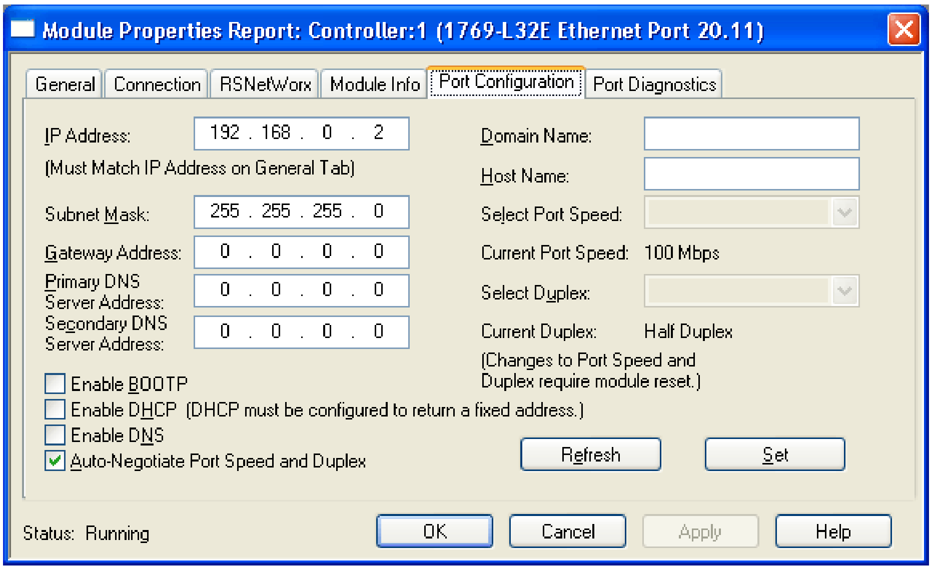


3. Go to the **General** tab of the "Module Properties: ..." dialog window and enter the IP address of the scanner.



4. Then click **OK**.

5. Go to the **Port Configuration** tab and enter the IP configuration, BOOTP setting, Ethernet baud rate and duplex mode.



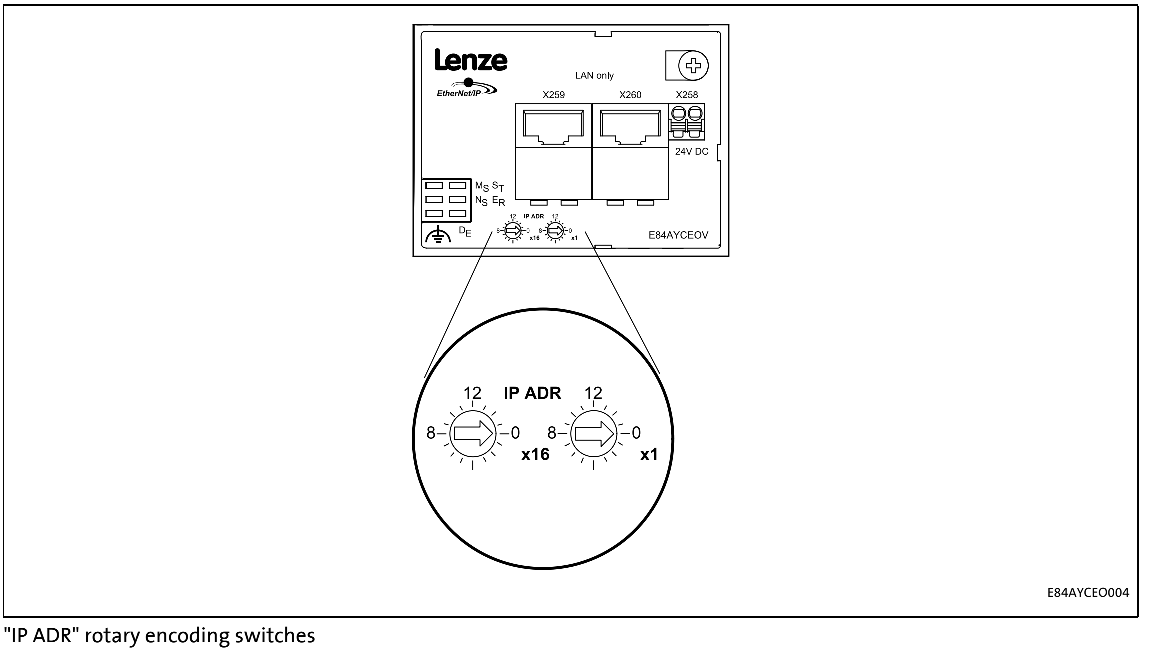
6. Then click **OK**.
   - Now, the scanner is configured for the EtherNet/IP network.
   - Here you will find information on project planning with the »RSLogix 5000« programming software from Rockwell:
     - I/O configuration with »RSLogix 5000« version 19 or lower (📖 68)
     - I/O configuration with »RSLogix 5000« version 20 or higher (📖 73)

6.3 Setting node addresses by means of rotary coding switches

Note!

- Each network node address must only be used once.
- Adjustable address range: 1 ... 254

You can set the node address (IP address) by means of the two "IP ADR" rotary coding switches located on the front.



[6-1] "IP ADR" rotary encoding switches

The inscription on the rotary coding switches corresponds to the valencies for determination of the IP address.

The associated IP address of the communication module is the result of address setting by means of the rotary encoder switches, e.g. IP address 192.168.100.**[IP ADR]**

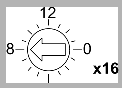
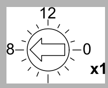
In the default state, the rotary coding switches have been set to "0".

The rotary coding switch is re-evaluated when the following events occur:

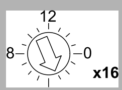
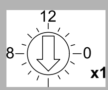
- Type 0 reset service to the Identity Object (1 / 0x01) (🕮 126);
- Type 1 reset service to the Identity Object (1 / 0x01) (🕮 126);
- Switch the voltage supply for the communication module off and then on again.

The current value of the rotary coding switch is indicted in code C13920.

Possible settings of the rotary coding switches

Position of the rotary coding switches				Description
				
0	× 16	0	× 1	Default status: • Value from code C13005 (IP configuration reference, 1: BOOTP) • The setting can also be made by means of write-access to attribute 3 (Configuration Control) of instance 1 of the TCP/IP Interface Object (245 / 0xF5) (🕮 141).
0 ... 15		1 ... 14		IP address 1 ... 254 • The static address stored in subcode C13000/1...3 is used. • The subcode C13000/4 is replaced with the value of the rotary coding switch.
15		15		This setting resets the codes for IP configuration: • C13000 (IP address) • C13001 (subnet mask) • C13002 (gateway address) • C13005 (IP configuration reference) • C13006 (Multicast IP start address) • C13017 (Ethernet setting: Autonegotiation) • C13018 (Multicast setting: Default algorithm) • C13021 (Quality of Service (VLAN tagging): deactivated) • C13846 (Address conflict detection (ACD): activated)

Example: Setting of the node address '52'

Position of the rotary coding switches				Resulting node address
				
3	× 16	**4**	× 1	(3 × 16) + (4 × 1) = **52**

6.4 Setting the IP configuration of the Inverter Drive 8400

IP configuration is necessary in order to assign an address to the Inverter Drive 8400 so that communication between the PC/»Engineer« or the scanner and the drive is possible via EtherNet/IP. For this purpose, an IP address, subnet mask and gateway address have to be assigned. You can assign these IP parameters for the Inverter Drive 8400 in the following ways:

- Setting via the EtherNet/IP configurator of the »Engineer« (📖 43)
- Setting via codes in the »Engineer« (📖 45)
- Setting via a BOOTP/DHCP server (📖 47)
- Setting via the TCP/IP Interface Object (0xF5) (📖 47)

Note!

- The assignment of invalid combinations of IP address, subnet mask, and gateway address can have the consequence that no connection to the EtherNet/IP network can be established.
- Codes C13010 (IP address), C13011 (subnet mask), C13012 (gateway address), and C13016 (multicast IP address) show the IP parameters currently used.
- In the case of impermissible settings, the error message EtherNet/IP: Invalid IP parameters [0x01bc6533] (📖 99) is output.

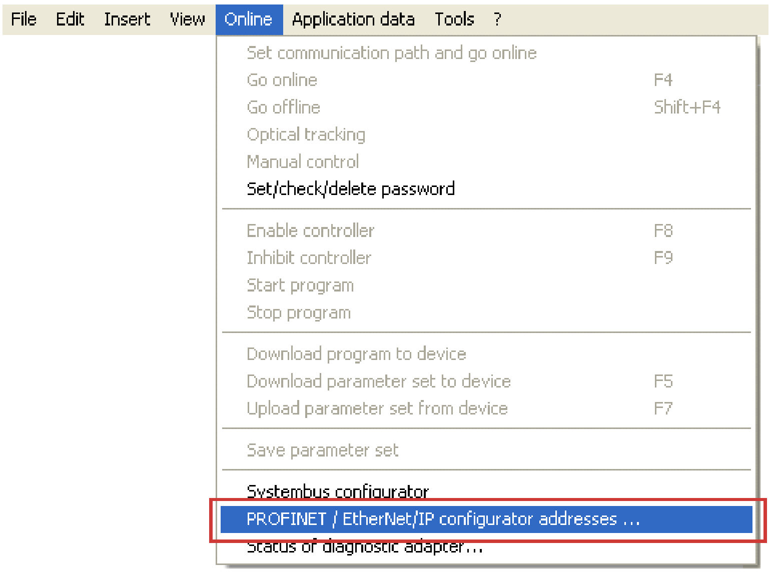
6.4.1 Setting via the EtherNet/IP configurator of the »Engineer«

Note!

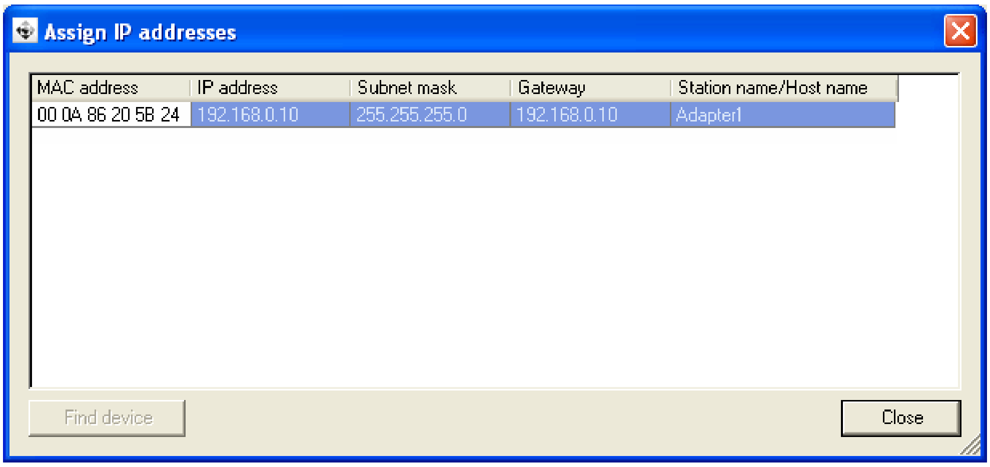
- Changes in the IP parameters will become effective immediately.
- An already existing IP connection to the Inverter Drive 8400 is interrupted.

How to set the IP parameters via the EtherNet/IP configurator:

1. Execute the menu command **Online → PROFINET /EtherNet/IP configurator addresses ...**.

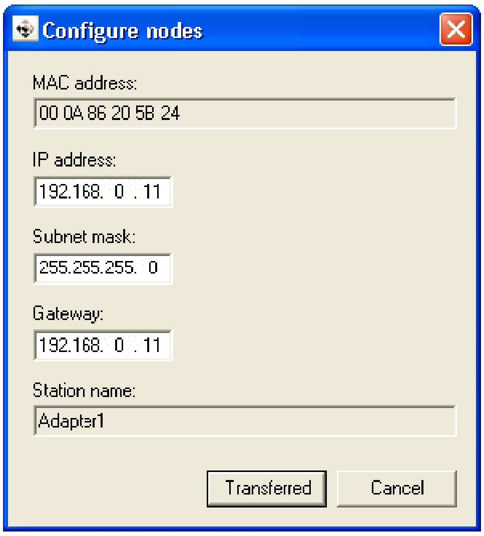


The "Assign IP addresses" dialog window is opened and all Lenze EtherNet/IP nodes connected are listed.



2. A double-click on a EtherNet/IP node opens the "Configure nodes" dialog window.

   Here you can set the IP parameters.



3. Click on **Transferred**.
   - The IP configuration is transferred to the corresponding EtherNet/IP node.
   - The communication module carries out a stack reset.
   - The IP parameters are written to codes C13000 (IP address), C13001 (subnet mask), and C13002 (gateway address).
   - Code C13005 (IP configuration reference) is set to '0: Saved address' to ensure that the transferred address can be used.

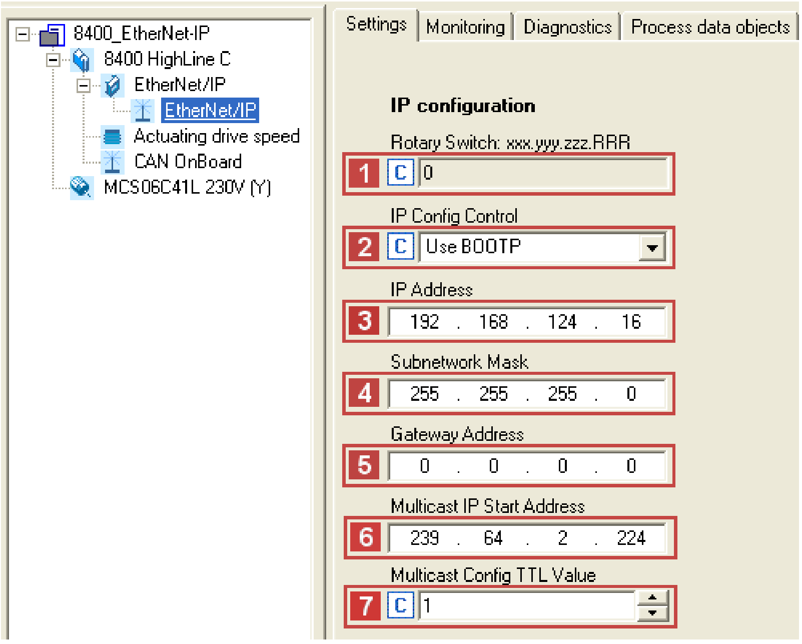
Tip!

Check whether the configuration has been transferred successfully.

For this purpose, open the "Assign IP addresses" dialog window (see step 1) and click the **Find device** button.

6.4.2 Setting via codes in the »Engineer«

You can also set the IP parameters manually in the »Engineer« under the **Settings** tab. The values will be transferred to the corresponding codes.



Settings / Display	Description
1 **Rotary coding switch**	Display of the value set by means of the rotary coding switches (C13920). ▸ Setting node addresses by means of rotary coding switches (📖 40)
2 **IP Config Control**	Selection (C13005) of how the IP configuration is to be made: • **0**: The IP configuration currently stored in the communication module is used. • **1**: The IP configuration is assigned by a BOOTP server using the BOOTP. • **2**: The IP configuration is assigned by a DHCP server using the DHCP.
3 **IP Address**	Setting of the IP address (C13000)
4 **Subnet mask**	Setting of the subnet mask (C13001)
5 **Gateway Address**	Setting of the gateway address (C13002)
6 **Multicast IP Start Address**	Setting of the Multicast IP start address (C13006) ▸ Setting the multicast configuration (📖 48)
7 **Multicast Config TTL Value**	Setting of the multicast TTL value (C13019)

How to activate changed settings in the »Engineer«:

1. Execute device command **C00002 = "11: Save all parameter sets"**.

   The current IP configuration is stored in the memory module of the drive.

2. Carry out a "Type 0 reset" for the Identity Object (1 / 0x01) (📖 126) of the node, or switch the voltage supply of the communication module off and on again.

IP address

The IP address is set/changed in C13000.

The IP address currently used is displayed in C13010/1...4.

Example: Display of the IP address 192.168.124.16				
Code	C13010/1	C13010/2	C13010/3	C13010/4
Value	192	168	124	16

Subnet mask

The subnet mask indicates which part of the IP address is evaluated as net ID or host ID.

Valid subnet masks are defined in accordance with RFC 1878

The subnet mask is set/changed in C13001.

The subnet mask currently used is displayed in C13011/1...4.

Example: Display of the subnet mask 255.255.255.0				
Code	C13011/1	C13011/2	C13011/3	C13011/4
Value	255	255	255	0

Gateway address

The gateway address is valid if the network address of the IP address and the gateway address are identical.

If the gateway address and the IP address are identical or if the address is ’0.0.0.0’, gateway functionality is not used.

The gateway address is set/changed in C13002.

The gateway address currently used is displayed in C13012/1...4.

Example: Display of the gateway address 192.168.124.16				
Code	C13012/1	C13012/2	C13012/3	C13012/4
Value	192	168	124	16

6.4.3 Setting via a BOOTP/DHCP server

DHCP is the acronym for "Dynamic Host Configuration Protocol". This protocol is defined in RFC 2131 and is a compatible advancement of the "Bootstrap Protocol" (BOOTP) according to RFC 951.

Both protocols enable network nodes to query information about the network configuration (e.g. the IP address) from a server via a TCP/IP network. The BOOTP/DHCP server assigns the IP address to the client dynamically from a defined address range. This means that the client receives an unambiguous IP address.

Code C13005 is used to select how the IP configuration is to be made:

- Value '0': The IP configuration currently saved in the communication module is used.
- Value '1': BOOTP is used. (Lenze standard setting)
- Value '2': DHCP is used.

The setting can also be selected by write access to attribute 3 (configuration control) of instance 1 of the TCP/IP Interface Object (245 / 0xF5) (📖 141).

6.4.4 Setting via the TCP/IP Interface Object (0xF5)

With a scanner, the IP configuration can be set via attribute 5 (interface configuration) of instance 1 of the TCP/IP Interface Object (245 / 0xF5) (📖 141).

After the IP configuration, carry out a node reset ("power off/on" or "Type 0 reset" for the Identity Object (1 / 0x01) (📖 126)).

In the »Engineer«, codes C13010 (IP address), C13011 (subnet mask), C13012 (gateway address), and C13016 (multicast IP address) show the IP parameters currently used.

6.4.5 Setting the multicast configuration

Several scanners ("Listen only" or "Input only" connections) can access multicast telegrams which are sent by drives. Settings for multicast configuration have to be carried out as well in the EtherNet/IP configuration software (z. B. »RSLogix 5000« from Rockwell).

By default, the communication module automatically generates the Multicast IP start address for I/O data transfer. The standard TTL value for Multicast transfer is ’1’; the Multicast I/O data packages are therefore distributed solely via the local network.

Note!

You can also explicitly set the multicast IP start address and the multicast TTL value. We recommend, however, not to change the default settings in order to ensure a secure multicast transmission.

The following multicast codes are configurable:

Code	Description
C13018	Selection for multicast IP addressing via instance attribute 9 (Mcast Config) in the TCP/IP Interface Object (245 / 0xF5) (🕮 141) • Value ’0’: The default algorithm is used. • Value ’1’: The address from code C13006 is used as multicast IP start address.
C13019	Setting of the multicast TTL value for the validity time of data packets in the EtherNet/IP network (Instance attribute 8 (TTL Value) in the TCP/IP Interface Object (245 / 0xF5) (🕮 141))
C13020	Used to set how many multicast IP addresses will be assigned. (Instance attribute 9 (Num Mcast) in the TCP/IP Interface Object (245 / 0xF5) (🕮 141))

Multicast IP Start Address

Multicast IP start addresses serve to send a message to the members of a certain group (i.e. possibly to several nodes).

The multicast IP start address is set/changed in C13006.

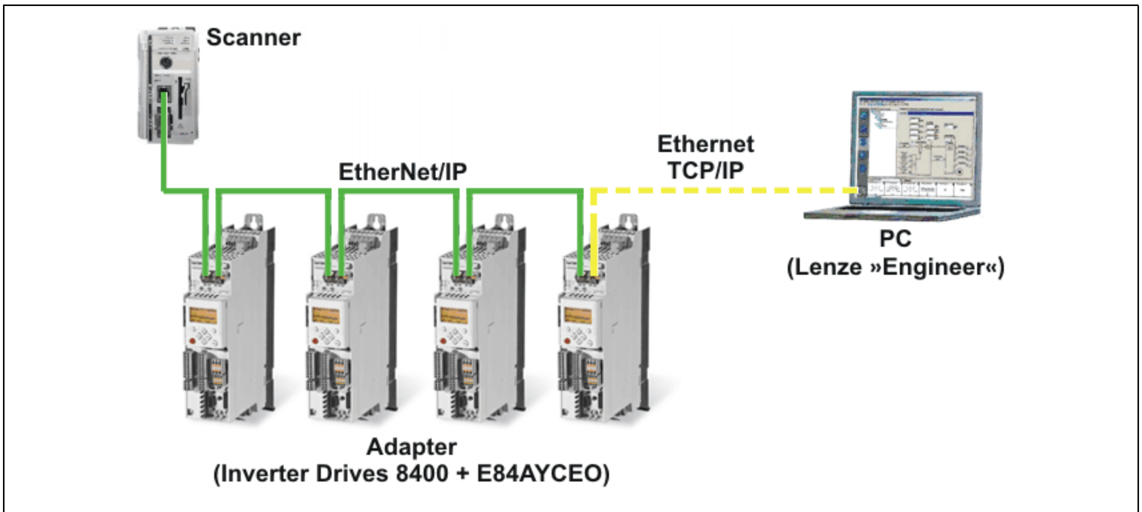
The currently used multicast IP address of the drive is displayed in C13016/1...4.

Example: Display of the multicast IP address 239.64.2.224				
Code	C13016/1	C13016/2	C13016/3	C13016/4
Value	239	64	2	224

6.5 Establishing an online connection via EtherNet/IP with the Lenze »Engineer«

Note!

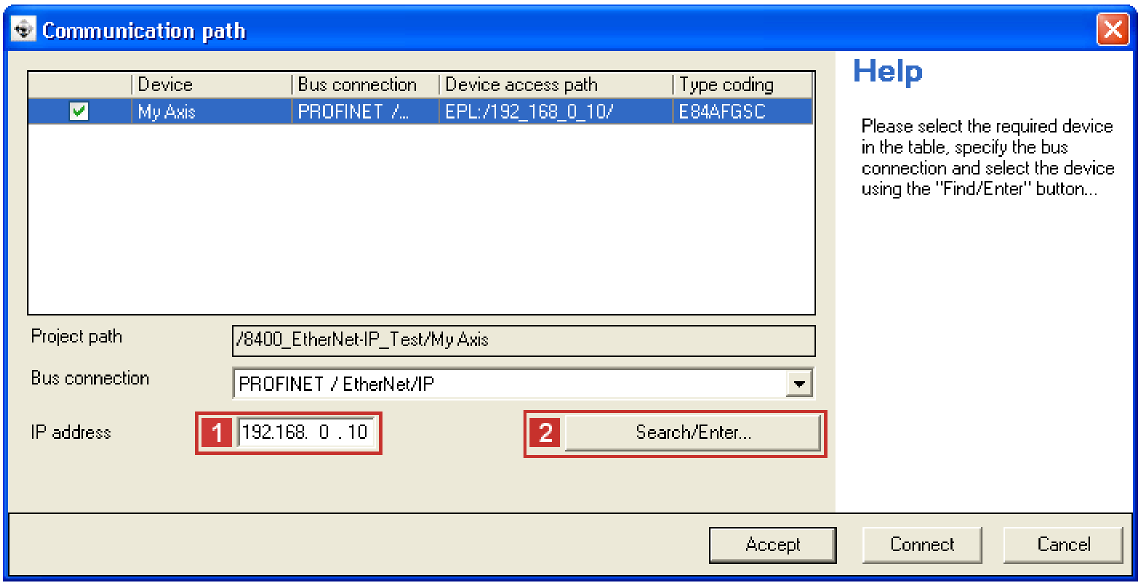
- In order to ensure perfect operation of cyclic EtherNet/IP communication, online access with the »Engineer« should be executed via an IEEE 802.1Q-capable switch.
- The IEEE 802.1Q-capable switch integrated in the communication module can manage cyclical EtherNet/IP-communication primarily for normal TCP/IP communication. In the case of EtherNet/IP, this is done by means of the VLAN identification in the Ethernet frame (can be set in C13021).
- If the redundancy protocol DLR (Device Level Ring) is used, the switch also must be DLR-compliant.



[6-2] Example set-up with an Allen Bradley CompactLogix Controller 1769-L32E (scanner)

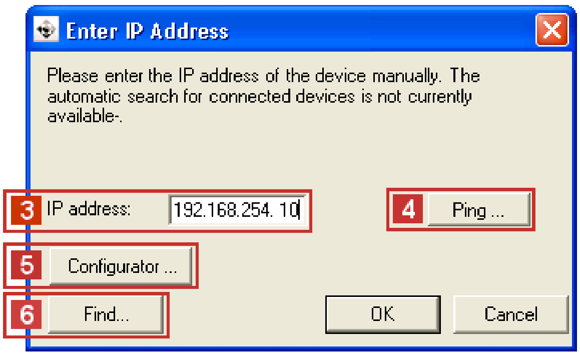
For an online connection between the »Engineer« and the drive, the drive must have an IP address (see Setting the IP configuration of the Inverter Drive 8400 (📖 42)).

In the »Engineer«, you can use the **Online → Set communication path and go online** menu command to select the EtherNet/IP communication path. The previously configured EtherNet/IP nodes are shown in the "Communication path" dialog window:



If the device access path is not configured correctly, the 1 **IP address** of the drive selected in the display field can be entered manually here.

Via the 2 **Search/Enter** button, you can establish a connection to devices which have not appeared in the display field. Corresponding settings for this can be made in the "Enter IP Address" dialog window that will appear:



Here you can enter an 3 **IP address** manually or execute the following actions using the buttons:

- Execute the console command 4 **Ping**.
- Assign the IP address via the 5 **Configurator**.
  ▸ Setting via the EtherNet/IP configurator of the »Engineer« (📖 43)
- Select the device access path to the desired drive by clicking 6 **Find**.

After having established the online connection, you can continue work with the »Engineer« as usual.

6.6 Initial switch-on

Documentation for the Inverter Drive 8400

Observe the safety instructions and information on residual hazards.

Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Activating changed setting

In order to activate any changed settings, ...

- execute the device command "11: Save all parameter sets" via the standard device code **C00002** and ...
- Carry out a "Type 0 reset" for the Identity Object (1 / 0x01) (📖 126) of the node, or switch the voltage supply of the communication module off and on again.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

The restart protection is activated in the Lenze setting of the Inverter Drives 8400.

You can set the restart behaviour of the drive via **C00142** ("Autostart Option"):

C00142 = 9 (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- Bit 0 (inhibit at "power-on") and bit 3 (inhibit in the case of undervoltage) are set.
- An explicit inverter enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X4/RFR.

C00142 = 8 (enabled)

- In order to directly enable the device at switch-on, bit 0 must be set to zero (FALSE).
- An uncontrolled restart of the drive is possible.

7 Data transfer

EtherNet/IP uses CIP™ (Common Industrial Protocol) for the data exchange between devices via an Ethernet network – just like the closely related bus systems DeviceNet and ControlNet.

Lenze implements the CIP following the ODVA standard (Open DeviceNet Vendor Association, www.odva.org) and supports the two main types of EtherNet/IP communication:

- Explicit messaging (for parameter data)
- Implicit messaging (for I/O data)

7.1 Communication channels

Note!

The terms "input" and "output" refer to the point of view of the scanner:
- Input data is produced by the adapter and consumed by the scanner.
- Output data is produced by the scanner and consumed by the adapter.

EtherNet/IP transmits parameter data and I/O data between the host system (scanner) and the drives (adapters) connected to the fieldbus. The data are transmitted via corresponding communication channels depending on their time-critical behaviour.

The I/O data channel transmits I/O data by means of "implicit messages".

- The inverter is controlled by means of the I/O data.
- The transmission of I/O data is time-critical.
- I/O data are transmitted cyclically between the host system (scanner) and the drives (adapters) (permanent exchange of current input and output data).
- The host system (scanner) has direct access to the I/O data (the data are, for example, stored directly in the I/O area).
- In the case of Inverter Drives 8400, a maximum of 16 data words (max. 32 bytes) can be exchanged per direction.
- I/O data are not stored in the inverter.
- I/O data are e.g. setpoints, actual values, control and status words

The parameter data channel transmits parameter data by means of "explicit messages".

- The transmission of parameter data is usually not time-critical.
- Examples of parameter data are operating parameters, motor data, and diagnostic information.
- The parameter data channel provides access to all Lenze codes.
- Parameter changes must be saved by means of code **C00002** of the Inverter Drive 8400.

7.2 Telegram types

The "implicit message" and "explicit message" telegram types are transmitted between the host system (scanner) and the drive (adapter).

Implicit messages (I/O data transfer)

"Implicit messages" are transmitted or received according to the producer/consumer principle. There is one transmitter and no receiver or an optional number of receivers.

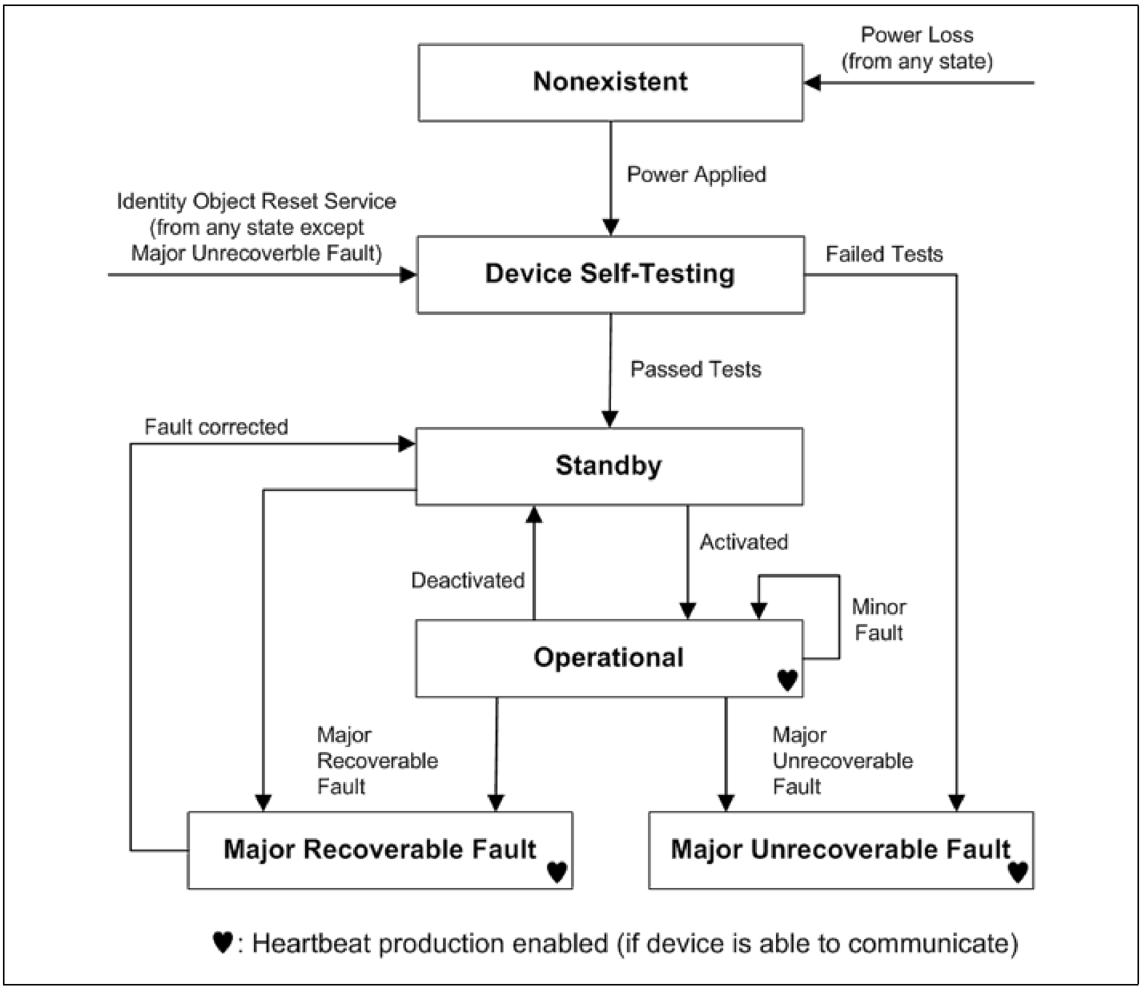
The "cyclic I/O data" transmission mode is supported. The scanner and the adapter use "cyclic I/O data" to generate their data independently of each other, which are then transmitted depending on a timer. The user must set the value of the timer in the scanner.

Explicit messages (parameter data transfer)

"Explicit messages" serve to configure and parameterise the individual EtherNet/IP nodes.

Two nodes have a client/server relationship:
The client transmits a job (request). The server receives this job and tries to accomplish it. The server then transmits the requested data (positive response) or an error message (negative response).

7.3 EtherNet/IP state diagram



[7-1] EtherNet/IP state diagram

The current EtherNet/IP device state is ...

- output via code C13861;
- output in the Identity Object (1 / 0x01) (📖 126) via instance attributes 5 and 8;
- indicated via the **MS** LED (see LED status displays (📖 89)).

8 I/O data transfer (implicit messages)

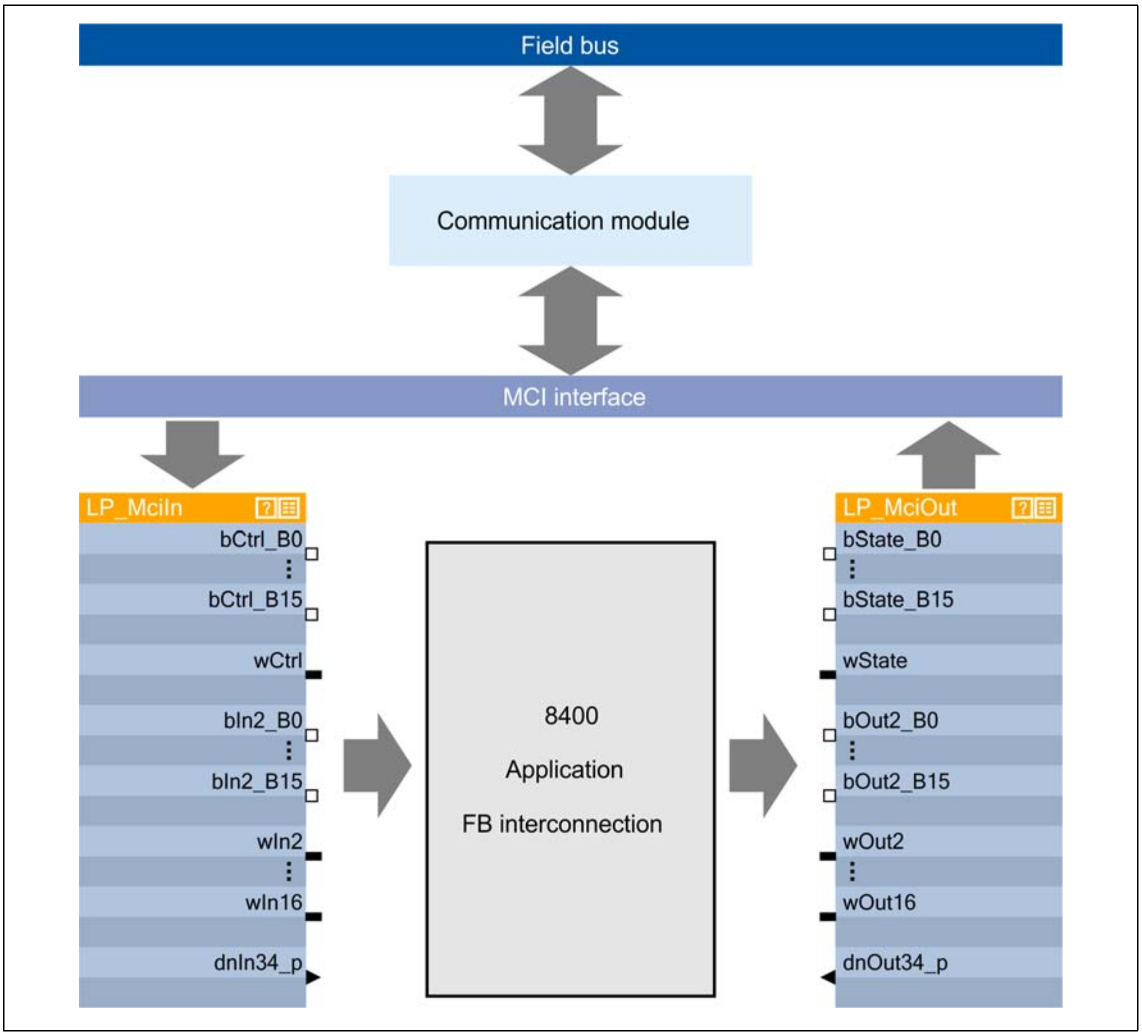
To exchange I/O data (implicit messages) between the host system (scanner) and the drive (adapter), you have to ...

- Assign the I/O data to the internal ports (MCI) in the drive (adapter):
  - I/O data mapping (📖 57)
  - I/O configuration in the »Engineer« (📖 62)
- Configure the I/O data transfer in the host (scanner):
  - I/O configuration with »RSLogix 5000« version 19 or lower (📖 68)
  - I/O configuration with »RSLogix 5000« version 20 or higher (📖 73)

8.1 I/O data mapping

I/O data transfer takes place via the MCI interface.

- Access to the I/O data takes place via port blocks **LP_MciIn** and **LP_MciOut**.
- The **LP_MciIn** port block maps the received data objects.
- The **LP_MciOut** port block maps the data objects to be sent.
- Up to 16 data words (32 bytes) per direction can be exchanged.
- The ports/function blocks of the I/O data objects are interconnected with the Lenze »Engineer«.



[8-1] Outer and inner data transfer between bus system, inverter, and application

Software manual / online help for the Inverter Drive 8400

Here you will find detailed information on the port/function block interconnection in the »Engineer« and on the port blocks.

8.2 Technology applications (TA) / drive profiles

The Inverter Drives 8400 is provided with various drive profiles. They define a standardised/ individual control and status word assignment and the standardisation of setpoints and actual value scalings.

The following drive profiles are supported by the Inverter Drives 8400:

- Lenze technology applications / user-definable parameter sets
- "AC Drive Profile" application

8.2.1 Lenze technology applications / user-definable parameter sets

The technology applications integrated in the drive provide the main signal flow for realising a general or specific drive solution.

For using the Lenze technology application selection in the »Engineer« via standard device code **C00005**, the following assembly object instances have to be used in the host (scanner):

Instance ID		Assembly object instance
[dec]	[hex]	
110	0x6E	Custom Output (from the adapter to the scanner)
111	0x6F	Custom Input (from the adapter to the scanner)

See also Assembly Object (4 / 0x04) (📖 129).

The custom assemblies also allow for a user-definable parameter setting, depending on the application. 16 data words can be freely assigned with variables of the MCI port blocks in the »Engineer«.

The user-definable parameter setting can be used in addition to the previously set technology application.

▶ Lenze technology application / configuring user-definable parameter sets (📖 62)

Tip!

Here you will find information on configuring with the »RSLogix 5000« programming software from Rockwell:

▶ I/O configuration with »RSLogix 5000« version 19 or lower (📖 68)
▶ I/O configuration with »RSLogix 5000« version 20 or higher (📖 73)

8.2.2 "AC Drive Profile" application

From inverter drive version V13.00 and communication module version V01.02, the EtherNet/IP-specific "AC Drive Profile" is supported.

The standard device code **C00005 = "1100: AC Drive Profile"** serves to select the "AC Drive Profile" application.

The "AC Drive Profile" contains ...

- the data basis for motor parameters,
- management functions of the motor control devices,
- Device-specific functions of the drive, e.g. speed ramps, torque control etc.

For using the "AC Drive Profile", the following assembly object instances in the host (scanner) have to be used:

Instance ID		Assembly object instance	
[dec]	[hex]		
20	0x14	Basic Speed Control Output	Outputs: From the scanner to the adapter
21	0x15	Extended Speed Control Output	
22	0x16	Speed and Torque Control Output	
23	0x17	Extended Speed and Torque Control Output	
70	0x46	Basic Speed Control Input	Inputs: From the adapter to the scanner
71	0x47	Extended Speed Control Input	
72	0x48	Speed and Torque Control Input	
73	0x49	Extended Speed and Torque Control Input	

See also:

- Assembly Object (4 / 0x04) (📖 129)
- "AC Drive Profile" objects (📖 148)

Tip!

Here you will find information on configuring with the »RSLogix 5000« programming software from Rockwell:

▸ I/O configuration with »RSLogix 5000« version 19 or lower (📖 68)
▸ I/O configuration with »RSLogix 5000« version 20 or higher (📖 73)

8.3 I/O assemblies

Note!

The terms "input" and "output" refer to the point of view of the scanner:
- Assembly input objects (input) are sent from the adapter to the scanner.
- Assembly output objects (output) are sent from the scanner to the adapter.

The length of the I/O data must correspond with the respective resulting length of the mapped ports (I/O data mapping (📖 57)).

The communication module supports the Assembly Object (4 / 0x04) (📖 129) and the "AC Drive Profile" objects (📖 148).

For data exchange, the communication module supports the following assembly object instances:

Application	Instance ID		Assembly object instance
	[dec]	[hex]	
Lenze technology applications / user-definable parameter sets	110	0x6E	Custom Output
	111	0x6F	Custom Input
"AC Drive Profile" application	20	0x14	Basic Speed Control Output
	21	0x15	Extended Speed Control Output
	22	0x16	Speed and Torque Control Output
	23	0x17	Extended Speed and Torque Control Output
	70	0x46	Basic Speed Control Input
	71	0x47	Extended Speed Control Input
	72	0x48	Speed and Torque Control Input
	73	0x49	Extended Speed and Torque Control Input

Assembly output objects (outputs) are usually used for controlling the enable/disable state of the drive and for supplying the speed or torque references.

Assembly input objects (inputs) are usually used to monitor the drive status and the runtime values such as actual speed, current, actual position and position error.

Depending on the data length defined by the scanner, the memory map of the I/O data can have different sizes.

Assembly output objects (scanner → Adapter)

Assembly output objects are assumed to have a 4-byte header (32-bit "run/idle header"). When mapping the assemblies, this header will automatically be added to the data stream by most Allen-Bradley PLC/SLC equipment.

If your PLC does not support this header (like the Rockwell PLCs do), add a preceding 32-bit header to the output image.

You can then define the **bit 0** of this header in the process image of your PLC:

- 0: Idle mode
- 1: Run mode

For the operation with Rockwell PLCs, adaptations are not required.

Assembly input objects (adapter → scanner)

The assembly input objects are mapped in the adapter memory starting with byte 0.

The input objects are transmitted in a "modeless" manner, i.e. a 4-byte header (32-bit "Run/Idle header") is not included in the transfer.

So the start address in the assembly memory map is the actual start of the first assembly data element.

Please observe the actual assembly lengths when mapping the input objects to the controller memory.

The contents of the input data depends on the I/O data arrangement in the drive (I/O data mapping (🕮 57)).

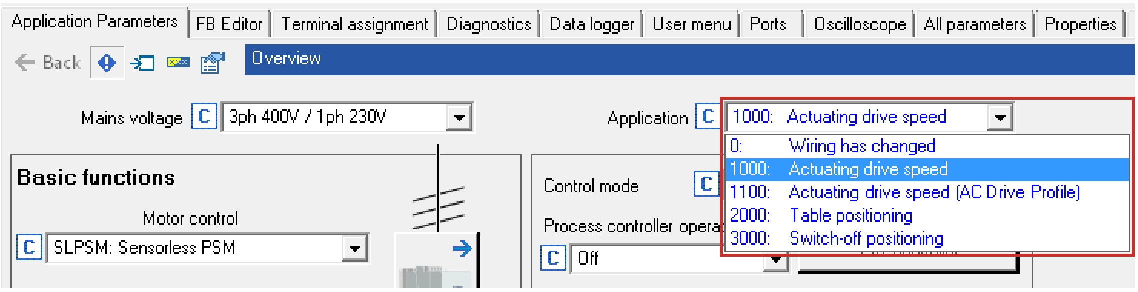
8.4 I/O configuration in the »Engineer«

8.4.1 Lenze technology application / configuring user-definable parameter sets

How to configure the Lenze technology applications / user-definable parameter sets in the »Engineer«:

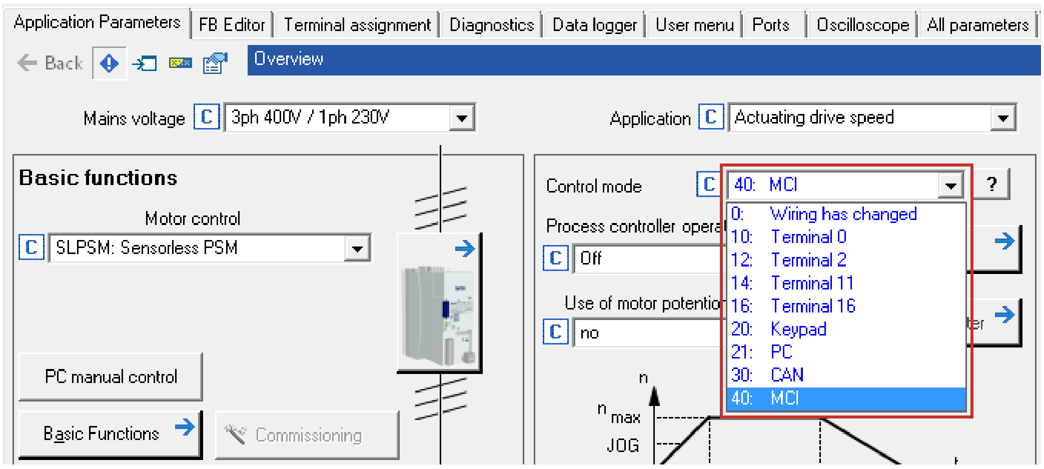
1. Select the application (C00005 = 1000) in the **Application parameters** tab.

   In the example, the "speed actuating drive" application is configured.

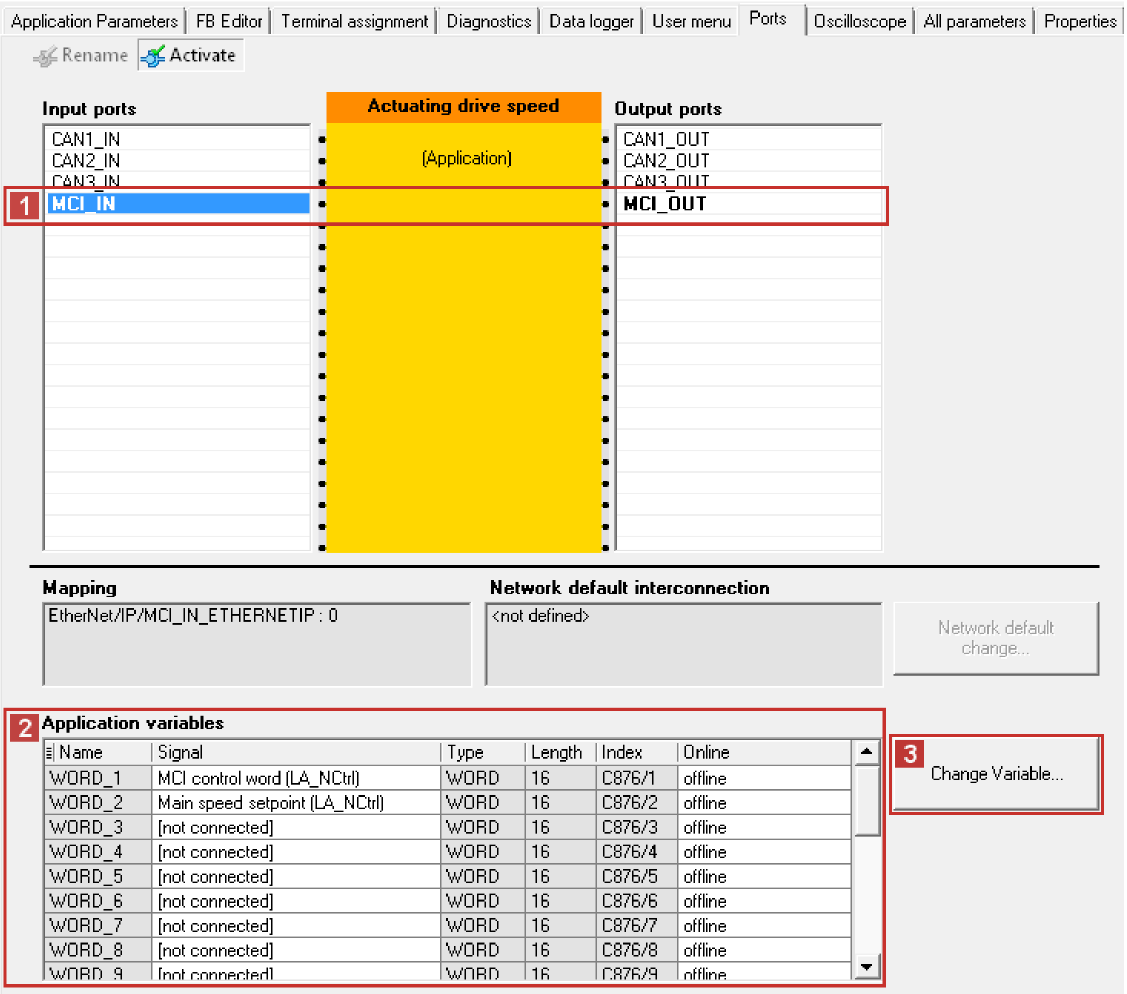


2. Make the default setting of the I/O configuration.

   Select **"MCI"** control mode (C00007 = 40).

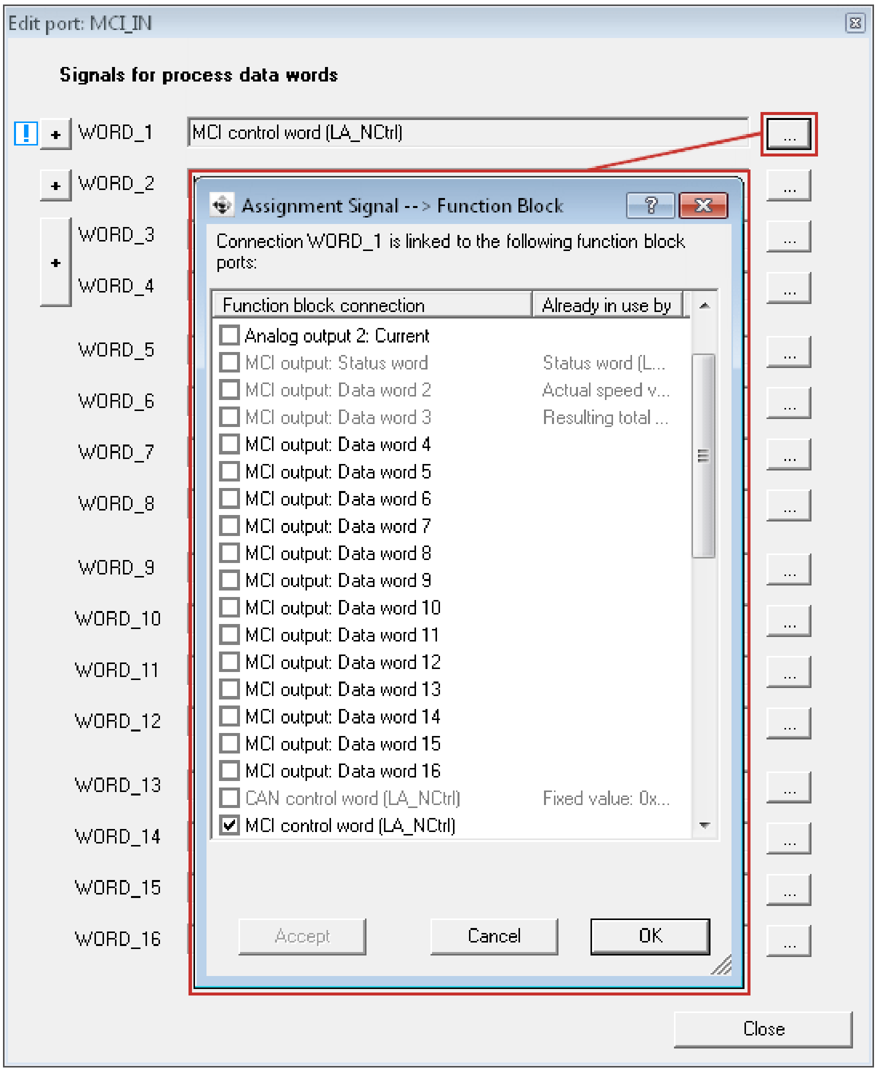


3. On the **Ports** tab, the port blocks 1 **MCI_IN** and **MCI_OUT** for the I/O data objects are shown.



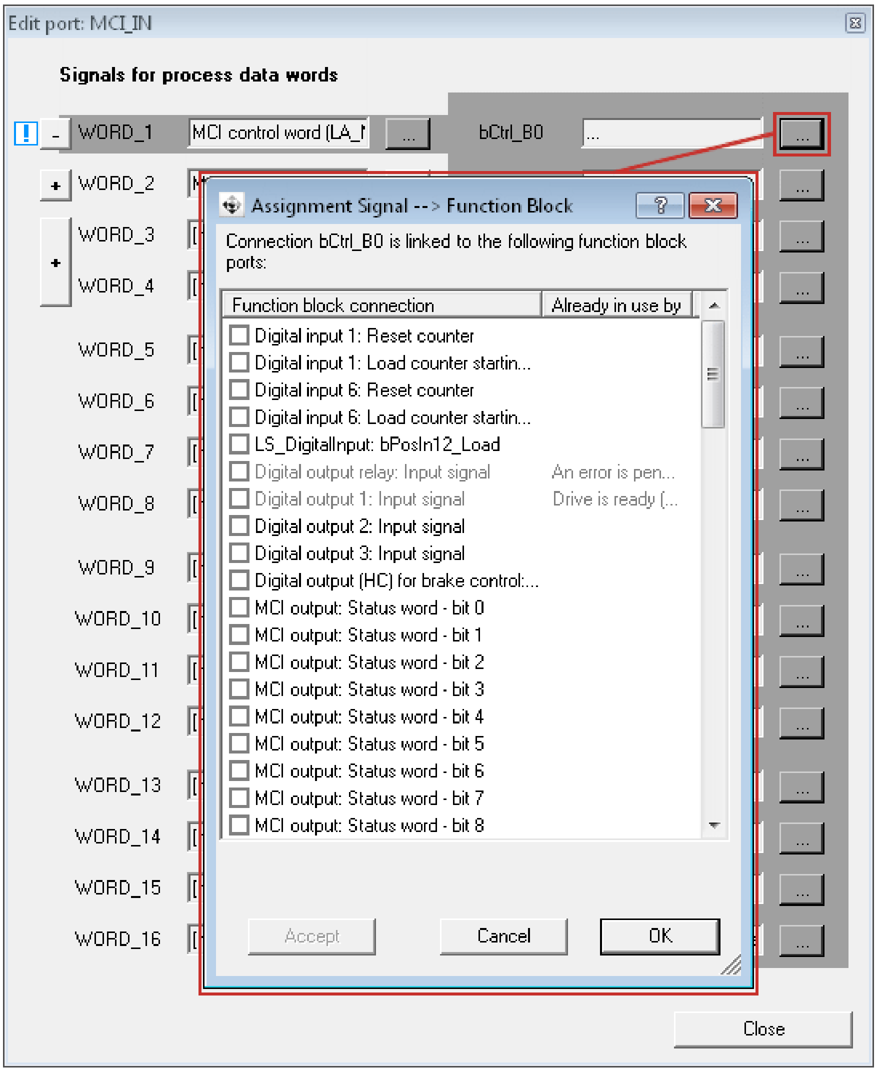
- By clicking the required port, the preconfigured signal combination can be obtained from the 2 **application variables**.
- If you want to complement or change the signal combination, click the 3 **Change Variable ...** button.

4. In the "Edit port" dialog window, you can assign signals to the I/O data words via the ... button.
   → Select the signals and confirm the selection with **OK**.



For some data words, you can also assign signals to the individual bits via the + and ... buttons.

→ Select the signals and then confirm the selection with **OK**.



5. Use the standard device code **C00002** to execute the command **"11: Save all parameter sets"**.

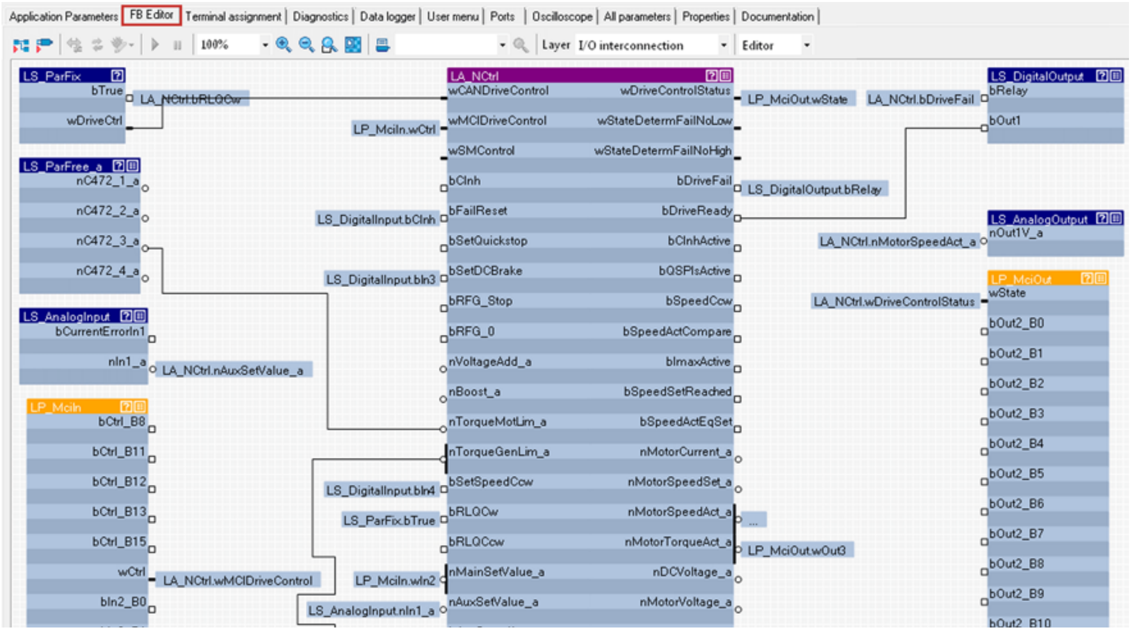
   The changed settings are activated and saved with mains failure protection.

Preconfigured signal combination in the "FB Editor"

By setting the standard device code **C00007 = "40: MCI"** , the preconfigured signal combination is activated.

The function block editor (FB Editor) is used to display the port blocks **LP_MciIn** and **LP_MciOut** with the preconfigured signal combinations.

Here, you can also complement or change the signal combination.



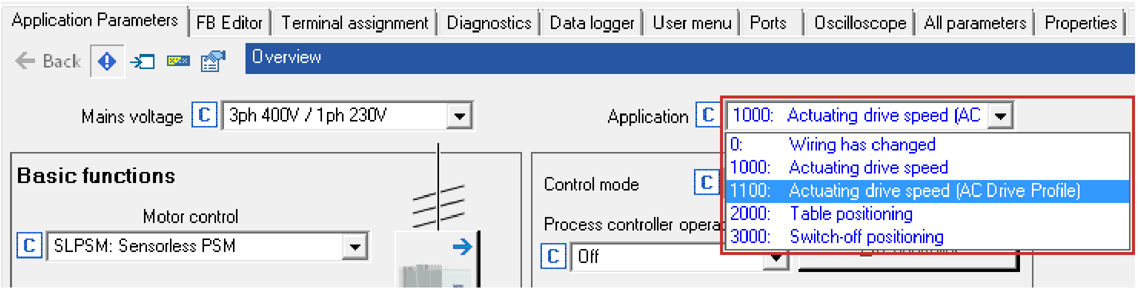
Software manual / online help for the Inverter Drive 8400

Here you will find detailed information on the port/function block interconnection in the »Engineer« and on the port blocks.

8.4.2 Configuring "AC Drive Profile" application

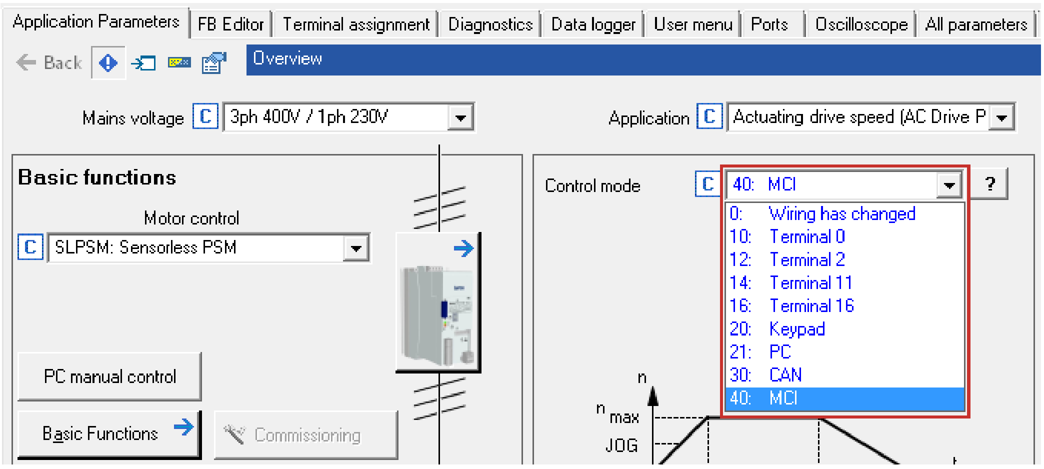
How to configure the "AC Drive Profile" application in the »Engineer«:

1. Select the "AC Drive Profile" application (C00005 = 1100) in the **Application parameters** tab.



2. Make the default setting of the I/O configuration.

   Select **"MCI"** control mode (C00007 = 40).



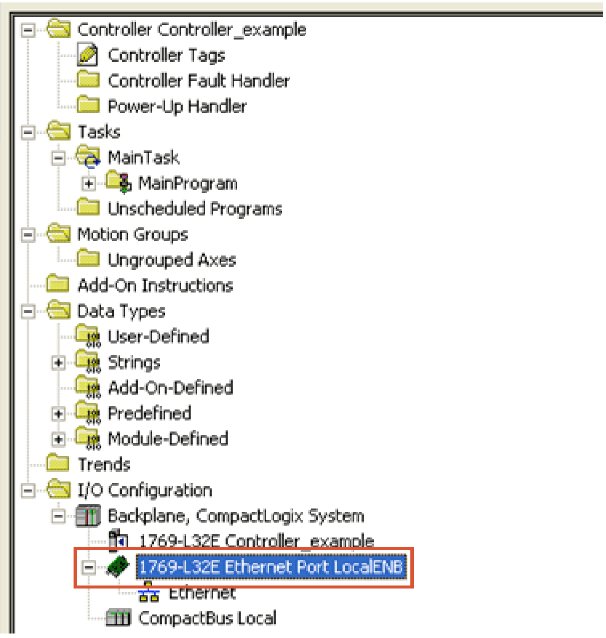
8.5 I/O configuration with »RSLogix 5000« version 19 or lower

The following example describes the I/O configuration of the Allen-Bradley 1769-L32E CompactLogix controller using the Rockwell »RSLogix 5000« programming software up to version 19.

Up to and including software version 19, the I/O configuration is carried out without EDS files.

How to carry out the I/O configuration, taking the CompactLogix controller 1769-L32E with »RSLogix 5000« as an example:

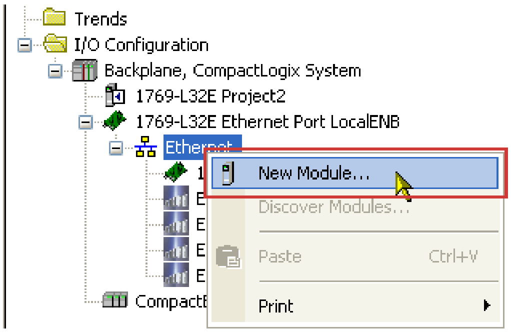
1. Click the **I/O Configuration** folder in the configuration tree.



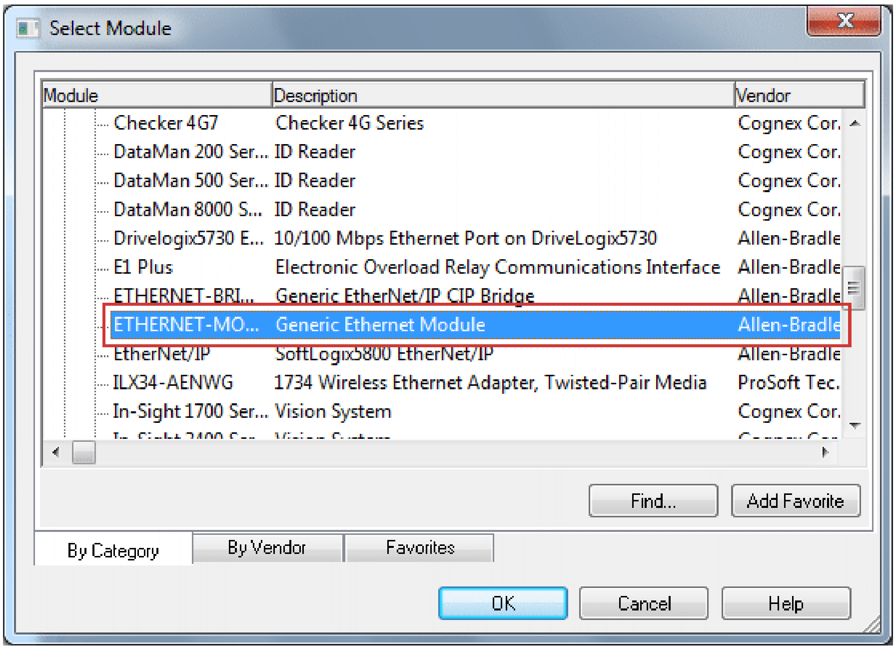
For the 1769-L32E CompactLogix controller, the I/O configuration already includes a local Ethernet port.

If a SoftLogic or ControlLogix controller is used, an Ethernet port scanner needs to be added to the configuration.

2. Right click on "Ethernet" and execute the **"New Module ..."** command from the context menu.



3. Open "Communications" and select "ETHERNET-MODULE | Generic Ethernet Module".

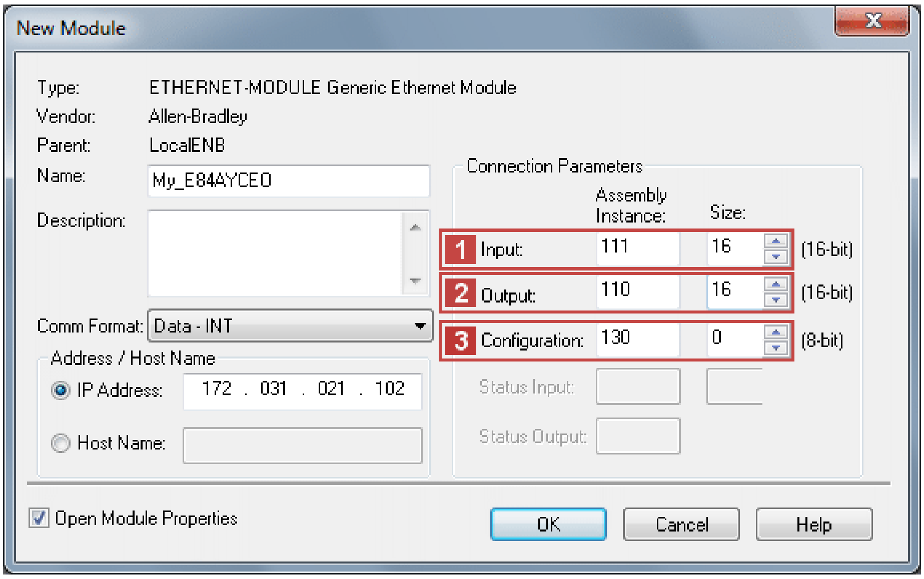


4. Confirm the selection with **OK**.

5. Go to the "New Module" dialog window and define the properties of the newly added device .

   The terms "input" and "output" refer to the point of view of the scanner:
   - Assembly input objects (input) are sent from the adapter to the scanner.
   - Assembly output objects (output) are sent from the scanner to the adapter.

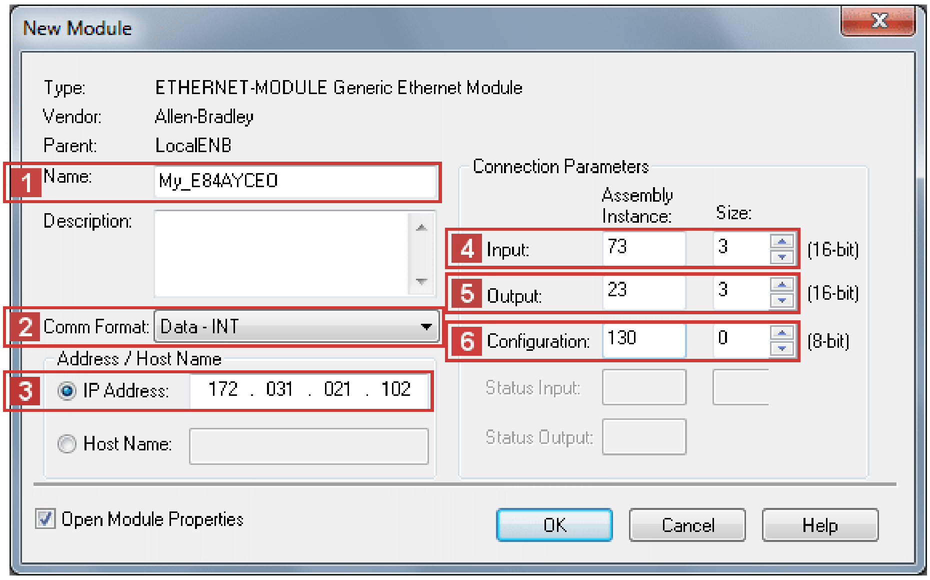
Settings for Lenze technology applications or user-definable parameter sets:



If you use a Lenze technology application or an individual parameter set in the drive, you can exchange up to 16 user-definable words (INT) via the ports MCI_IN and MCI_OUT using the assembly object instances 1 **"Input = 111"** and 2 **"Output = 110"**.

Go to 3 "Configuration" and enter the **Assembly instance "130"** and the **Size "0"**.

Settings for an "AC Drive Profile" application:

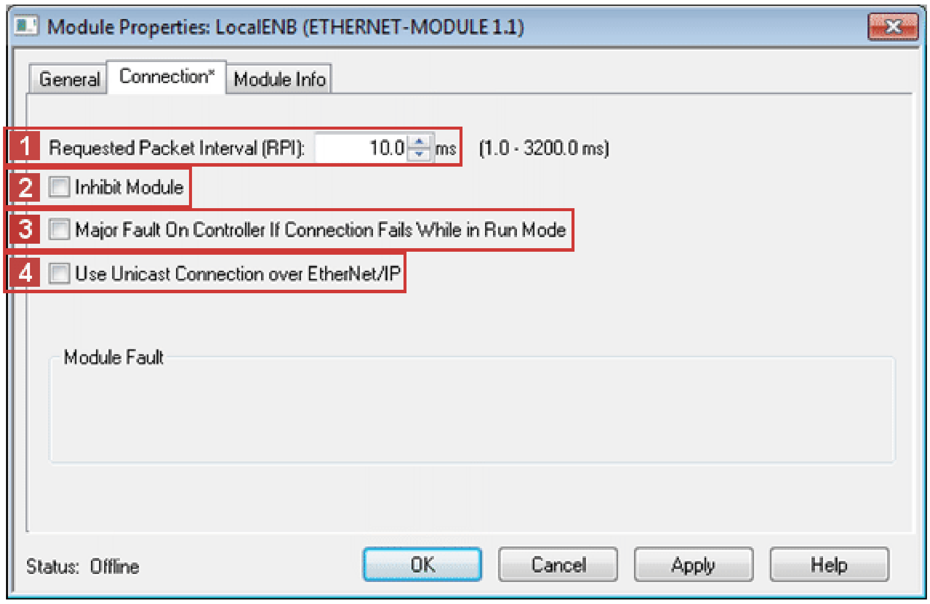


The assembly object instances 4 **"Input = 73"** and 5 **"Output = 23"** shown here exemplify the use of the AC Drive Profile "Extended Speed and Torque".

Settings		Description
1	**Name**	Device name or type of the drive, usually with reference to the process (in the example "My_E84AYCEO")
2	**Comm Format**	Data format for the assembly object instances ("connection parameters" area)
3	**IP Address**	IP address of the drive • The IP address must be in the same subnetwork like the controller. (Depending on the subnet mask; in general, the first 3 octets of the IP address must be identical.) • DNS is not supported; the host name merely describes the device.
4	**Input**	Assembly object instance for input objects • Max. 16 input data words (32 bytes, 16 bits/word) • The number of input data must correspond to the length of the mapped ports in the transmit PDO (PDO_TX0), otherwise the adapter will reject the connection with the error message "Invalid Target to Originator Size" (0x0128). ▸ I/O data mapping (📖 57)
5	**Output**	Assembly object instance for output objects • Max. 16 output data words (32 bytes, 16 bits/word) • The number of output data must correspond to the length of the mapped ports in the receive PDO (PDO_RX0), otherwise the adapter will reject the connection with the error message "Invalid Originator to Target Size" (0x0127). ▸ I/O data mapping (📖 57)
6	**Configuration**	Select the **Assembly instance "130"** and the **Size "0"** for the configuration. These values are required!

6. Complete the settings with **OK**.

7. Under **I/O Configuration** in the configuration tree, right click "1769-L32E Ethernet Port LocalENB" and select "Properties".
8. Go to the **Connection** tab and set further properties.



Required setting

Setting		Description
1	**Requested Packet Interval (RPI)**	Set **RPI** ≥ **4.0 ms**. (Standard setting: 10 ms) The RPI [ms] specifies the intervals for the I/O data exchange between the drive (adapter) and the controller (scanner).

Optional settings

Settings		Description
2	**Inhibit module**	This option serves to interrupt or inhibit the communication to the adapter.
3	**Major Fault On Controller If Connection Fails While In Run Mode**	This option serves to put the controller into the error status if the EtherNet/IP connection to the drive fails while the controller is running.
4	**Use Unicast Connection over EtherNet/IP**	**Option deactivated (standard setting):** • The input data is sent from the adapter to the scanner by means of **multicast telegrams**. • Other scanners can access this data in addition to the currently configured scanner ("Listen only" or "Input only" connections).
		Option activated: The input data is sent from the adapter to the scanner by means of **unicast telegrams**.

9. Complete the settings with **OK**.
   - The I/O configuration is now complete.
   - The corresponding tags will then be created in the "Controller Tags" of the controller project.
10. The last step is Saving the I/O configuration in »RSLogix 5000« (83).

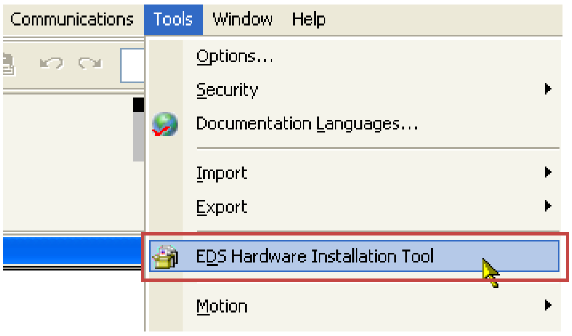
8.6 I/O configuration with »RSLogix 5000« version 20 or higher

The following example describes the I/O configuration of the Allen-Bradley 1769-L32E CompactLogix controller using the Rockwell »RSLogix 5000« programming software from version 20.

From software version 20 onwards, EDS files (37) are used for the I/O configuration.

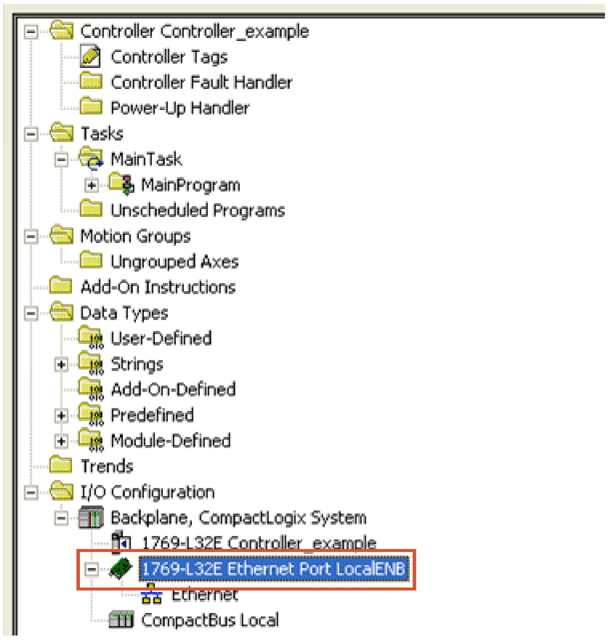
How to carry out the I/O configuration, taking the CompactLogix controller 1769-L32E with »RSLogix 5000« as an example:

1. Use the "EDS Hardware Installation Tool" item to import the EDS files (37) of the EtherNet/IP nodes.



In »RSLogix 5000«, the dialog for the "EDS Hardware Installation Tool" is self-explanatory and not described further in this documentation.

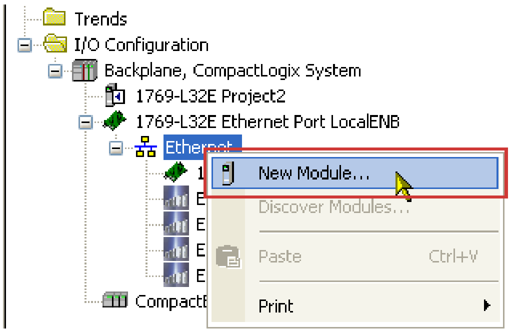
2. Click the **I/O Configuration** folder in the configuration tree.



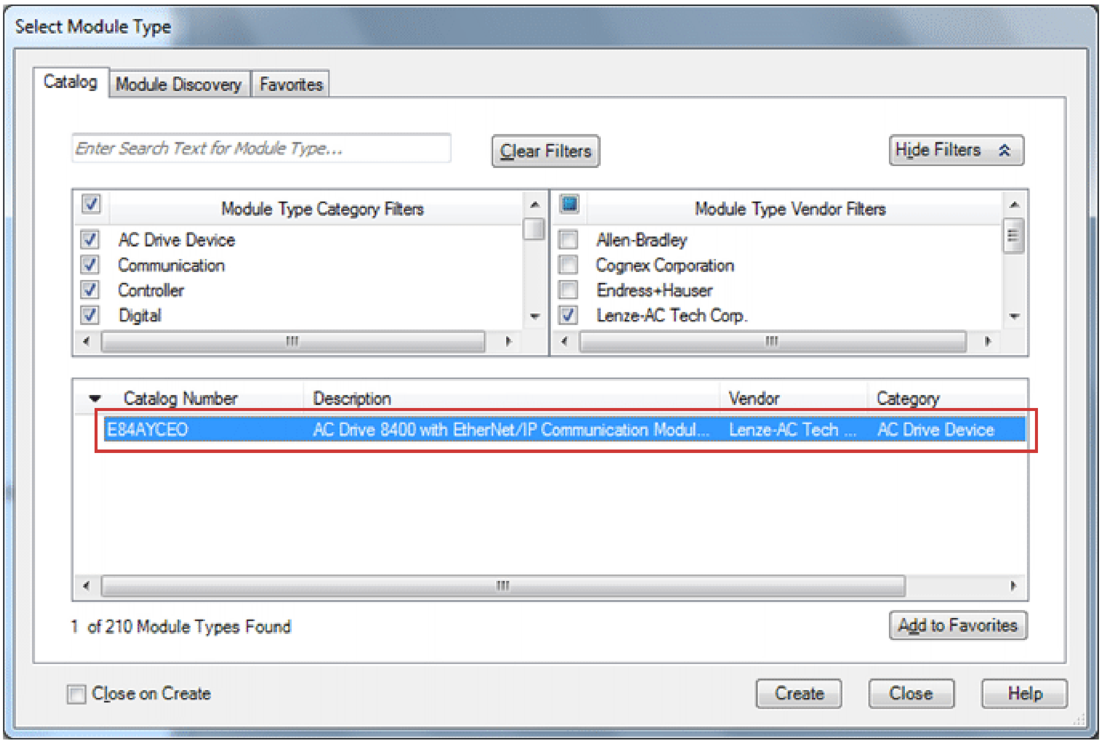
For the 1769-L32E CompactLogix controller, the I/O configuration already includes a local Ethernet port.

If a SoftLogic or ControlLogix controller is used, an Ethernet port scanner needs to be added to the configuration.

3. Right click on "Ethernet" and execute the **"New Module ..."** command from the context menu.



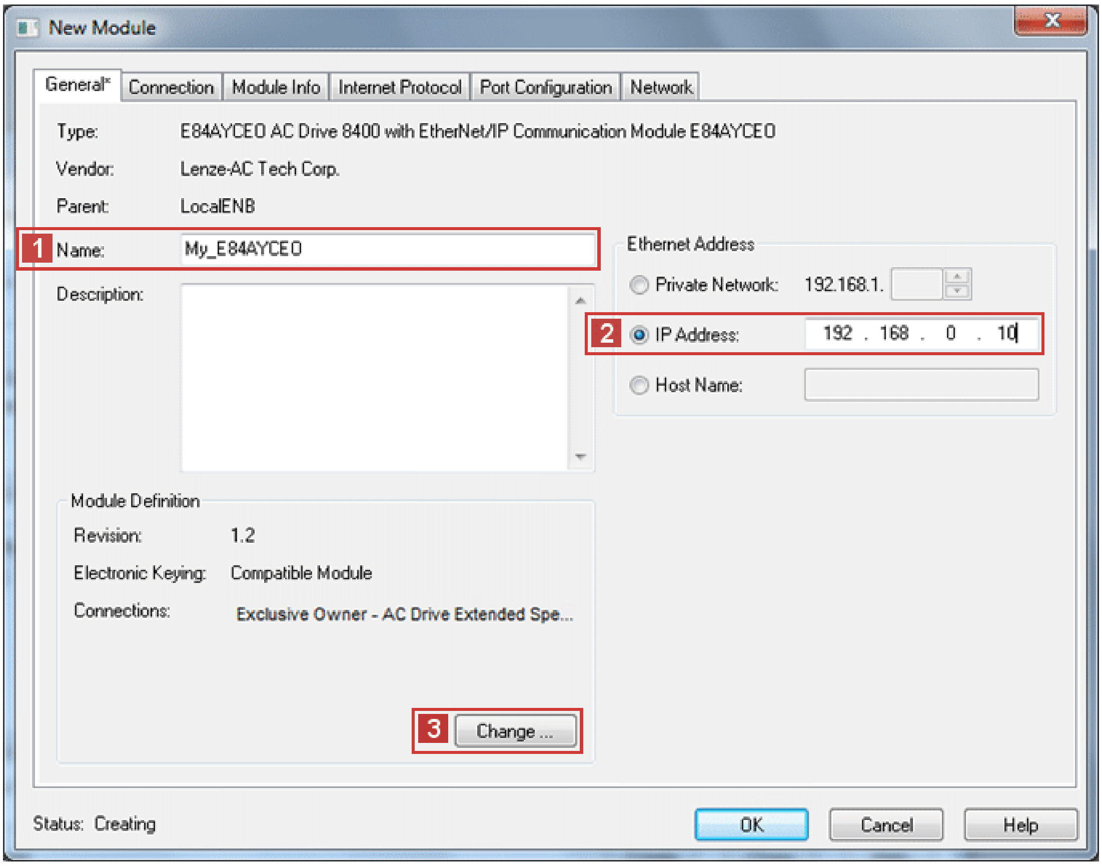
4. Go to the "Select Module Type" dialog box and select "E84AYCEO" on the **Catalog** tab.



5. Confirm the selection with **Create**.

6. In the "New Module" dialog window, select the **General** tab and allocate a 1 **name** and an unique 2 **IP address**.

   Example settings:



   DNS is not supported; the host name merely describes the device.

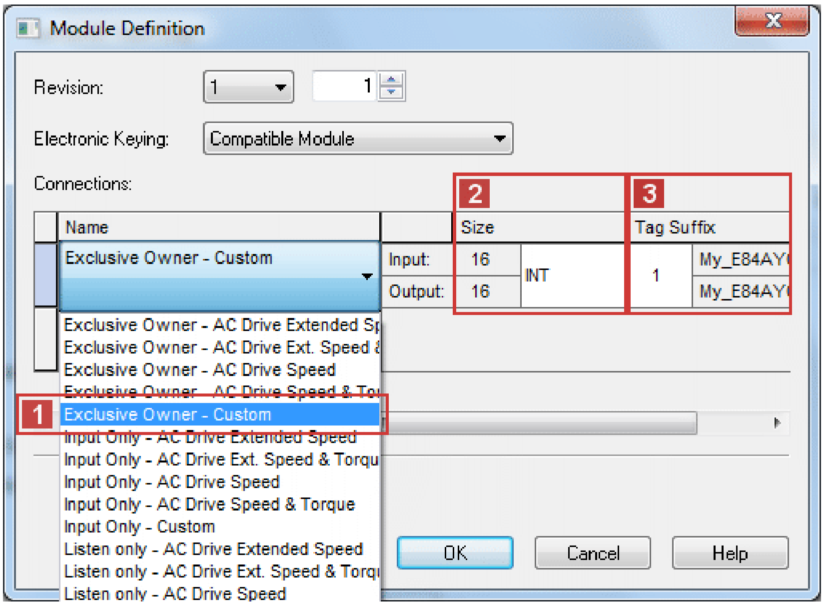
7. Click the 3 **Change ...** button.

8. Go to the "Module Definition" dialog window and make the connection settings.

   The terms "input" and "output" refer to the point of view of the scanner:
   - Assembly input objects (input) are sent from the adapter to the scanner.
   - Assembly output objects (output) are sent from the scanner to the adapter.

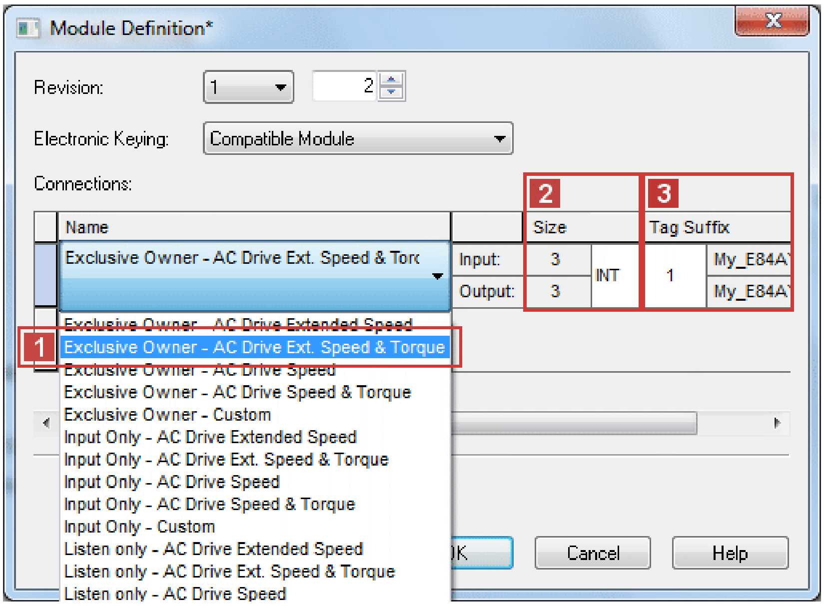
Settings for Lenze technology applications or user-definable parameter sets:

- 1 Select **"Exclusive Owner - Custom"** connection.
  If you use the Lenze technology applications or user-definable parameter sets in the drive, the "Exclusive Owner - Custom" assembly object instance always has to be selected.
- 2 Set **Data type = INT**.
  The INT data type is exchanged via the ports MCI_IN and MCI_OUT of the Inverter Drives 8400. In case of the SINT data type, an additional PLC logic is required for conversion.
- 3 Set **Tag Suffix = 1**.
  A tag suffix formulates a module-describing tag name.



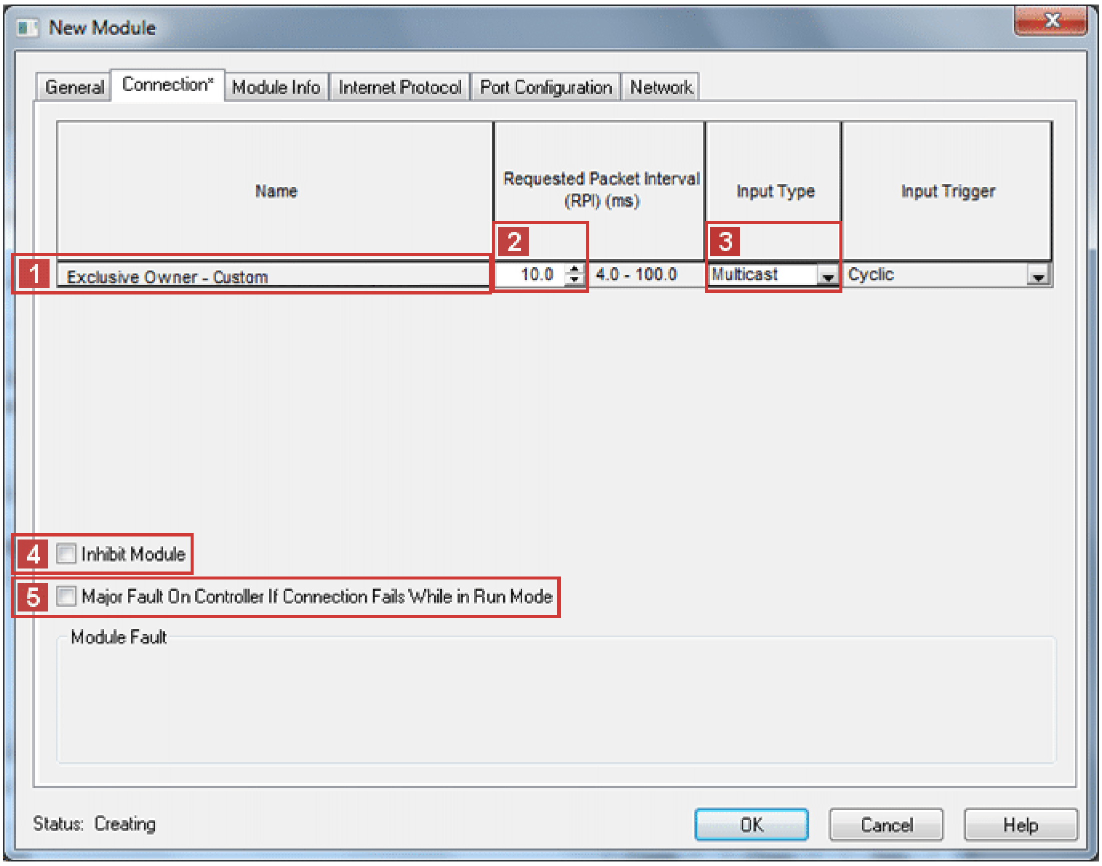
Settings for an "AC Drive Profile" application:

- 1 Select **"AC Drive"** connection.
  In the example, the AC Drive Profile "Exclusive Owner - AC Drive Ext. Speed & Torque" is used.
- 2 Set **Data type = INT**.
  The INT data type is exchanged via the ports MCI_IN and MCI_OUT of the Inverter Drives 8400. In case of the SINT data type, an additional PLC logic is required for conversion.
- 3 Set **Tag Suffix = 1**.
  A tag suffix formulates a module-describing tag name.



9. Complete the settings with **OK**.

10. Open the "New Module" dialog window, select the **Connection** tab and set further properties.



1 **"Name"** displays the name of the connection set under 8..

The example shows an "Exclusive Owner - Custom" connection. According to this, the name of an "AC Drive Profile" connection may be displayed here too.

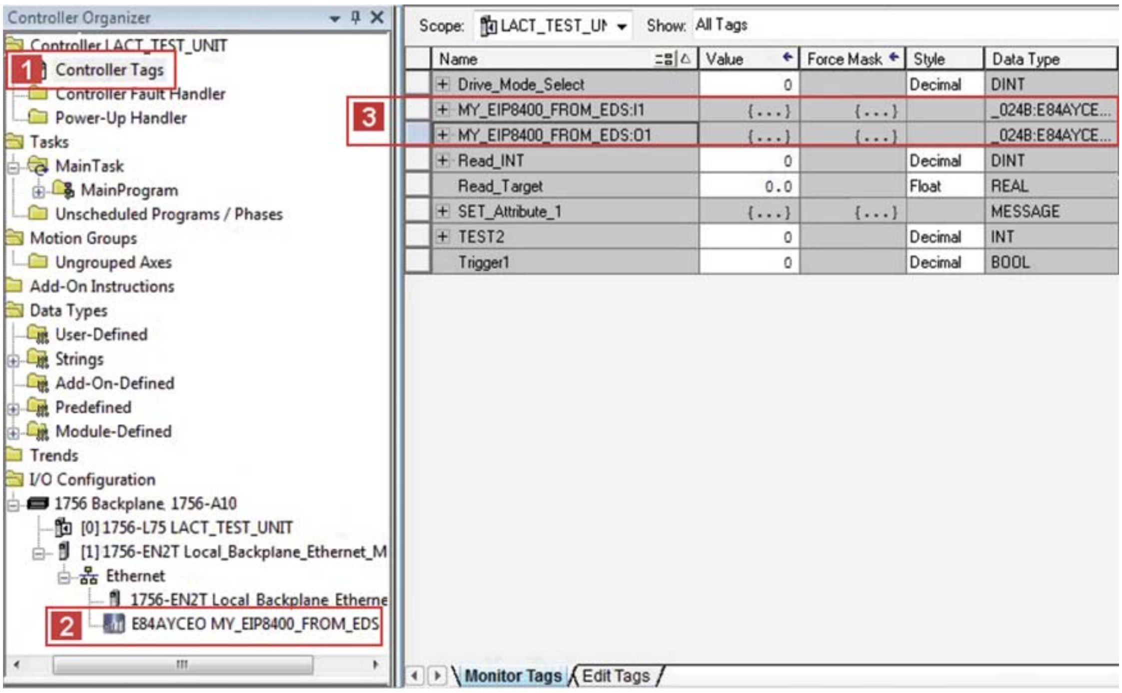
Required settings

Settings		Description
2	**Requested Packet Interval (RPI)**	Set **RPI** ≥ **4.0 ms**. (Standard: 10 ms) The RPI [ms] specifies the intervals for the I/O data exchange between the drive (adapter) and the controller (scanner).
3	**Input type**	Select the **"Multicast"** input type. • The input data is sent from the adapter to the scanner by means of multicast telegrams. • Other scanners can access this data in addition to the currently configured scanner ("Listen only" or "Input only" connections).

Optional settings

Settings		Description
4	**Inhibit module**	This option serves to interrupt or inhibit the communication to the adapter.
5	**Major Fault On Controller If Connection Fails While In Run Mode**	This option serves to put the controller into the error status if the EtherNet/IP connection to the drive fails while the controller is running.

11. Complete the settings with **OK**.
    - The drive is inserted in the configuration tree under the 2 **"I/O Configuration"**.
    - The corresponding tags are created in the 1 **"Controller Tags"**.
    - The example 3 shows the ...
      input assembly tags as "MY_EIP8400_FROM_EDS:I1";
      output assembly tags as "MY_EIP8400_FROM_EDS:O1";



If you click the "+" in front of the assembly tag name, all data contained in the assembly tags are shown below it. You can create "alias tags" to reference to single bits of the assembly tag.

12. Create an "alias tag".

Example with the assembly object instance 23 (0x17):

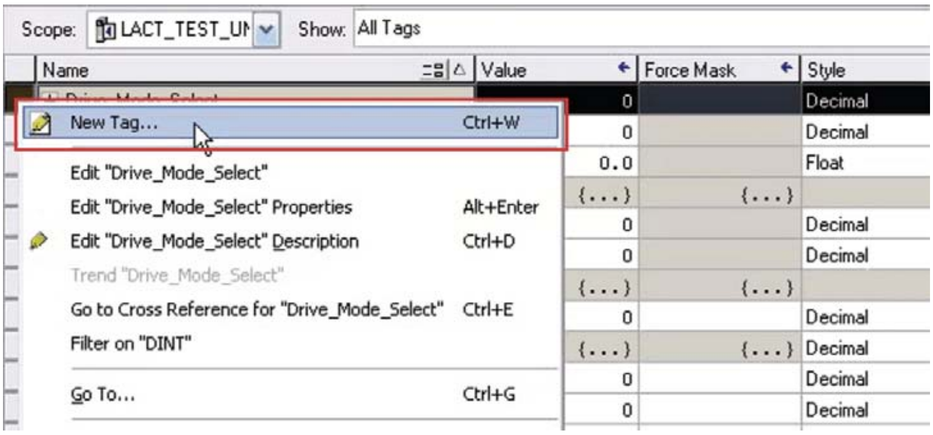
For a forward motion of a conveyor, bit ’0’ (Run Fwd) is to be referenced by byte ’0’.

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
23 (0x17)	**0**		Net Ref	NetCtrl			Fault Reset	Run Rev	**Run Fwd**
	1								
	2	Speed Reference (Low Byte)							
	3	Speed Reference (High Byte)							
	4	Torque Reference (Low Byte)							
	5	Torque Reference (High Byte)							

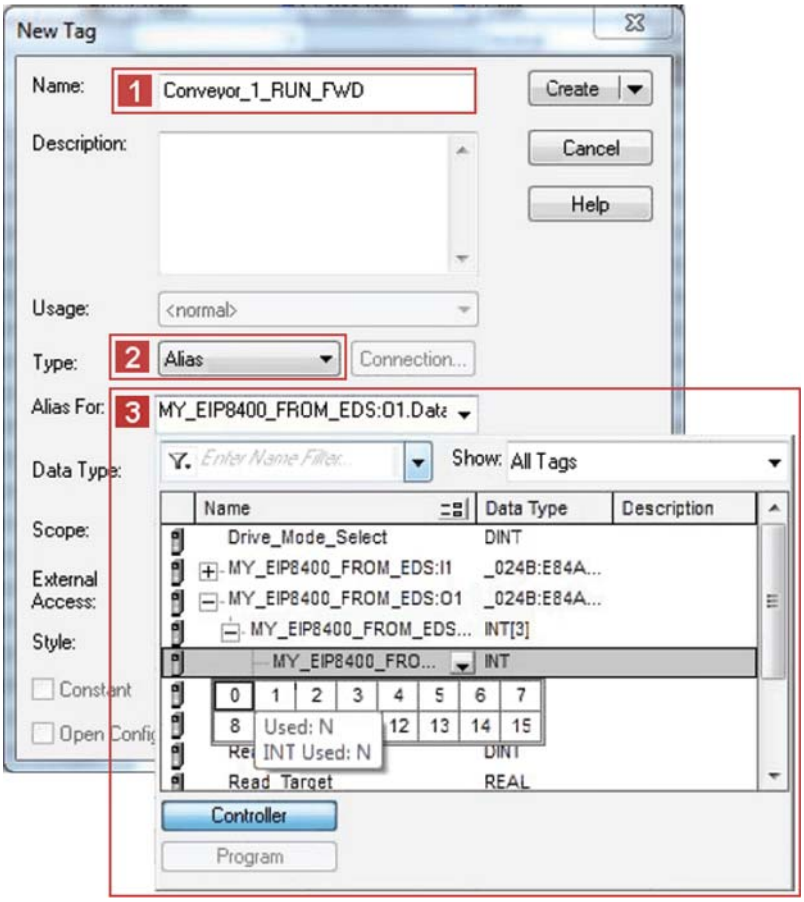
Note!

- **NetCtrl (Bit 5)** and **NetRef (Bit 6)** have to be set in order that the drive can receive start/stop commands and speed/torque command via the network.
- In order to be able to use the torque control of the **assembly object distance 23 (0x17)**, the "DriveMode" attribute has to be written by means of explicit message transmission.
  ▶ Write "DriveMode" attribute (📖 154)

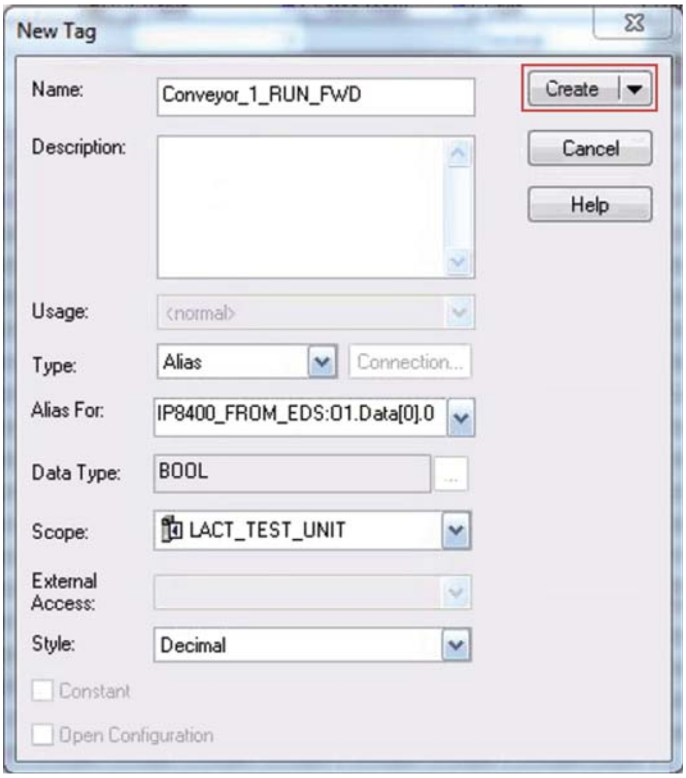
Right-click an assembly tag and select the command **"New Tag..."** in the context menu.



13. Go to the "New Tag" dialog window and ...
    - assign a 1 **Name** for the tag (in the example: "Conveyor_1_RUN_FWD");
    - 2 set **Type = "Alias"**;
    - select the 3 **Alias address** which is to be referenced for the alias tag.
      (in the example: "MY_EIP8400_FROM_EDS:O1.Data(0).0" (bit '0' of byte '0')



14. Confirm the settings with the **Create** button.



The alias tag "Conveyor_1_RUN_FWD" is created below the **"Controller Tags"**:

Name	Value	Force Mask	Style	Data Type
...	...	...	...	...
[–] MY_EIP8400_FROM_EDS:O1	{...}	{...}		_024B:E84AYCE...
[+] MY_EIP8400_FROM_EDS:O1.Data[1]	0		Decimal	SINT
[+] MY_EIP8400_FROM_EDS:O1.Data[2]	0		Decimal	SINT
[+] MY_EIP8400_FROM_EDS:O1.Data[3]	0		Decimal	SINT
[+] MY_EIP8400_FROM_EDS:O1.Data[4]	0		Decimal	SINT
[+] MY_EIP8400_FROM_EDS:O1.Data[5]	0		Decimal	SINT
Conveyor_1_RUN_FWD	**0**		**Decimal**	**BOOL**

15. The last step is Saving the I/O configuration in »RSLogix 5000« (📖 83).

8.7 Saving the I/O configuration in »RSLogix 5000«

After adding the scanner and the adapter to the I/O configuration, the configuration must be downloaded to the controller. The configuration file should also be saved on your computer.

How to save the I/O configuration:

1. Select the menu command **Communications → Download**.
   - The "Download" dialog box will open.
   - If a message box reports that »RSLogix 5000« is unable to go online, select the menu command **Communications → Communications Who Active** and try to find your controller in the "Who Active" dialog box. If the controller is not shown there, the EtherNet/IP driver needs to be added to »RSLinx« or configured in »RSLinx«. For more information, please refer to the »RSLinx« online help.
2. Click the **Download** button.
   - The I/O configuration is downloaded to the controller.
   - When the download has been successfully completed, »RSLogix 5000« changes to online mode and the I/O OK box in the upper left of the screen is green.
3. Select the menu command **File → Save**.
   - If this is the first time the I/O configuration is saved, the "Save As" dialog box will open.
   - Select a folder and enter a file name to save the configuration to a file on your computer.
   - Finally, click the **Save** button.

9 Parameter data transfer (explicit messages)

An "explicit message" is a logical instruction in the PLC program used for messaging. It can be used to read or write to either a parameter setting or the data of an assembly object.

For all Allen-Bradley devices of the CompactLogix, ControlLogix and SoftLogix series, the MSG instruction provides the application possibilities described in this chapter. For other PLC types, please consult the programming documentation for the corresponding PLC.

Note!

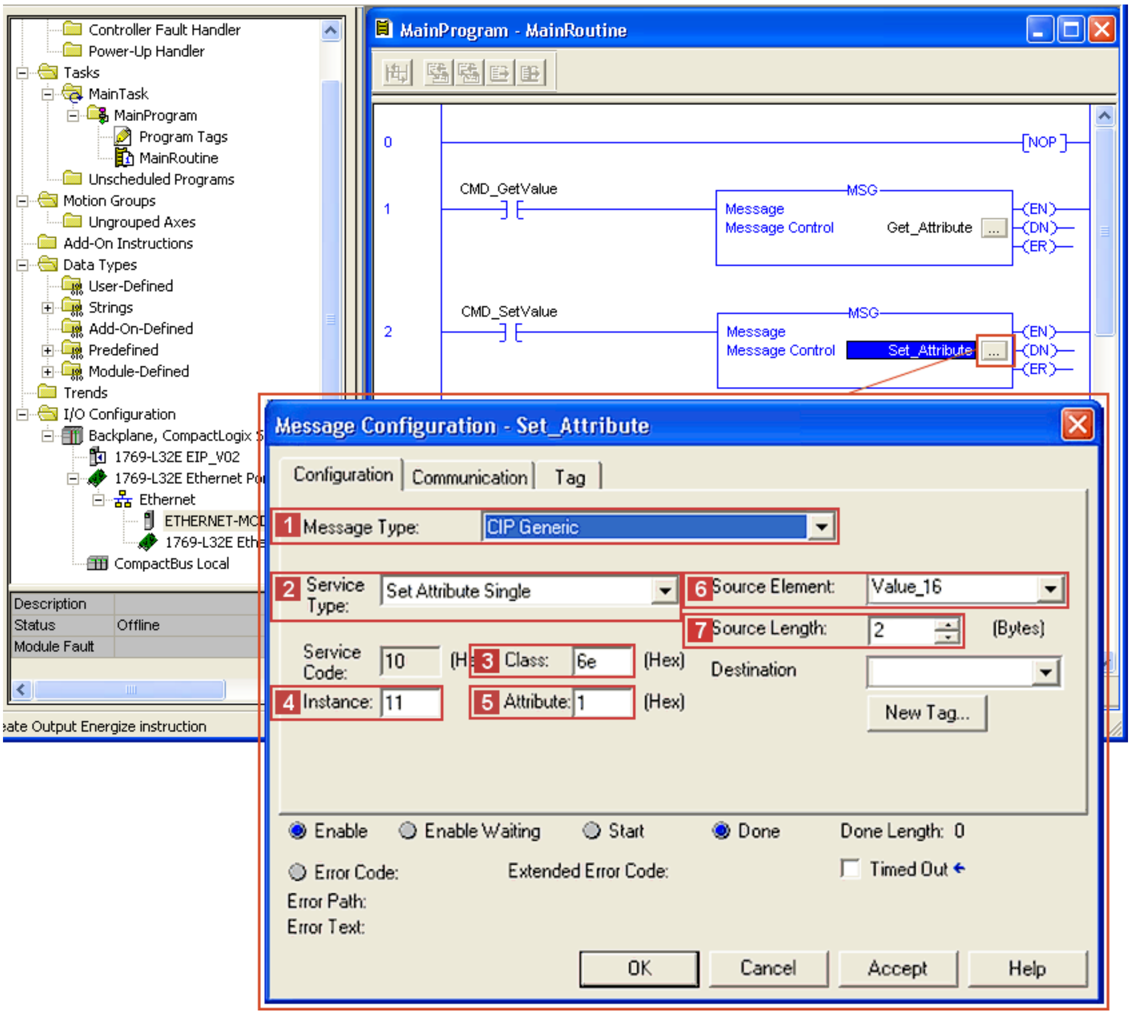
When you use several MSG BLOCKs per adapter, you can work resource-friendly by sequential triggering and hold available enough communication reserves in the EtherNet/IP module for further possible clients.

Application note

An example of parameter data transfer (read/write parameters) in a "AC Drive Profile" application can be found in the download area (Application Knowledge Base) at www.Lenze.com.

9.1 Write parameters

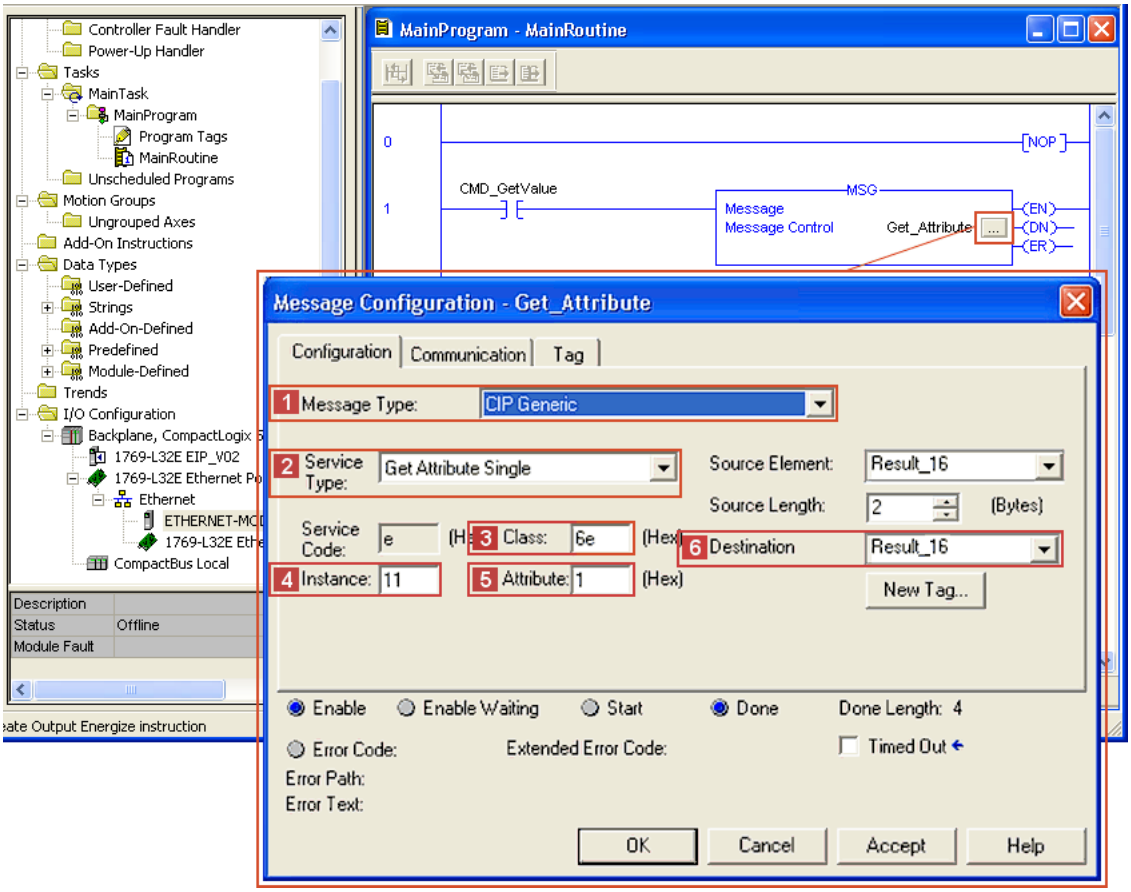
In order to write data into code **C00011** (reference speed) of the Inverter Drive 8400 by means of explicit message transfer, for example, the following settings are necessary:



Settings		Value / description
1	**Message Type**	"CIP Generic"
2	**Service Type**	"Set Attribute Single" (service code "0x10")
3	**Class**	"6E" (access to Lenze code)
4	**Instance**	"11" = Lenze code C00011 of Inverter Drive 8400
5	**Attribute**	"1" = Subcode of the Lenze code • If the corresponding Lenze code does not have a subcode, the value '1' must be entered here. • A display code cannot be configured by the "SET" service.
6	**Source Element**	Variable in the PLC program used as data source for writing.
7	**Source Length**	The source length has to be set to the length (data type) of the current parameter (see parameter reference in the software manual/online help of the inverter). For writing to code C00011, set the source length to "2 bytes".

9.2 Read parameters

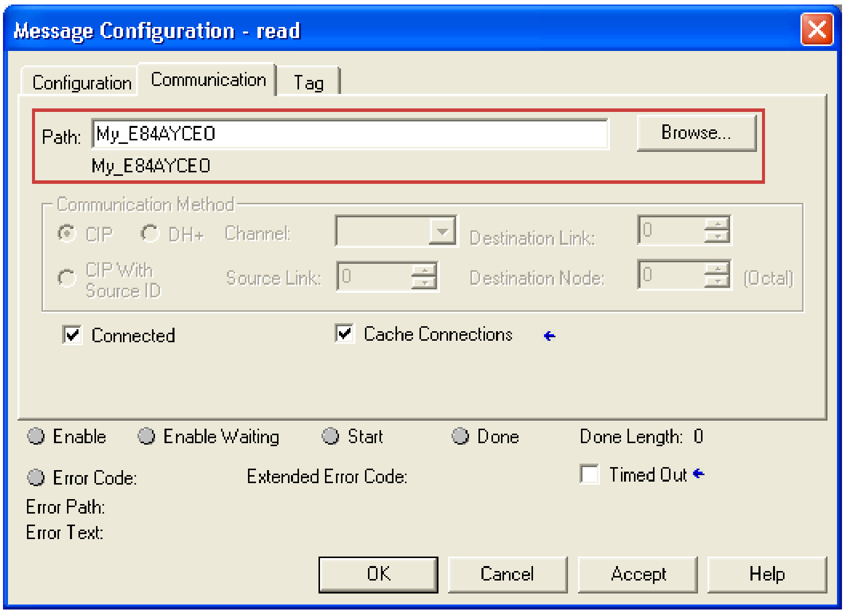
In order to read out Lenze code **C00011** (reference speed) of the Inverter Drive 8400 by means of explicit message transfer, for example, the following settings are necessary:



Settings		Value / description
1	**Message Type**	"CIP Generic"
2	**Service Type**	"Get Attribute Single" (service code "0x0E")
3	**Class**	"6E" (access to to Lenze code)
4	**Instance**	"11" = Lenze code C00011 of Inverter Drive 8400
5	**Attribute**	"1" = Subcode of the Lenze code If the corresponding Lenze code does not have a subcode, the value '1' must be entered here.
6	**Destination**	Variable in the PLC program the drive data will be copied to. When reading code C00011, make sure that the tag used as destination is a single word in UINT16 format.

For every "explicit message", the path for sending the message via the Ethernet port of the PLC to the IP address of the drive needs to be set under the **Communication** tab. This path depends on the PLC used.

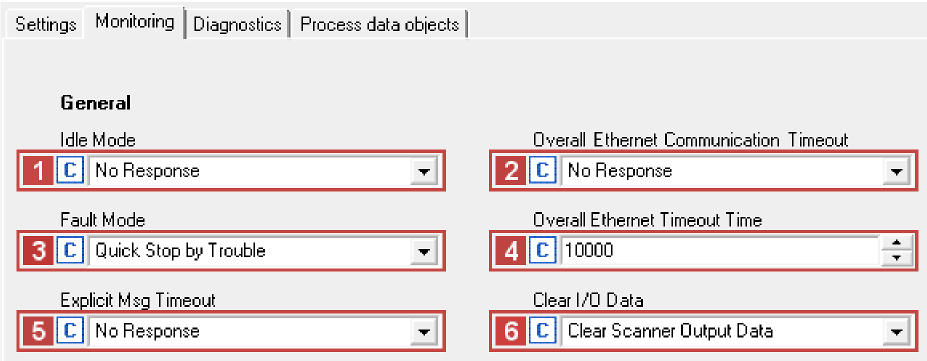
If you need assistance with the setting of this path, please consult the PLC manufacturer.



10 Monitoring

Fault with regard to EtherNet/IP communication

You can set the reactions of the Inverter Drive 8400 to a fault in EtherNet/IP communication in »Engineer« by opening the **Monitoring** tab and making the appropriate settings.



Settings		Description
1	**Idle Mode**	The 32-bit real time header sent by the scanner is evaluated. Run/Idle Flag (Bit 0) = TRUE: • The scanner indicates the validity of the I/O data. Run/Idle Flag (Bit 0) = FALSE: • The I/O data are invalid and the response parameterised here (C13880/1) is executed on the drive. • The I/O data are processed as set in 6 (C13885).
2	**Overall Ethernet Communication Timeout**	If there is no reception of explicit or implicit messages after expiry of the time set in 4 (C13881) or if access via »Engineer« no longer takes place after this time, the reaction parameterised here takes place in the drive (C13880/4). Access to the web server is not monitored.
3	**Fault Mode**	The adapter (communication module) monitors the I/O connection to the scanner. If no "implicit message" has been received within the timeout time for implicit messages parameterised by the scanner, the response parameterised here (C13880/2) is executed on the drive.
4	**Overall Ethernet Timeout Time**	Here, the overall message timeout time (C13881) is set. If no message has been received within this time, the response parameterised in 2 (C13880/4) is executed. The following messages are monitored: • Implicit messages • Explicit messages • »Engineer« access via EtherNet/IP
5	**Explicit Msg Timeout**	If no "explicit message" has been received within the timeout time for explicit messages parameterised by the scanner, the response parameterised here (C13880/3) is executed on the drive.
6	**Clear I/O Data**	(C13885), serves to set the I/O data to be processed by the adapter to maintain internal communication if ... • the CIP network status (C13862) of the controlling I/O connection is not "Connected" or • an idle event has occurred.

11 Diagnostics

The LEDs on the front of the communication module serve to diagnose faults. Moreover, the »Engineer« serves to show diagnostic information.

11.1 LED status displays

Note!

LED status displays for trouble-free operation:

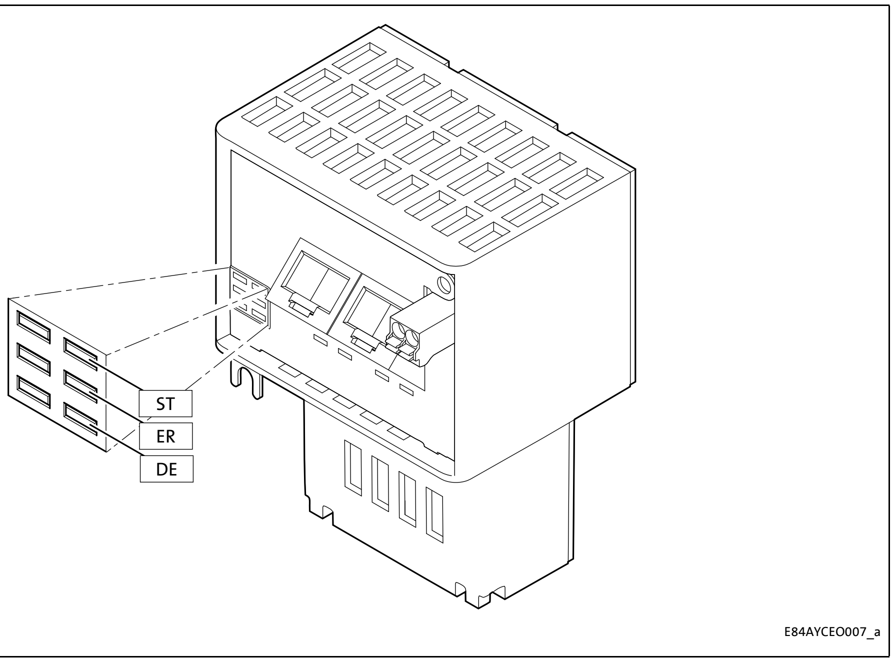
- The LEDs **MS** (📖 90) and **NS** (📖 91) light up permanently.
- On the RJ45 sockets **X259** and **X260**, the green LEDs are lit and the yellow LEDs are blinking or jittering (📖 94).

The following status displays are distinguished:

- Module status displays (📖 90)
- CIP™ status displays (📖 91)
- Status indicators at the RJ45 sockets (X259, X260) (📖 94)

11.1.1 Module status displays

Module states are indicated by the **ST**, **ER** and **DE** LEDs.

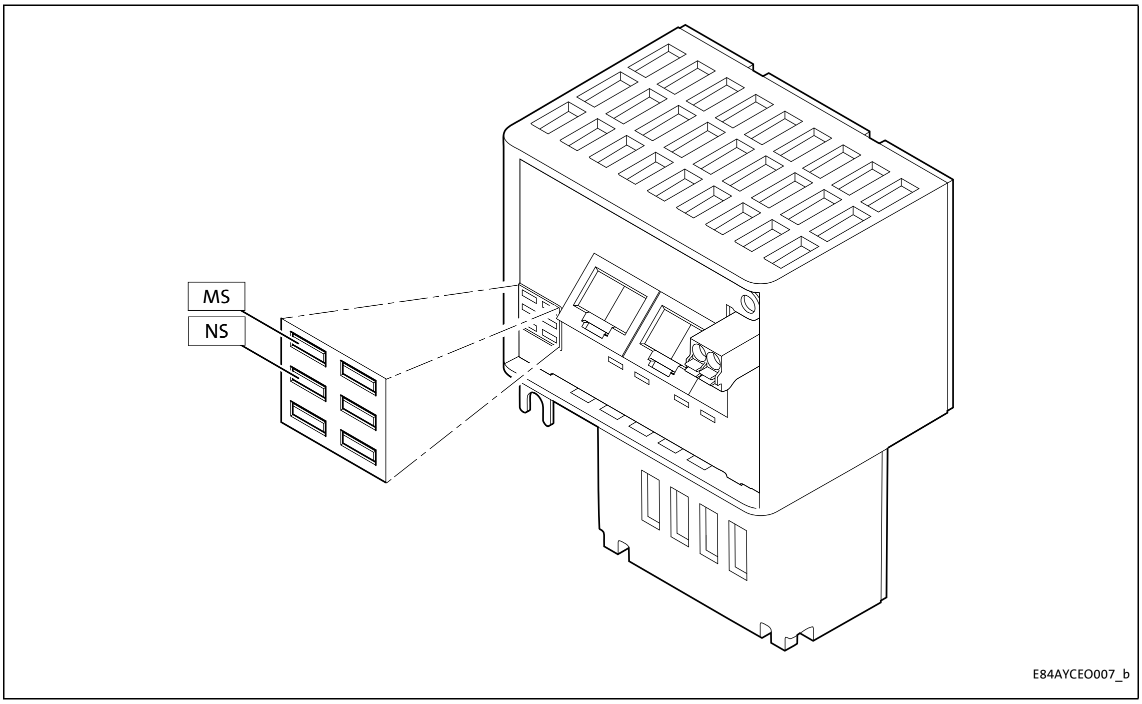


[11-1] LEDs ST, ER, DE

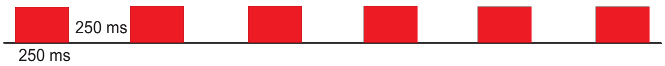
LED	Colour	Status	Description
ST	Green	On	 The communication module is supplied with voltage and is connected to the standard device.
		Blinking	 The communication module is supplied with voltage, but is not connected to the standard device. (Standard device is switched off, in the initialisation phase, or not available.)
ER	Red	On	 An error has occurred in the communication module.
DE	Red	On	 The communication module is not accepted by the basic device or the basic device is not active (see notes in the documentation relating to the basic device.)

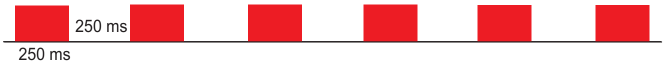
11.1.2 CIP™ status displays

CIP statuses are indicated by the **MS** and **NS** LEDs.



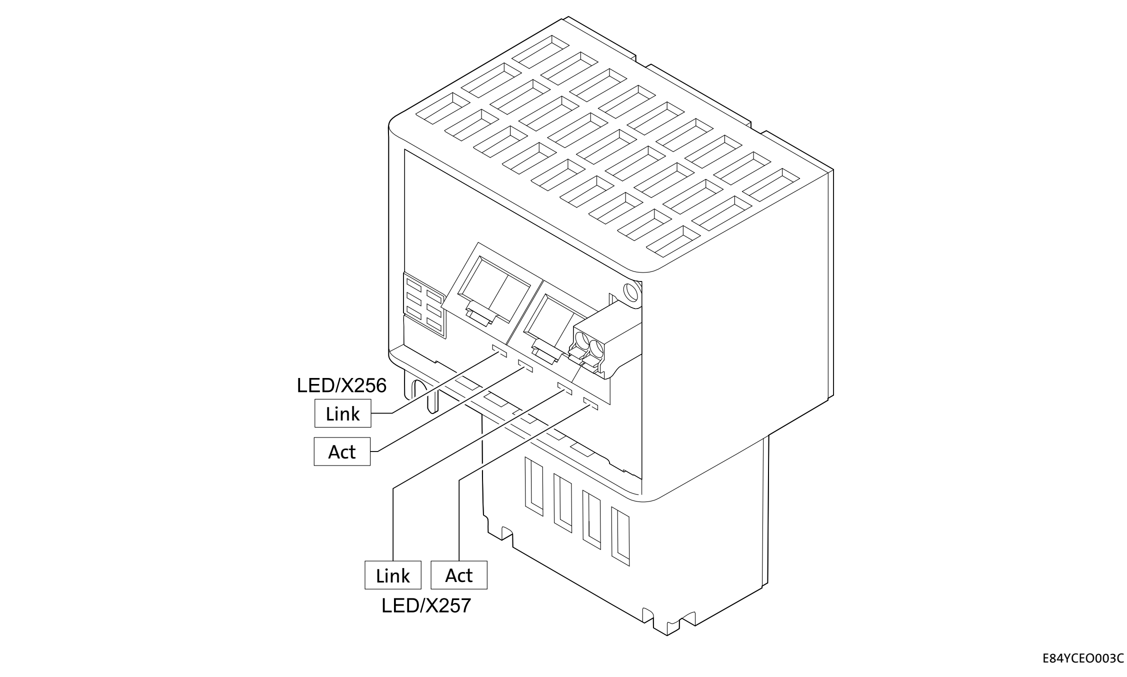
[11-2] LEDs MS, NS

LED	Colour / status		Description
MS	**Green**	**Red**	
	Off	Off	CIP module status: "Nonexistent" The communication module is not being supplied with voltage.
	Off	On	 CIP module status: "Major Unrecoverable Fault" The communication module has a fault that cannot be rectified. The status is set if the pending status determining device error shows the "System fault" response.
	Off	Blinking	 CIP module status: "Major Recoverable Fault" The communication module has a fault that can be rectified. The status is set if the pending status determining device error shows the "Fault", "Trouble", "Quick stop by trouble", "Warning locked", or "Warning" response.
	On	Off	 CIP module status: "Operational" The communication module is working perfectly.
	Blinking	Off	 CIP module status: "Standby" The communication module has not been completely configured or the configuration is defective.
	Blinking	Blinking	 CIP module status: "Device Self Testing" The communication module is currently undergoing a self-test.

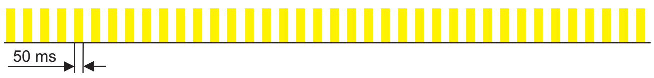
LED	Colour / status		Description
NS	Green	Red	
	Off	Off	CIP network status: "No IP Address" The communication module is not being supplied with voltage or has not been given an IP address.
	Off	On	 CIP network status: "Duplicate IP" The communication module is unable to gain access to the fieldbus (IP address conflict).
	Off	Blinking	 CIP network status: "Connection Timeout" A time-out is executed.
	On	Off	 CIP network status: "Connected" The communication module is working perfectly and has established a connection to the scanner.
	Blinking	Off	 CIP network status: "No Connections" The communication module ... • is working correctly; • has been assigned an IP address; • has not been integrated into the network by the scanner yet.
	Blinking	Blinking	 CIP network status: "Self-Test" The communication module is currently undergoing a self-test.

11.1.3 Status indicators at the RJ45 sockets (X259, X260)

The LEDs at RJ45 sockets **X259** and **X260** indicate the status of the Ethernet connection.

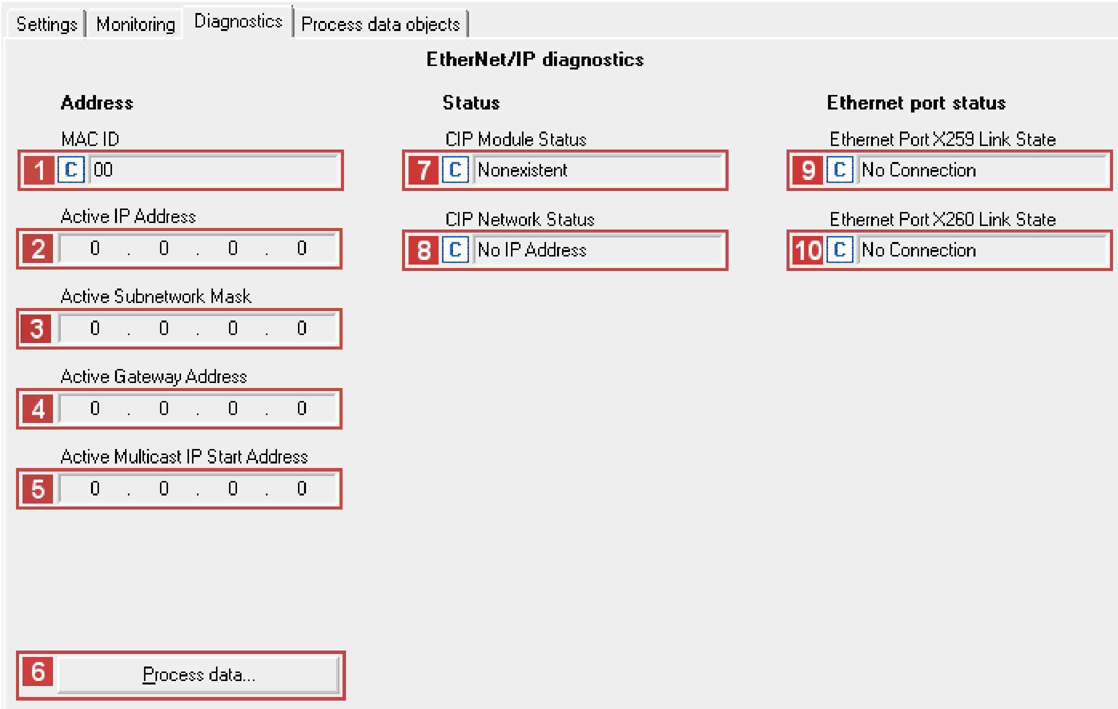


[11-3] LEDs Link , Act

LED	Colour	Status	Description
Link	Green	Off	No Ethernet connection
		On	 There is a physical Ethernet connection.
Act	Yellow	Off	No Ethernet data transfer
		On *or* flickers	 Data are being exchanged via Ethernet.

11.2 Diagnostics with the »Engineer«

In the »Engineer«, the **Diagnostics** tab displays various pieces of EtherNet/IP diagnostic information.



Display		Code
1	**MAC-ID**	C13003
2	**Active IP Address**	C13010
3	**Active Subnetwork Mask**	C13011
4	**Active Gateway Address**	C13012
5	**Active multicast IP address**	C13016
6	**Process data**	C13850, C13851, C13852, C13853
7	**CIP Module Status**	C13861
8	**CIP Network Status**	C13862
9	**Ethernet Port X259 connection status**	C13863/1
10	**Ethernet Port X260 connection status**	C13863/2

12 Error messages

This chapter supplements the error list in the software manual and in the »Engineer« online help system for the Inverter Drive 8400 by adding the EtherNet/IP error messages to the list.

12.1 Short overview of the EtherNet/IP error messages

Software manual/»Engineer« online help for Inverter Drives 8400

Here you will find general information on diagnostics & fault analysis and on error messages.

The following table lists all EtherNet/IP error messages in the numerical order of the error numbers. Furthermore, the preset error response and - if available - the parameter for setting the error response are specified.

Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error no. [hex]	Subject area no. [dec]	Error no. [dec]	Error text	Error type (Error response)	Adjustable in
0x01bc3100	444	12544	EtherNet/IP: Exist. connect. to 8400 lost	0: No Response	-
0x01bc5531	444	21809	EtherNet/IP: Memory: No Access	1: No Response	-
0x01bc5532	444	21810	EtherNet/IP: Memory: Read Error	1: No Response	-
0x01bc5533	444	21811	EtherNet/IP: Memory: Write Error	1: No Response	-
0x01bc6010	444	24592	EtherNet/IP: Restart by Watchdogreset	1: No Response	-
0x01bc6011	444	24593	EtherNet/IP: Internal Error	1: No Response	-
0x01bc6100	444	24832	EtherNet/IP: Internal Error	1: No Response	-
0x01bc6101	444	24833	EtherNet/IP: Internal Error	1: No Response	-
0x01bc641f	444	25631	EtherNet/IP: Invalid Parameter Set	1: No Response	-
0x01bc6420	444	25632	EtherNet/IP: Error: Lenze Setting Loaded	1: No Response	-
0x01bc6430	444	25648	EtherNet/IP: Invalid Configuration	1: No Response	-
0x01bc6533	444	25907	EtherNet/IP: Invalid IP Parameter	1: No Response	-
0x01bc8111	444	33041	EtherNet/IP: Fault Mode	1: No Response	C13880/2
0x01bc8112	444	33042	EtherNet/IP: Explicit Message Timeout	0: No Response	C13880/3
0x01bc8114	444	33044	EtherNet/IP: Overall Ethernet Timeout	0: No Response	C13880/4
0x01bc8121	444	33057	EtherNet/IP: Ethernet cable pulled out	1: No Response	-
0x01bc8132	444	33074	EtherNet/IP: Idle Mode	0: No Response	C13880/1
0x01bc8273	444	33395	EtherNet/IP: Duplicate IP Address	1: No Response	-

12.2 Possible causes and remedies

In this chapter, all EtherNet/IP error messages are listed in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

EtherNet/IP: Exist. connect. to 8400 lost [0x01bc3100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☒ **None** ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• The communication module is being supplied with external voltage but the Inverter Drive 8400 is not being supplied with voltage. • The communication module has not been correctly connected to the Inverter Drive 8400.	• Supply the Inverter Drive 8400 with voltage. • Securely connect the communication module to the Inverter Drive 8400.

EtherNet/IP: Memory: No Access [0x01bc5531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access to memory was not possible.	Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Memory: Read Error [0x01bc5532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameter could not be read.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Memory: Write Error [0x01bc5533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameter could not be written.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Restart by Watchdogreset [0x01bc6010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

EtherNet/IP: Internal Error [0x01bc6011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

EtherNet/IP: Internal Error [0x01bc6100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

EtherNet/IP: Internal Error [0x01bc6101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

EtherNet/IP: Invalid Parameter Set [0x01bc641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Error: Lenze Setting Loaded [0x01bc6420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set was not possible.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Invalid Configuration [0x01bc6430]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module configuration is faulty.	Check and correct module configuration.

EtherNet/IP: Invalid IP parameters [0x01bc6533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
One or more IP parameters are faulty.	Check and correct IP configuration. ▶ Setting the IP configuration of the Inverter Drive 8400 (📖 42)

EtherNet/IP: Fault Mode [0x01bc8111]

Response (Lenze setting printed in bold)	**Setting:** C13880/2
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Connection to scanner has been interrupted. • Controlling I/O connection failed by timeout. • Within the timeout time for implicit messages parameterised by the scanner, no "implicit messages" have been received.	• Check cables and terminals. • Plug in the network cable into the Ethernet port. • Check Requested Package Interval (RPI) of I/O connection. • Increase timeout time for implicit messages.

EtherNet/IP: Explicit Message Timeout [0x01bc8112]

Response (Lenze setting printed in bold)	**Setting:** C13880/3
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Connection to scanner has been interrupted. • Failure of an explicit connection • Within the timeout time for explicit messages parameterised by the scanner, no "explicit messages" have been received.	• Check cables and terminals. • Plug in the network cable into the Ethernet port. • Check Requested Package Interval (RPI) of the explicit connection. • Increase timeout time for explicit messages.

EtherNet/IP: Overall Ethernet Timeout [0x01bc8114]

Response (Lenze setting printed in bold)	**Setting:** C13880/4
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Failure of »Engineer« communication via Ethernet • When the time set in C13881 has expired, there is no access via the »Engineer«.	• Check cables and terminals. • Plug in the network cable into the Ethernet port. • Increase the overall Ethernet timeout time in C13881. ▶ Fault with regard to EtherNet/IP communication (📖 88)

EtherNet/IP: Ethernet cable pulled out [0x01bc8121]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ system fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Network cable has been detached from the Ethernet terminal. • Network cable (plug) is defective.	• Plug the network cable into the Ethernet terminal. • Check the network cable (plug) and replace it if necessary.

EtherNet/IP: Idle Mode [0x01bc8132]

Response (Lenze setting printed in bold)	**Setting:** C13880/1
☑ None ☐ system fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Scanner has received an idle event. • The scanner is in the "PROG" mode. • In the "Scanner Command Register", the run/idle flag (bit 0) is 0.	Set the scanner to the run mode. Run/idle flag (bit 0) = 1

EtherNet/IP: Duplicate IP Address [0x01bc8273]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ system fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
An IP address has been allocated twice within the network. The addresses of the network nodes must differ from each other.	Correct the IP address (C13000). ▶ Setting the IP configuration of the Inverter Drive 8400 (📖 42)

12.3 CIP™ error messages

Error code [hex]	Error designation	Description
0x000	SUCCESS	No error
0x001	...	Instance error messages (📖 136) des Connection Manager Object (6 / 0x06) (📖 135)
0x002	RESOURCE_UNAVAILABLE	Resource required to perform the service not available.
0x003	INVALID_PARAM_VALUE	Invalid parameter value
0x008	SERVICE_NOT_SUPP	Service is not supported.
0x009	INVALID_ATTRIB_VALUE	Invalid attribute.
0x00B	ALREADY_IN_STATE	The object is already in the required state.
0x00C	OBJ_STATE_CONFLICT	The object cannot perform the service.
0x00E	ATTR_NOT_SETTABLE	The attribute is write-protected.
0x00F	PRIVILEGE_VIOLATION	Access denied.
0x010	DEVICE_STATE_CONFLICT	The current state of the device prohibits performing the requested service.
0x011	REPLY_DATA_TOO_LARGE	The response data are longer than the response buffer
0x013	NOT_ENOUGH_DATA	The data length is too short.
0x014	ATTRIBUTE_NOT_SUPP	The attribute is not supported.
0x015	TOO_MUCH_DATA	The data length is too long.
0x016	OBJECT_DOES_NOT_EXIST	The object is not supported by the adapter.
0x017	FRAGMENTATION	The fragmentation for the requested service is currently not activated.
0x020	INVALID_PARAMETER	Invalid parameter

12.4 Mapping of Lenze device errors to DRIVECOM errors

Via the instance attribute "FaultCode" of the Control Supervisor Object (41 / 0x29) (📖 150) Lenze device errors are output with the DRIVECOM error numbers.

From inverter drive version V13.00 and communication module version V01.02 onwards, "CAN Emergency Error Codes" are output as DRIVECOM error numbers!

The following tables show the assignment of the Lenze device errors and "CAN Emergency Error Codes" to the DRIVECOM errors.

Software manual/»Engineer« online help for Inverter Drive 8400

Here you will find detailed information about the Lenze error messages listed in the following table.

Lenze error		CAN	DRIVECOM error	
Error number [32 bits]	Error message	Emergency Error Code	Error number [hex]	Error message
xx.0111.00002	Su02: One mains phase is missing	0x3000	0x3000	Voltage
xx.0111.00003	Su03: Too frequent mains switching	0x3000	0x3000	Voltage
xx.0111.00004	Su04: CU insufficiently supplied	0x3000	0x3000	Voltage
xx.0111.00006	Su06: Power input overload	0x3000	0x3000	Voltage
xx.0119.00000	OH4: Heat sink temp. > Switch-off temp. -5°C	0x4000	0x4000	Temperature
xx.0119.00001	OH1: Heatsink overtemperature	0x4000	0x4000	Temperature
xx.0119.00002	OH7: Motor temperature resolver > C121	0x4000	0x4000	Temperature
xx.0119.00003	OH9: Motor overtemperature resolver	0x4000	0x4000	Temperature
xx.0119.00012	Sd6: Error temperature sensor resolver	0x7300	0x7300	Sensor
xx.0119.00015	OH3: Motor temperature (X106) triggered	0x4000	0x4000	Temperature
xx.0119.00020	OH6: Motor temperature MultiEncoder > C121	0x4000	0x4000	Temperature
xx.0119.00021	OH12: Motor overtemperature MultiEncoder	0x4000	0x4000	Temperature
xx.0119.00022	Sd12: Error temperature sensor MultiEncoder	0x7300	0x7300	Sensor
xx.0119.00050	OC5: Ixt overload	0x2000	0x2000	Current
xx.0123.00001	OT1: Maximal torque reached	0x8300	0x8302	Torque limiting
xx.0123.00007	OC7: Motor overcurrent	0x2000	0x2000	Current
xx.0123.00014	OU: DC bus overvoltage	0x3100	0x3110	Mains overvoltage
xx.0123.00015	LU: DC bus undervoltage	0x3100	0x3120	Mains undervoltage
xx.0123.00016	OC1: Power section - short circuit	0x2000	0x2130	Short Circuit
xx.0123.00017	OC2: Power section - earth fault	0x2000	0x2120	Short to Earth
xx.0123.00024	Sd2: Wire breakage resolver	0x7300	0x7303	Resolver 1 defective
xx.0123.00026	Sd7: Error encoder communication	0x7300	0x7305	Incremental encoder 1 defective
xx.0123.00027	Sd4: Wire breakage MultiEncoder	0x7300	0x7300	Sensor
xx.0123.00030	OC10: Maximum current reached	0x2000	0x2000	Current
xx.0123.00031	OC17: Clamp sets pulse inhibit	0xF000	0xF000	Additional functions
xx.0123.00032	OS1: Maximum speed limit reached	0x8400	0x8402	Velocity Limiting
xx.0123.00033	OS2: Max. motor speed	0x8400	0x8400	Speed Controller
xx.0123.00056	ID2: Motor data identification error	0xF000	0xF000	Additional functions
xx.0123.00057	ID1: Motor data identification error	0xF000	0xF000	Additional functions
xx.0123.00058	ID3: CINH motor data identification	0xF000	0xF000	Additional functions
xx.0123.00059	ID4: Error resistor identification	0xF000	0xF000	Additional functions
xx.0123.00060	ID7: Motor control does not match motor data	0xF000	0xF000	Additional functions
xx.0123.00062	Sd8: Encoder angular drift monitoring	0x7300	0x7300	Sensor
xx.0123.00065	OC12: I2xt overload - brake resistor	0xF000	0x7110	Brake Chopper

Lenze error		CAN	DRIVECOM error	
Error number [32 bits]	Error message	Emergency Error Code	Error number [hex]	Error message
xx.0123.00071	OC11: Current clamp for too long (>1 sec)	0xF000	0xF000	Additional functions
xx.0123.00074	ID5: Error pole position identification	0xF000	0xF000	Additional functions
xx.0123.00075	ID6: Error resolver ident.	0xF000	0xF000	Additional functions
xx.0123.00090	OC13: Maximum current for Fch exceeded	0x2000	0x2000	Current
xx.0123.00093	OT2: Speed controller output is limited	0xF000	0x7310	Speed
xx.0123.00094	FC01: Switching frequency reduction	0x2000	0xF000	Additional functions
xx.0123.00095	FC02: Maximum speed for Fchop	0xF000	0xF000	Additional functions
xx.0123.00096	OC14: Limitation direct-axis current controller	0xF000	0xF000	Additional functions
xx.0123.00097	OC15: Limitation cross current controller	0xF000	0xF000	Additional functions
xx.0123.00098	OC16: Limitation torque controller	0xF000	0xF000	Additional functions
xx.0123.00099	FC03: Limitation field controller	0xF000	0xF000	Additional functions
xx.0123.00105	OC6: I2xt overload - motor	0x2000	0x7120	Motor
xx.0123.00145	LP1: Motor phase failure	0x3000	0x3130	Phase Failure
xx.0123.00200	SD10: Speed limit - feedback system 12	0x7300	0x7300	Sensor
xx.0123.00201	SD11: Speed limit - feedback system 67	0x7300	0x7300	Sensor
xx.0123.00205	SD3: Open circuit - feedback system	0x7300	0x7301	Tacho defective
xx.0125.00001	An01: AIN1_I < 4 mA	0xF000	0xF000	Additional functions
xx.0125.00002	An02: AIN2_I < 4 mA	0xF000	0xF000	Additional functions
xx.0126.00001	Ab01: Axis bus timeout	0x8000	0x8000	Monitoring
xx.0126.00002	Ab02: Axis bus IO error	0x8100	0x8100	Communication
xx.0127.00002	CE04: MCI communication error	0x7000	0x7500	Communication
xx.0127.00015	CE0F: MCI control word	0xF000	0xF000	Additional functions
xx.0131.00000	CE4: CAN Bus Off	0x8000	0x8000	Monitoring
xx.0131.00006	CA06: CAN CRC error	0x8000	0x8000	Monitoring
xx.0131.00007	CA07: CAN Bus Warn	0x8000	0x8000	Monitoring
xx.0131.00008	CA08: CAN Bus Stopped	0x8000	0x8000	Monitoring
xx.0131.00011	CA0b: CAN HeartBeatEvent	0x8130	0x8000	Monitoring
xx.0131.00015	CA0F: CAN control word	0xF000	0x8000	Monitoring
xx.0135.00001	CE1: CAN RPDO1	0x8100	0x8100	Communication
xx.0135.00002	CE2: CAN RPDO2	0x8100	0x8100	Communication
xx.0135.00003	CE3: CAN RPDO3	0x8100	0x8100	Communication
xx.0135.00004	CP04: CAN RPDO4	0x8100	0x8100	Communication
xx.0140.00013	CI01: Module missing/incompatible	0x7000	0x7000	Additional Modules
xx.0144.00001	PS01: No memory module	0x6300	0x6300	Date Set
xx.0144.00002	PS02: Par. set invalid	0x6300	0x6300	Date Set
xx.0144.00003	PS03: Par. set device invalid	0x6300	0x6300	Date Set
xx.0144.00004	PS04: Par. set device incompatible	0x6300	0x6300	Date Set
xx.0144.00007	PS07: Par. mem. module invalid	0x6300	0x6300	Date Set
xx.0144.00008	PS08: Par. device invalid	0x6300	0x6300	Date Set
xx.0144.00009	PS09: Par. format invalid	0x6300	0x6300	Date Set
xx.0144.00010	PS10: Memory module link invalid	0x5000	0x5000	Device Hardware
xx.0144.00031	PS31: Ident. error	0x6300	0x6300	Date Set
xx.0145.00014	dF14: SW-HW invalid	0x5530	0x6000	Device software
xx.0145.00015	dF15: DCCOM CU2 error	0x6100	0x6100	Internal software
xx.0145.00024	dF18: BU RCOM error	0x6100	0x6100	Internal software
xx.0145.00025	dF25: CU RCOM error	0x6100	0x6100	Internal software
xx.0145.00026	dF26: Appl. watchdog	0x6200	0x6010	Software reset (watchdog)
xx.0145.00033	dF21: BU watchdog	0x6100	0x6010	Software reset (watchdog)
xx.0145.00034	dF22: CU Watchdog	0x6100	0x6010	Software reset (watchdog)
xx.0145.00035	dF10: AutoTrip reset	0xF000	0xF000	Additional functions

Lenze error		CAN	DRIVECOM error	
Error number [32 bits]	Error message	Emergency Error Code	Error number [hex]	Error message
xx.0145.00050	dF50: Retain error	0x6100	0x6100	Internal software
xx.0145.00051	dF51: CuCcr error	0x6100	0x6100	Internal software
xx.0145.00052	dF52: BuCcr error	0x6100	0x6100	Internal software
xx.0184.00001	Ck01: Pos. HW limit switch	0x8600	0x8600	Positioning controller
xx.0184.00002	Ck02: Neg. HW limit switch	0x8600	0x8600	Positioning controller
xx.0184.00005	Ck15: Error message sig. brake	0x8600	0x8600	Positioning controller
xx.0184.00007	Ck03: Pos. SW limit position	0x8600	0x8600	Positioning controller
xx.0184.00008	Ck04: Neg. SW limit position	0x8600	0x8600	Positioning controller
xx.0184.00015	Ck14: Target position outside SW limit position	0x8600	0x8600	Positioning controller
xx.0184.00064	Ck16: Time overrun manual operation	0x8600	0x8600	Positioning controller
xx.0184.00153	Ck05: Error following error 1	0x8611	0x8611	Following error
xx.0184.00154	Ck06: Error following error 2	0x8611	0x8611	Following error
xx.0184.00155	Ck07: Traversing range limit exceeded	0x8612	0x8612	Reference limit
xx.0184.00156	Ck08: Reference position unknown	0x8612	0x8612	Reference limit
xx.0184.08005	Ck09: Positioning mode invalid	0x8600	0x8600	Positioning controller
xx.0184.08007	Ck10: Profile data implausible	0x8600	0x8600	Positioning controller
xx.0184.08009	Ck11: Operating mode invalid	0x8600	0x8600	Positioning controller
xx.0184.08014	Ck12: Profile number invalid	0x8600	0x8600	Positioning controller
xx.0184.08015	Ck13: Error FB MCKCtrlInterface	0x8600	0x8600	Positioning controller
xx.0400.00009	dH09: EEPROM power unit	0x5530	0x7600	Data memory
xx.0400.00016	dH10: Fan failure	0x5000	0x5000	Device Hardware
xx.0400.00104	dH68: Adjustment data error CU	0x5530	0x6000	Device software
xx.0400.00105	dH69: Adjustment data error BU	0x5530	0x6000	Device software
xx.0980.00001	User error 1	0x6200	0x6200	User software
xx.0981.00002	User error 2	0x6200	0x6200	User software
xx.0982.00003	User error 3	0x6200	0x6200	User software
xx.0983.00004	User error 4	0x6200	0x6200	User software
xx.0984.00001	User error 5	0x6200	0x6200	User software
xx.0985.00002	User error 6	0x6200	0x6200	User software
xx.0986.00003	User error 7	0x6200	0x6200	User software
xx.0987.00004	User error 8	0x6200	0x6200	User software

EtherNet/IP error messages

Lenze error		DRIVECOM error	
Error number [32 bits]	Error message	Error number [hex]	Error message
xx.0444.12544	EtherNet/IP: Exist. connect. to 8400 lost	0x7510	Serial Interface No 1
xx.0444.21809	EtherNet/IP: Memory: No Access	0x7600	Data memory
xx.0444.21810	EtherNet/IP: Memory: Read Error	0x7600	Data memory
xx.0444.21811	EtherNet/IP: Memory: Write Error	0x7600	Data memory
xx.0444.24592	EtherNet/IP: Restart by Watchdogreset	0x6010	Software reset (watchdog)
xx.0444.24593	EtherNet/IP: Internal Error	0x6100	Internal software
xx.0444.24832	EtherNet/IP: Internal Error	0x6100	Internal software
xx.0444.24833	EtherNet/IP: Internal Error	0x6100	Internal software
xx.0444.25631	EtherNet/IP: Invalid Parameter Set	0x7421	Invalid Parameters
xx.0444.25632	EtherNet/IP: Error: Lenze Setting Loaded	0x7421	Invalid Parameters
xx.0444.25648	EtherNet/IP: Invalid Configuration	0x7421	Invalid Parameters
xx.0444.25907	EtherNet/IP: Invalid IP Parameter	0x7421	Invalid Parameters
xx.0444.33041	EtherNet/IP: Fault Mode	0x7000	Additional Modules
xx.0444.33042	EtherNet/IP: Explicit Message Timeout	0x7500	Communication
xx.0444.33044	EtherNet/IP: Overall Ethernet Timeout	0x7500	Communication
xx.0444.33057	EtherNet/IP: Ethernet cable pulled out	0x9000	External malfunction
xx.0444.33074	EtherNet/IP: Idle Mode	0x7000	Additional Modules
xx.0444.33395	EtherNet/IP: Duplicate IP Address	0x7421	Invalid Parameters

13 Parameter reference

This chapter supplements the parameter list and the table of attributes for the Inverter Drive 8400 contained in the software manual and in the »Engineer« online help by the parameters of the communication module E84AYCEO (EtherNet/IP).

13.1 Parameters of the communication module

Software manual/»Engineer« online help for Inverter Drives 8400
Here you will find general information on parameters.

This chapter lists the parameters of the E84AYCEO communication module (EtherNet/IP) in numerically ascending order.

C13000

Parameter | Name: **C13000 | IP Address**			Data type: UNSIGNED_8 Index: 11575 = 0x2D37
Setting of the IP address ▸ Setting the IP configuration of the Inverter Drive 8400 (📖 42)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13000/1	192		IP address (most significant byte)
C13000/2	168		IP address
C13000/3	124		IP address
C13000/4	16		IP address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13001

Parameter | Name: **C13001 | Subnetwork Mask**			Data type: UNSIGNED_8 Index: 11574 = 0x2D36
Setting of the subnet mask ▸ Setting the IP configuration of the Inverter Drive 8400 (📖 42)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13001/1	255		Subnet mask (most significant byte)
C13001/2	255		Subnet mask
C13001/3	255		Subnet mask
C13001/4	0		Subnet mask (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13002

Parameter | Name: **C13002 | Gateway Address**			Data type: UNSIGNED_8 Index: 11573 = 0x2D35
Setting of the gateway address ▸ Setting the IP configuration of the Inverter Drive 8400 (📖 42)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13002/1	0		Gateway address (most significant byte)
C13002/2	0		Gateway address
C13002/3	0		Gateway address
C13002/4	0		Gateway address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13003

Parameter | Name: **C13003 | MAC ID**	Data type: OCTET_STRING Index: 11572 = 0x2D34
Display of the MAC-ID	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13005

Parameter | Name: **C13005 | IP Config Control**			Data type: UNSIGNED_8 Index: 11570 = 0x2D32
Selection how the IP configuration is to be effected. (Instance attribute 3 (Configuration Control) in the TCP/IP Interface Object (245 / 0xF5) (📖 141)) ▸ Setting the IP configuration of the Inverter Drive 8400 (📖 42)			
Selection list (Lenze setting printed in bold)		**Info**	
0	Use stored IP	The IP configuration currently saved in the communication module is used.	
1	**Use BOOTP**	The IP configuration is assigned by the scanner using BOOTP.	
2	Use DHCP	The IP configuration is assigned by the scanner using DHCP. The assignment of a gateway address which is not in the same subnetwork than the IP address, is rejected.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13006

Parameter | Name: **C13006 | Multicast IP Start Address**			Data type: UNSIGNED_8 Index: 11569 = 0x2D31
Setting of the multicast IP address ▶ Setting the IP configuration of the Inverter Drive 8400 (📖 42)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13006/1	239		Multicast IP start address (most significant byte)
C13006/2	64		Multicast IP start address
C13006/3	2		Multicast IP start address
C13006/4	224		Multicast start address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13010

Parameter | Name: **C13010 | Active IP Address**			Data type: UNSIGNED_8 Index: 11565 = 0x2D2D
Display of the active IP address (Instance attribute 5 (IP Address) in the TCP/IP Interface Object (245 / 0xF5) (📖 141))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13010/1			Active IP address (most significant byte)
C13010/2			Active IP Address
C13010/3			Active IP Address
C13010/4			Active IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13011

Parameter | Name: **C13011 | Active Subnetwork Mask**			Data type: UNSIGNED_8 Index: 11564 = 0x2D2C
Display of the active subnetwork mask (Instance attribute 5 (IP Network Mask) in the TCP/IP Interface Object (245 / 0xF5) (📖 141))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13011/1			Active subnet mask (most significant byte)
C13011/2			Active Subnetwork Mask
C13011/3			Active Subnetwork Mask
C13011/4			Active subnet mask (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13012

Parameter | Name: **C13012 | Active Gateway Address**		Data type: UNSIGNED_8 Index: 11563 = 0x2D2B
Display of the active gateway address (Instance attribute 5 (Gateway Address) in the TCP/IP Interface Object (245 / 0xF5) (📖 141))		
Display range (min. value | unit | max. value)		
0 | | 255		
Subcodes	**Info**	
C13012/1	Active gateway address (most significant byte)	
C13012/2	Active gateway address	
C13012/3	Active gateway address	
C13012/4	Active gateway address (least significant byte)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13016

Parameter | Name: **C13016 | Active Multicast IP Address**		Data type: UNSIGNED_8 Index: 11559 = 0x2D27
Display of the active multicast IP address		
Display range (min. value | unit | max. value)		
0 | | 255		
Subcodes	**Info**	
C13016/1	Multicast IP address (most significant byte)	
C13016/2	Multicast IP address	
C13016/3	Multicast IP address	
C13016/4	Multicast IP address (least significant byte)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

_ _

C13017

Parameter | Name: **C13017 | Ethernet Config Control**		Data type: UNSIGNED_16 Index: 11558 = 0x2D26
Setting of the baud rate for the Ethernet connections Dependent on the configuration of the Ethernet port of the device to be connected, we recommend the use of a cross-over cable. ▶ EtherNet/IP connection (📖 29)		
Selection list		
0	Auto-Negotiation	
1	10 Mbps	
2	100 Mbps	
3	Reserved	
4	Reserved	
5	10 Mbps/Half Duplex	
6	10 Mbps/Full Duplex	
7	100 Mbps/Half Duplex	
8	100 Mbps/Full Duplex	
9	Reserved	
10	Reserved	
11	Reserved	
12	Reserved	
Subcodes	**Lenze setting**	**Info**
C13017/1	0: Auto-Negotiation	Ethernet setting port X259
C13017/2	0: Auto-Negotiation	Ethernet setting port X260
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13018

Parameter | Name: **C13018 | Multicast Config Alloc Control**		Data type: UNSIGNED_8 Index: 11557 = 0x2D25
Selection for multicast IP addressing via instance attribute 9 (Mcast Config) in the TCP/IP Interface Object (245 / 0xF5) (📖 141)		
Selection list (Lenze setting printed in bold)		
0	**Default Allocation Algorithm**	
1	Multicast IP Start Address	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13019

Parameter | Name: **C13019 | Multicast Config TTL Value**			Data type: UNSIGNED_8 Index: 11556 = 0x2D24
Setting of the multicast TTL value for the validity time of data packets in the EtherNet/IP network (Instance attribute 8 (TTL Value) in the TCP/IP Interface Object (245 / 0xF5) (📖 141))			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		255	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13020

Parameter | Name: **C13020 | Multicast Config Num Mcast**			Data type: UNSIGNED_8 Index: 11555 = 0x2D23
Used to set how many multicast IP addresses will be assigned. (Instance attribute 9 (Num Mcast) in the TCP/IP Interface Object (245 / 0xF5) (📖 141))			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		8	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13021

Parameter | Name: **C13021 | Quality of Service (VLAN-Tagging)**		Data type: UNSIGNED_8 Index: 11554 = 0x2D22
Used to set whether QoS tags will be used for the prioritisation of the data packets to be transferred. (Instance attribute 1 (802.1Q Tag Enable) in the Quality of Service (QoS) Object (72 / 0x48) (📖 139))		
Selection list (Lenze setting printed in bold)		
0	**802.1Q Tag Disable**	
1	802.1Q Tag Enable	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13022

Parameter | Name: **C13022 | Quality of Service (DSCP)**			Data type: UNSIGNED_8 Index: 11553 = 0x2D21
Setting for the prioritisation of the data packets to be transferred using Differentiated Services Codepoints (DSCP)			
Setting range (min. value | unit | max. value)			
0		63	
Subcodes	**Lenze setting**		**Info**
C13022/1	59		Reserved
C13022/2	47		Reserved
C13022/3	55		Reserved
C13022/4	47		QoS DSCP Scheduled (Instance attribute 5 (DSCP Scheduled) in the Quality of Service (QoS) Object (72 / 0x48) (📖 139))
C13022/5	43		QoS DSCP High Prio (Instance attribute 6 (DSCP High Prio) in the Quality of Service (QoS) Object (72 / 0x48) (📖 139))
C13022/6	31		Reserved
C13022/7	27		QoS DSCP Explicit Msg (Instance attribute 8 (DSCP Explicit Msg.) in the Quality of Service (QoS) Object (72 / 0x48) (📖 139))
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13840

Parameter | Name: **C13840 | DLR Network Topology**		Data type: UNSIGNED_8 Index: 10735 = 0x29EF
Display of the used DLR network topology (Device Level Ring) (Instance attribute 1 (Network Topology) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 137))		
Selection list (read only)		
0	Linear	
1	Ring	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13841

Parameter | Name: **C13841 | DLR Network Status**		Data type: UNSIGNED_8 Index: 10734 = 0x29EE
Display of the DLR network status (Device Level Ring) (Instance attribute 2 (Network Status) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 137))		
Selection list (read only)		
0	Normal	
1	Ring Fault	
2	Unexpected Loop detected	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13842

Parameter | Name: **C13842 | Supervisor IP Address**			Data type: UNSIGNED_8 Index: 10733 = 0x29ED
Display of the supervisor IP address (Instance attribute 10 (Supervisor IP Address) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 137))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13842/1			Supervisor IP address (most significant byte)
C13842/2			Supervisor IP Address
C13842/3			Supervisor IP Address
C13842/4			Supervisor IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13843

Parameter | Name: **C13843 | Supervisor MAC ID**	Data type: OCTET_STRING Index: 10732 = 0x29EC
Display of the supervisor MAC ID (Instance attribute 10 (Supervisor MAC Address) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 137))	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13844

Parameter | Name: **C13844 | Beacon Times**	Data type: UNSIGNED_32 Index: 10731 = 0x29EB
Display of the beacon times (µs)	
Subcodes	**Info**
C13844/1	Beacon interval
C13844/2	Beacon timeout
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13845

Parameter | Name: **C13845 | Beacon Frames**	Data type: UNSIGNED_32 Index: 10730 = 0x29EA
Display of beacon frame information	
Subcodes	**Info**
C13845/1	Beacon frames - port X259
C13845/2	Beacon frame error - port X259
C13845/3	Beacon frames - port X260
C13845/4	Beacon frame error - port X260
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13846

Parameter | Name: **C13846 | Address Conflict Detection**		Data type: UNSIGNED_8 Index: 10729 = 0x29E9
Activation of the address conflict detection (ACD) (Instance attribute 10 (SelectAcd) in the TCP/IP Interface Object (245 / 0xF5) (📖 141)) Changing this value requires a reset of the device ("power off/on" or "type 0 reset").		
Selection list (Lenze setting printed in bold)		
0	Deactivated	
1	**Enable**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13847

Parameter | Name: **C13847 | Active Conflict Detection State**		Data type: UNSIGNED_8 Index: 10728 = 0x29E8
Display of the status of address conflict detection (ACD)		
Selection list (read only)		
0	Not conflicted	
1	Conflicted	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13848

Parameter | Name: **C13848 | Last Conflicted MAC ID**	Data type: OCTET_STRING Index: 10727 = 0x29E7
Display of the MAC address of the EtherNet/IP node with the last address conflict (ACD). The data of the last conflict will only be saved in this code if ACD is active at the moment when the conflict occurs (C13846 = 1). (Instance attribute 11 (RemoteMAC) in the TCP/IP Interface Object (245 / 0xF5) (🕮 141))	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13849

Parameter | Name: **C13849 | Last Conflicted IP Address**			Data type: UNSIGNED_8 Index: 10726 = 0x29E6
Display of the MAC address of the EtherNet/IP node with the last address conflict (ACD). The data of the last conflict will only be saved in this code if ACD is active at the moment when the conflict occurs (C13846 = 1).			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13849/1			Last conflicted IP address (most significant byte)
C13849/2			Last conflicted IP address
C13849/3			Last conflicted IP address
C13849/4			Last conflicted IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13850

Parameter | Name: **C13850 | All words to scanner**			Data type: INTEGER_16 Index: 10725 = 0x29E5
Display of the I/O data words transferred from the communication module (adapter) to the scanner. In the subcodes, all I/O data words transferred to the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13850/1			
...			
C13850/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from scanner**			Data type: INTEGER_16 Index: 10724 = 0x29E4
Display of the I/O data words transferred from the scanner to the communication module (adapter). In the subcodes, all I/O data words transferred from the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13851/1			
...			
C13851/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to the basic device**			Data type: INTEGER_16 Index: 10723 = 0x29E3
Display of the I/O data words transferred from the scanner to the communication module (adapter). In the subcodes, all I/O data words transferred from the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13852/1			
...			
C13852/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words to the basic device**			Data type: INTEGER_16 Index: 10722 = 0x29E2
Display of the I/O data words transferred from the communication module (adapter) to the scanner. In the subcodes, all I/O data words transferred to the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13853/1			
...			
C13853/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13858

Parameter | Name: **C13858 | Ethernet Port Statistics**	Data type: UNSIGNED_32 Index: 10717 = 0x29DD
Display of statistical values for the data transfer via the Ethernet connections	
Subcodes	**Info**
C13858/1	Ethernet port X259: RX
C13858/2	Ethernet port X259: RX CRC error
C13858/3	Ethernet port X259: RX discarded
C13858/4	Ethernet port X259: TX
C13858/5	Ethernet port X259: TX discarded
C13858/6	Ethernet port X260: RX
C13858/7	Ethernet port X260: RX CRC error
C13858/8	Ethernet port X260: RX discarded
C13858/9	Ethernet port X260: TX
C13858/10	Ethernet port X260: TX discarded
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13861

Parameter | Name: **C13861 | CIP Module Status**		Data type: UNSIGNED_16 Index: 10714 = 0x29DA
Display of the current CIP module status (Instance attribute 8 (State) in the Identity Object (1 / 0x01) (📖 126)) • The status is also indicated via the **MS** LED. ▸ LED status displays (📖 89)		
Selection list (read only)		
0	Nonexistent	
1	Device Self Testing	
2	Standby	
3	Operational	
4	Major Recoverable Fault	
5	Major Unrecoverable Fault	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

_ _

C13862

Parameter | Name: **C13862 | CIP Network Status**		Data type: UNSIGNED_16 Index: 10713 = 0x29D9
Display of the current CIP network status • The status is also indicated via the **NS** LED. ▸ LED status displays (📖 89)		
Selection list (read only)		
0	No IP Address	
1	Nonexistent	
2	Established	
3	Timed Out	
4	Duplicate IP	
5	Self-Test	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13863

Parameter | Name: **C13863 | Ethernet Port**		Data type: UNSIGNED_16 Index: 10712 = 0x29D8
Display of the baud rate currently used on the Ethernet connections		
Selection list (read only)		
0	Nonexistent	
1	10 Mbps/Half Duplex	
2	10 Mbps/Full Duplex	
3	100 Mbps/Half Duplex	
4	100 Mbps/Full Duplex	
5	Reserved	
6	Reserved	
Subcodes		**Info**
C13863/1		Ethernet port X259 link state
C13863/2		Ethernet port X260 link state
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13870

Parameter | Name: **C13870 | CIP Connections State**		Data type: UNSIGNED_16 Index: 10705 = 0x29D1
Display of the current CIP connection status		
Selection list (read only)		
0	Nonexistent	
3	Established	
4	Timed Out	
Subcodes		**Info**
C13870/1		Status of CIP connection 1
...		...
C13870/8		Status of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

_ _

C13871

Parameter | Name: **C13871 | CIP Connections Type**		Data type: UNSIGNED_16 Index: 10704 = 0x29D0
Display of the current CIP connection types • "Listen Only" connections are not displayed.		
Selection list (read only)		
0	Nonexistent	
1	Exclusive Owner	
2	Input Only	
3	Listen Only	
4	Explicit Connection	
Subcodes		**Info**
C13871/1		Type of CIP connection 1
...		...
C13871/8		Type of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13872

Parameter | Name: **C13872 | CIP Connection Triggers**		Data type: UNSIGNED_16 Index: 10703 = 0x29CF
Display of the current CIP connection class		
Selection list (read only)		
0	Nonexistent	
1	Class 1, Cyclic, Client	
163	Class 3, App. Obj., Server	
Subcodes		**Info**
C13872/1		Trigger of CIP connection 1
...		...
C13872/8		Trigger of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13873

Parameter | Name: **C13873 | CIP Connections RPI**			Data type: UNSIGNED_32 Index: 10702 = 0x29CE
Display of the RPI times (Requested Package Interval) currently used for the CIP connections ("Originator to Target" time)			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C13873/1			RPI of CIP connection 1
...			...
C13873/8			RPI of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13874

Parameter | Name: **C13874 | CIP Connections Timeout Time**			Data type: UNSIGNED_32 Index: 10701 = 0x29CD
Display of the timeouts (ms) of the CIP connections			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C13874/1			Timeout time of CIP connection 1
...			...
C13874/8			Timeout time of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13875

Parameter | Name: **C13875 | CIP Connections RUN/IDLE Flag**		Data type: UNSIGNED_16 Index: 10700 = 0x29CC
Display of the run and idle flags of the CIP connections		
Selection list (read only)		
0	Nonexistent	
1	IDLE	
2	RUN	
Subcodes		**Info**
C13875/1		RUN/IDLE flag - CIP connection 1
...		...
C13875/8		RUN/IDLE flag - CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13880

Parameter | Name: **C13880 | Monitoring Reaction**		Data type: UNSIGNED_8 Index: 10695 = 0x29C7
Setting of the monitoring response in the event of a Fault with regard to EtherNet/IP communication (📖 88) (Mapping of the Lenze object Lenze Class (101 / 0x65) (📖 155)) A change in the monitoring response becomes immediately effective.		
Selection list		
0	No response	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
Subcodes	**Lenze setting**	**Info**
C13880/1	0: No Response	Idle Mode
C13880/2	3: Quick stop by trouble	Fault Mode (only for "exclusive owner" connections)
C13880/3	0: No Response	Expl. message timeout
C13880/4	0: No Response	Overall Ethernet communication timeout
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13881

Parameter | Name: **C13881 | Overall Ethernet Timeout Time**			Data type: UNSIGNED_16 Index: 10694 = 0x29C6
Setting of the overall monitoring time (see Fault with regard to EtherNet/IP communication (📖 88)) A change in the monitoring response becomes immediately effective.			
Setting range (min. value | unit | max. value)			**Lenze setting**
500	ms	65535	**10000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Clear I/O Data**		Data type: UNSIGNED_8 Index: 10690 = 0x29C2
Setting which I/O data are to be processed by the adapter to maintain internal communication if ... • the network status of the controlling I/O connection is "Not connected" (see C13862) or • an idle event has occurred. Changes in the setting becomes effective immediately. (See Fault with regard to EtherNet/IP communication (📖 88).)		
Selection list (Lenze setting printed in bold)		
0	Use of last Scanner Output Data	
1	**Clear Scanner Output Data**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13899

Parameter | Name: **C13899 | Hostname**		Data type: VISIBLE_STRING Index: 10676 = 0x29B4
Each subcode contains a string with a length of 32 bytes indicating the designation of the EtherNet/IP node. (Instance attribute 6 (Host Name) in the TCP/IP Interface Object (245 / 0xF5) (📖 141))		
Subcodes	**Lenze setting**	**Info**
C13899/1		Host name
C13899/2		Host name
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13900

Parameter | Name: **C13900 | Firmware Product Type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3
The code contains a string with a length of 8 bytes. The identification code "E84AYCEO" is output.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware Compilation Date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2
The code contains a string with a length of 20 bytes. The software creation date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed (e.g. "Mar 21 2005 12:31:21").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware Version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1
The code contains a string with a length of 11 bytes. The firmware version is displayed (e.g. "01.00.01.00").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13920

Parameter | Name: **C13920 | Rotary coding switches: xxx.yyy.zzz.DDD**			Data type: UNSIGNED_8 Index: 10655 = 0x299F
Indication of the current value set by means of the rotary coding switches. ▸ Setting node addresses by means of rotary coding switches (📖 40)			
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

13.2 Table of attributes

The table of attributes contains information required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ No decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing is only possible when the controller is inhibited	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13000	IP address	11575	0x2D37	A	4	UNSIGNED_8		☑	☑	
C13001	Subnet mask	11574	0x2D36	A	4	UNSIGNED_8		☑	☑	
C13002	Gateway address	11573	0x2D35	A	4	UNSIGNED_8		☑	☑	
C13003	MAC-ID	11572	0x2D34	E	1	OCTET_STRING		☑		
C13005	IP Config Control	11570	0x2D32	E	1	UNSIGNED_8	1	☑	☑	
C13006	Multicast IP Start Address	11569	0x2D31	A	4	UNSIGNED_8		☑	☑	
C13010	Active IP Address	11565	0x2D2D	A	4	UNSIGNED_8	1	☑		
C13011	Active Subnetwork Mask	11564	0x2D2C	A	4	UNSIGNED_8	1	☑		
C13012	Active gateway address	11563	0x2D2B	A	4	UNSIGNED_8	1	☑		
C13016	Active Multicast IP Address	11559	0x2D27	A	4	UNSIGNED_8	1	☑		
C13017	Ethernet Config Control	11558	0x2D26	A	2	UNSIGNED_16	1	☑	☑	
C13018	Multicast Config Alloc Control	11557	0x2D25	E	1	UNSIGNED_8	1	☑	☑	
C13019	Multicast Config TTL Value	11556	0x2D24	E	1	UNSIGNED_8	1	☑	☑	
C13020	Multicast Config Num Mcast	11555	0x2D23	E	1	UNSIGNED_8	1	☑	☑	
C13021	Quality of Service (VLAN-Tagging)	11554	0x2D22	E	1	UNSIGNED_8	1	☑	☑	
C13022	Quality of Service (DSCP)	11553	0x2D21	A	7	UNSIGNED_8		☑	☑	
C13840	DLR Network Topology	10735	0x29EF	E	1	UNSIGNED_8	1	☑		
C13841	DLR Network Status	10734	0x29EE	E	1	UNSIGNED_8	1	☑		
C13842	Supervisor IP Address	10733	0x29ED	A	4	UNSIGNED_8	1	☑		
C13843	Supervisor MAC ID	10732	0x29EC	E	1	OCTET_STRING		☑		
C13844	Beacon Times	10731	0x29EB	A	2	UNSIGNED_32	1	☑		
C13845	Beacon Frames	10730	0x29EA	A	4	UNSIGNED_32	1	☑		
C13846	Address Conflict Detection	10729	0x29E9	E	1	UNSIGNED_8	1	☑	☑	
C13847	Active Conflict Detection State	10728	0x29E8	E	1	UNSIGNED_8	1	☑		
C13848	Last Conflicted MAC ID	10727	0x29E7	E	1	OCTET_STRING		☑		
C13849	Last Conflicted IP Address	10726	0x29E6	A	4	UNSIGNED_8	1	☑		
C13850	All words to scanner	10725	0x29E5	A	16	INTEGER_16	1	☑		
C13851	All words from scanner	10724	0x29E4	A	16	INTEGER_16	1	☑		
C13852	All words to the basic device	10723	0x29E3	A	16	INTEGER_16	1	☑		
C13853	All words to the basic device	10722	0x29E2	A	16	INTEGER_16	1	☑		
C13858	Ethernet Port Statistics	10717	0x29DD	A	10	UNSIGNED_32	1	☑		
C13861	CIP Module Status	10714	0x29DA	E	1	UNSIGNED_16	1	☑		
C13862	CIP Network Status	10713	0x29D9	E	1	UNSIGNED_16	1	☑		
C13863	Ethernet Port	10712	0x29D8	A	2	UNSIGNED_16	1	☑		
C13870	CIP Connections State	10705	0x29D1	A	8	UNSIGNED_16	1	☑		
C13871	CIP Connections Type	10704	0x29D0	A	8	UNSIGNED_16	1	☑		
C13872	CIP Connection Triggers	10703	0x29CF	A	8	UNSIGNED_16	1	☑		
C13873	CIP Connections RPI	10702	0x29CE	A	8	UNSIGNED_32	1	☑		
C13874	CIP Connections Timeout Time	10701	0x29CD	A	8	UNSIGNED_32	1	☑		
C13875	CIP Connections RUN/IDLE Flag	10700	0x29CC	A	8	UNSIGNED_16	1	☑		
C13880	Monitoring Reaction	10695	0x29C7	A	4	UNSIGNED_8	1	☑	☑	
C13881	Overall Ethernet Timeout Time	10694	0x29C6	E	1	UNSIGNED_16	1	☑	☑	
C13885	Clear I/O Data	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13899	Hostname	10676	0x29B4	A	2	VISIBLE_STRING		☑	☑	
C13900	Firmware Product Type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware Compilation Date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware Version	10673	0x29B1	E	1	VISIBLE_STRING		☑		
C13920	Rotary coding switch: xxx.yyy.zzz.DDD	10655	0x299F	E	1	UNSIGNED_8	1	☑		

14 Implemented CIP™ objects

An EtherNet/IP node can be seen as an accumulation of objects. An individual object is characterised by its class, its instances and attributes. Several services such as read and write services can be applied to these objects.

Note!

This chapter only describes the CIP objects implemented by Lenze and the properties supported by them (attributes, service codes etc.).

Not all object properties described in the "Common Industrial Protocol Specification" of the ODVA are supported.

"Common Industrial Protocol Specification" of the ODVA

Here you will find detailed information about the CIP objects.

Overview of the implemented CIP objects

CIP objects	Description
General objects	
Identity Object (1 / 0x01) (📖 126)	Identification and general information about the device
Message Router Object (2 / 0x02) (📖 128)	Addressing of a service for the transfer of data to any object class or instance
Assembly Object (4 / 0x04) (📖 129)	Input/output data of the scanner
Connection Manager Object (6 / 0x06) (📖 135)	Management of the internal resources for the transfer of data (implicit/explicit messaging)
EtherNet/IP objects	
Device Level Ring (DLR) Object (71 / 0x47) (📖 137)	Status information for the DLR protocol
Quality of Service (QoS) Object (72 / 0x48) (📖 139)	Classification and prioritisation of the data packets for EtherNet/IP communication
TCP/IP Interface Object (245 / 0xF5) (📖 141)	Configuration of the TCP/IP network interface of the device
Ethernet Link Object (246 / 0xF6) (📖 145)	General information and status information about the Ethernet interfaces of the device
AC drive profile objects	
Motor Data Object (40 / 0x28) (📖 149)	Data basis for motor parameters
Control Supervisor Object (41 / 0x29) (📖 150)	Management functions of the motor control devices.
AC Drive Object (42 / 0x2A) (📖 152)	Device-specific functions of the drive, e.g. speed ramps, torque control etc.
Lenze objects	
Lenze Class (101 / 0x65) (📖 155)	Lenze error responses to EtherNet/IP errors
Lenze Class (103 / 0x67) (📖 157)	Image of the scanner input data
Lenze Class (104 / 0x68) (📖 158)	Image of the scanner output data
Lenze Class (110 / 0x6E) (📖 159)	Access to Lenze codes

General class attributes

Attribute ID	Service	Name	Data type	Description
1	Get	Revision	UINT	Revision no. of the object
2	Get	Max. Instance	UINT	Max. number of instances of the object
3	Get	Number of Instances	UINT	Number of instances of the object
4	Get	Optional Attribute List:	STRUCT of:	List of the optional instance attributes:
		Number Attributes	UINT	Number of optional attributes
		Optional Attributes	ARRAY of UINT	Listing of the optional attributes
5	Get	Optional Service List:	STRUCT of:	List of the optional services:
		Number Services	UINT	Number of optional services
		Optional Services	ARRAY of UINT	Listing of the optional services
6	Get	Max. ID Number Class Attributes	UINT	The attribute ID of the last class attribute of the class description implemented in the device
7	Get	Max. ID Number Instance Attributes	UINT	The attribute ID of the last instance attribute of the class description implemented in the device

14.1 General CIP objects

14.1.1 Identity Object (1 / 0x01)

The "Identity Object" provides the identification and general information about the device.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)
6	Get	Max. ID Number Class Attributes	UINT	7 (0x0007)
7	Get	Max. ID Number Instance Attributes	UINT	8 (0x0008)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Vendor ID	UINT	587 (0x024B)
2	Get	Device Type	UINT	2 (0x0002): AC Drive
3	Get	Product Code	UINT	8400 (0x20D0)
4	Get	Revision:	STRUCT of:	Firmware revision of the device
		Major Revision	USINT	
		Minor Revision	USINT	
5	Get	Status	WORD	Current device status (status bits) • Instance attribute "Status" (attribute 5) (📖 127) • EtherNet/IP state diagram (📖 55)
6	Get	Serial Number	UDINT	Serial number of the device
7	Get	Product Name	SHORT_STRING	E84AYCEO
8	Get	State	USINT	Current device state: • 0: Nonexistent • 1: Device self-testing • 2: Standby • 3: Operational • 4: Major recoverable fault • 5: Major unrecoverable fault • 6 ... 254: Reserved • 255: Standard for "Get_Attributes_All" service (See also C13861, LED status displays (📖 89))

Instance attribute "Status" (attribute 5)

Bits	Name	Description
0	Owned	The state '1' indicates that the device (or an object within the device) has an owner. Within a master/slave model, the state '1' indicates that the "predefined master/slave connection set" is assigned to a master. Outside the master/slave model, it means "TBD".
1	-	Reserved (0)
2	Configured	The state '1' indicates that the activities the device application carries out differ from the "out of box" standard configuration. This should not comprise the configuration of communication.
3	-	Reserved (0)
4 ... 7	Extended Device Status	• 0000: Status is "self-testing" or unknown • 0001: Firmware update is being carried out • 0010: At least one faulty I/O connection • 0011: No I/O connections available • 0100: Non-volatile configuration is faulty • 0101: "Major fault" (bit 10 or 11 is '1') • 0110: At least one I/O connection is in the "run mode" • 0111: At least one I/O connection is available, all are in the "idle mode" • 1000: Reserved • 1001: Reserved • 1010 ... 1111: Reserved / manufacturer-specific
8	Minor Recoverable Fault	The state '1' indicates that a "Minor Recoverable Fault" has occurred.
9	Minor Unrecoverable Fault	The state '1' indicates that a "Minor Unrecoverable Fault" has occurred.
10	Major Recoverable Fault	The state '1' indicates that a "Major Recoverable Fault" has occurred.
11	Major Unrecoverable Fault	The state '1' indicates that a "Major Unrecoverable Fault" has occurred.
12 ... 15	Extended Device Status 2	Reserved (0) / manufacturer-specific

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x05	Reset	These reset service types are supported: • 0: Mains switching (power off/on) is simulated. • 1: The parameters of the device are reset to the Lenze setting and mains switching (power off/on) is simulated.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.1.2 Message Router Object (2 / 0x02)

With the "Message Router Object", a client can address a service for the transfer of data to any object class or instance.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)
4	Get	Optional Attribute List:	STRUCT of:	
		Number Attributes	UINT	2 (0x0002)
		Optional Attributes	ARRAY of UINT	1 , 2 (0x0001.0002)
5	Get	Optional Service List:	STRUCT of:	
		Number Services	UINT	1 (0x0001)
		Optional Services	ARRAY of UINT	10 (0x000A)
6	Get	Max. ID Number Class Attributes	UINT	7 (0x0007)
7	Get	Max. ID Number Instance Attributes	UINT	6 (0x0006)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Object list:	STRUCT of:	Object list:
		Number	UINT	Number of supported object class codes
		Classes	ARRAY of UINT	Listing of the supported object class codes
2	Get	Number Available	UINT	Max. number of supported connections

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.1.3 Assembly Object (4 / 0x04)

For data exchange, the communication module supports the following assembly object instances:

Application	Instance ID		Assembly object instance
	[dec]	[hex]	
Lenze technology applications / user-definable parameter sets	110	0x6E	Custom Output
	111	0x6F	Custom Input
"AC Drive Profile" application	20	0x14	Basic Speed Control Output
	21	0x15	Extended Speed Control Output
	22	0x16	Speed and Torque Control Output
	23	0x17	Extended Speed and Torque Control Output
	70	0x46	Basic Speed Control Input
	71	0x47	Extended Speed Control Input
	72	0x48	Speed and Torque Control Input
	73	0x49	Extended Speed and Torque Control Input

The contents of the input and output data depends on the I/O data arrangement in the drive (I/O data mapping (📖 57)).

Application note

An example of parameter data transfer (read/write parameters) in a "AC Drive Profile" application can be found in the download area (Application Knowledge Base) at www.Lenze.com.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	130 (0x0082)
3	Get	Number of Instances	UINT	11 (0x000B)
4	Get	Optional Attribute List:	STRUCT of:	
		Number Attributes	UINT	1 (0x0001)
		Optional Attributes	ARRAY of UINT	4 (0x0004)
6	Get	Max. ID Number Class Attributes	UINT	7 (0x0007)
7	Get	Max. ID Number Instance Attributes	UINT	4 (0x0004)

Instance attributes for output data of the scanner

Attribute ID	Service	Name	Data type	Value
3	Get / Set	Data	ARRAY of SINT / INT / DINT	Max. 16 bytes from the scanner to the adapter: • 20 (0x14): Basic Speed Control Output • 21 (0x15): Extended Speed Control Output • 22 (0x16): Speed and Torque Control Output • 23 (0x17): Extended Speed and Torque Control Output • 110 (0x6E): Custom Output Instance attribute "Data" (attribute 3) (131)
4	Get	Size	UINT	Number of bytes in attribute 3 (Data)

Assembly output objects (scanner to adapter) are assumed to have a 4-byte header (32-bit "run/idle header"). When mapping the assemblies, this header will automatically be added to the data stream by most Allen-Bradley PLC/SLC equipment.

If your PLC does not support this header (like the Rockwell PLCs do), add a preceding 32-bit header to the output image.

You can then define the **bit 0** of this header in the process image of your PLC:

- State ’0’: idle mode
- State ’1’: run mode

For the operation with Rockwell PLCs, adaptations are not required.

The Lenze Class (104 / 0x68) (158) provides the image of the output data of the scanner.

Instance attributes for input data of the scanner

Attribute ID	Service	Name	Data type	Value
3	Get / Set	Data	ARRAY of SINT / INT / DINT	Max. 16 bytes from the adapter to the scanner: • 70 (0x46): Basic Speed Control Input • 71 (0x47): Extended Speed Control Input • 72 (0x48): Speed and Torque Control Input • 73 (0x49): Extended Speed and Torque Control Input • 111 (0x6F): Custom Input ▶ Instance attribute "Data" (attribute 3) (📖 131)
4	Get	Size	UINT	Number of bytes in attribute 3 (Data)

The assembly input objects (adapter to scanner) are mapped in the adapter memory starting with byte 0.

The input objects are transmitted in a "modeless" manner, i.e. a 4-byte header is not included in the transfer.

So the start address in the assembly memory map is the actual start of the first assembly data element.

Please observe the actual assembly lengths when mapping the input objects to the controller memory.

The Lenze Class (103 / 0x67) (📖 157) provides the image of the input data of the scanner.

Instance attribute "Data" (attribute 3)

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
20 (0x14)	0						Fault Reset		Run Fwd
	1								
	2	Speed Reference (Low Byte)							
	3	Speed Reference (High Byte)							
21 (0x15)	0		Net Ref	NetCtrl			Fault Reset	Run Rev	Run Fwd
	1								
	2	Speed Reference (Low Byte)							
	3	Speed Reference (High Byte)							
22 (0x16)	0						Fault Reset		Run Fwd
	1								
	2	Speed Reference (Low Byte)							
	3	Speed Reference (High Byte)							
	4	Torque Reference (Low Byte)							
	5	Torque Reference (High Byte)							

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
23 (0x17)	0		Net Ref	NetCtrl			Fault Reset	Run Rev	Run Fwd
	1								
	2	Speed Reference (Low Byte)							
	3	Speed Reference (High Byte)							
	4	Torque Reference (Low Byte)							
	5	Torque Reference (High Byte)							
110 (0x6E)	0	Custom Output							
	...	...							
	31	Custom Output							
70 (0x46)	0						Running1 (Fwd)		Faulted
	1								
	2	Speed Actual (Low Byte)							
	3	Speed Actual (High Byte)							
71 (0x47)	0	At Reference	RefFrom Net	CtrlFrom Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
	1	Drive State							
	2	Speed Actual (Low Byte)							
	3	Speed Actual (High Byte)							
72 (0x48)	0						Running1 (Fwd)		Faulted
	1								
	2	Speed Actual (Low Byte)							
	3	Speed Actual (High Byte)							
	4	Torque Actual (Low Byte)							
	5	Torque Actual (High Byte)							
73 (0x49)	0	At Reference	RefFrom Net	CtrlFrom Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
	1	Drive State							
	2	Speed Actual (Low Byte)							
	3	Speed Actual (High Byte)							
	4	Torque Actual (Low Byte)							
	5	Torque Actual (High Byte)							
111 (0x6F)	0	Custom Input							
	...	...							
	31	Custom Input							

Note!

In order to be able to use the torque control for the **assembly object instances 22 (0x16), 23 (0x17), 72 (0x48), 73 (0x49)**, the "DriveMode" attribute has to be written by means of explicit message transfer.

▸ Write "DriveMode" attribute (📖 154)

Data mapping of the output assemblies

Data component [Bits 0 ... 7]	Class		Instance Number	Attribute	
	Name	Number		Name	Number
RunFwd [Bit 0]	Control Supervisor	0x29	1	Run1	3
RunRev [Bit 1]	Control Supervisor	0x29	1	Run2	4
FaultReset [Bit 2]	Control Supervisor	0x29	1	FaultRst	12
NetCtrl [Bit 5]	Control Supervisor	0x29	1	NetCtrl	5
NetRef [Bit 6]	AC Drive	0x2A	1	NetRef	4
Drive Mode [Bits 0 ... 7]	AC Drive	0x2A	1	DriveMode	6
Speed Reference [Bits 0 ... 7]	AC Drive	0x2A	1	SpeedRef	8
Torque Reference [Bits 0 ... 7]	AC Drive	0x2A	1	TorqueRef	12
Custom Output [Bits 0 ... 7]					

Note!

In case of the **assembly object instances 21 (0x15) and 23 (0x17)**, **NetCtrl (Bit 5)** and **NetRef (Bit 6)** have to be set in order that the drive can receive start/stop commands and speed/torque commands via the network.

Data mapping of the input assemblies

Data component [Bits 0 ... 7]	Class		Instance Number	Attribute	
	Name	Number		Name	Number
Faulted [Bit 0]	Control Supervisor	0x29	1	Faulted	10
Warning [Bit 1]	Control Supervisor	0x29	1	Warning	11
Running1 (Fwd) [Bit 2]	Control Supervisor	0x29	1	Running1	7
Running2 (Rev) [Bit 3]	Control Supervisor	0x29	1	Running2	8
Ready [Bit 4]	Control Supervisor	0x29	1	Ready	9
CtrlFromNet [Bit 5]	Control Supervisor	0x29	1	CtrlFromNet	15
RefFromNet [Bit 6]	AC Drive	0x2A	1	RefFromNet	29
At Reference [Bit 7]	AC Drive	0x2A	1	AtReference	3
Drive State [Bits 0 ... 7]	Control Supervisor	0x29	1	State	6
Speed Actual [Bits 0 ... 7]	AC Drive	0x2A	1	SpeedActual	7
Torque Actual [Bits 0 ... 7]	AC Drive	0x2A	1	TorqueActual	11
Custom Input [Bits 0 ... 7]					

Supported service codes

Service code [hex]	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.1.4 Connection Manager Object (6 / 0x06)

The "Connection Manager Object" manages the internal resources for the I/O data transfer (implicit messaging) and the parameter data transfer (explicit messaging). The instance specified by the "connection manager" class refers to a "connection instance" or a "connection object".

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)
4	Get	Optional Attribute List:	STRUCT of:	
		Number Attributes	UINT	8 (0x0008)
		Optional Attributes	ARRAY of UINT	1 ... 8 0x0001.0002.0003.0004.0005.0006.0007.0008
6	Get	Max. ID Number Class Attributes	UINT	7 (0x0007)
7	Get	Max. ID Number Instance Attributes	UINT	8 (0x0008)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Set 1)	Open Requests	UINT	Number of "Forward Open Service Requests" received
2	Set 1)	Open Format Rejects	UINT	Number of "Forward Open Service Requests" rejected due to faulty format.
3	Set 1)	Open Resource Rejects	UINT	Number of "Forward Open Service Requests" rejected due to a lack of resources.
4	Set 1)	Open Other Rejects	UINT	Number of "Forward Open Service Requests" rejected due to other reasons than faulty format or lack of resources.
5	Set 1)	Close Requests	UINT	Number of "Forward Close Service Requests" received
6	Set 1)	Close Format Requests	UINT	Number of "Forward Close Service Requests" rejected due to faulty format.
7	Set 1)	Close Other Requests	UINT	Number of "Forward Close Service Requests" rejected due to other reasons than faulty format.
8	Set 1)	Connection Timeouts	UINT	Total number of "Connection Timeouts" that have occurred within the connections monitored by this object.

1) A device can reject a "Request" of the attribute with the general status code "0x09" (invalid attribute value) if the attribute value sent is unequal to zero.

Instance error messages

Error code [hex]	Extended code [hex]	Error designation	Description
0x000	-	SUCCESS	No error
0x001	0x106	OWNERSHIP_CONFLICT	The connection could not be established because another connection has already occupied the required resources. It is only possible to establish an "exclusive owner" connection to the adapter.
0x001	0x119	NON- LISTEN ONLY CONNECTION NOT OPENED	The connection could not be established because there is no "non-listen only" connection (input only, exclusive owner). The "non-listen only" connection must be of the "multicast" connection type.
0x001	0x127	INVALID_ORIGINATOR_TO_ TARGET_SIZE	The resulting length of the ports mapped in the receive object PDO_RX0 does not correspond to the number of data bytes of the assembly object instance 110 (0x6E, Custom Output) defined in the scanner.
0x001	0x128	INVALID_TARGET_TO_ ORIGINATOR_SIZE	The resulting length of the ports mapped in the transmit object PDO_TX0 does not correspond to the number of data bytes of the assembly object instance 111 (0x6F, Custom Input) defined in the scanner.
0x001	0x204	UNCONNECTED_REQUEST_ TIMED_OUT	The adapter does not respond to the establishment of the connection. • There might be no physical connection. • The adapter is switched off. • The adapter has an invalid IP configuration.
0x001	0x320	ACCESS_CONTENTION	Manufacturer-specific error: • The configurations of the assembly input and output objects are mixed up. • The connection could not be established because another connection has already occupied the required resources. It is only possible to establish an "exclusive owner" connection to the adapter.
0x001	0x111	ROUTER_EXT_ERR_RPI_ NOT_SUPPORTED	The RPI set for a connection is not supported. • Min. class 1 RPI = 4 ms • Min. class-3-RPI = 10 ms
0x001	0x112	RROUTER_EXT_ERR_RPI_ VALUE_NOT_ACCEPTABLE	The RPI set for a connection is not supported. • Min. class 1 RPI = 4 ms • Min. class-3-RPI = 10 ms
0x001	0x123	ROUTER_EXT_ERR_INVALID_ TO_ CONNECTION_TYPE	The output image connection type is invalid or not supported.
0x001	0x124	ROUTER_EXT_ERR_INVALID_ TO_ CONNECTION_TYPE	The input image connection type is invalid or not supported.
0x001	0x12A	ROUTER_EXT_ERR_INVALID_ CONSUMING_PATH	The path specification for the output data from the scanner is invalid.
0x001	0x12B	ROUTER_EXT_ERR_INVALID_ PRODUCING_PATH	The path specification for the input data to the scanner is invalid.

Supported service codes

Service code [hex]	Name	Description
0x54	Forward_Open	Opens a CIP connection from the PLC to the target drive.
0x4E	Forward_Close	Closes a CIP connection from the PLC to the target drive.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.2 EtherNet/IP objects

14.2.1 Device Level Ring (DLR) Object (71 / 0x47)

The "Device Level Ring (DLR) Object" provides status information for the DLR protocol. The DLR protocol is a "layer 2" protocol enabling the use of an Ethernet ring topology.

Note!

Only the "beacon-based ring node" mode is supported.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)
6	Get	Max. ID Number Class Attributes	UINT	7 (0x0007)
7	Get	Max. ID Number Instance Attributes	UINT	2 (0x0002)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Network Topology	USINT	Current network topology • 0: Line topology • 1: Ring topology (Display via C13840)
2	Get	Network Status	USINT	Current network status • 0: Normal • 1: Ring Fault (only for ring topology) • 2: Unexpected Loop Detected (only for line topology) (Display via C13841)
10	Get	Active Supervisor Address	STRUCT of:	IP and MAC address of the active ring supervisor
		Supervisor IP Address	UDINT	Ethernet MAC address The value '0' indicates that no IP address has been configured for the device. (Display via C13842)
		Supervisor MAC Address	ARRAY of USINT[6]	Ethernet MAC address (Display via C13843)
12	Get	Capability Flags	DWORD	Telegram processing method for the ring node implementation • 2: Beacon-based ring node ▶ Instance attribute "Capability Flags" (attribute 12) (📖 138)

Instance attribute "Capability Flags" (attribute 12)

Bits	Name	Description
0	Announce-based Ring Node	Is not supported (state ’0’).
1	Beacon-based Ring Node	The state ’1’ is set if the ring node implementation is based on the processing of "beacon frames". See also: • C13844 (Beacon Times) • C13845 (Beacon Frames)
2 ... 31	-	Reserved (0)

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.
0x18	Get_Member	Outputs the members of a certain attribute.

14.2.2 Quality of Service (QoS) Object (72 / 0x48)

The "Quality of Service (QoS) Object" enables different classifications and prioritisations of the data packets for EtherNet/IP communication. For this purpose, the EtherNet/IP messages are marked with "802.1Q tags" and "Differentiated Services Codepoints" (DSCP).

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)
6	Get	Max. ID Number Class Attributes	UINT	7 (0x0007)
7	Get	Max. ID Number Instance Attributes	UINT	8 (0x0008)

Instance attributes

The instance attributes act independently of each other.

The DSCP values are used for the IP headers.

Irrespective of this, VLAN tagging can be activated in addition (C13021 = 1).

The VLAN ID of Lenze devices is ’0’.

The VLAN priority results from the DSCP values configured.

Changes in the attribute values will only take effect after a reset of the device ("power off/on" or "type 0 reset").

Note!

Before activating VLAN tagging, please ensure that all involved components support VLAN tagging. It may occur that the devices not supporting VLAN tagging cannot be accessed anymore.

Attribute ID	Service	Name	Data type	Value
1	Set	802.1Q Tag Enable	USINT	Enables the sending of data packets with 802.1Q tags (C13021) • 0: No use of 802.1Q tags (Lenze setting) • 1: Use of 802.1Q tags
4	Set	DSCP Urgent	USINT	55: Urgent/imperative messages **Is not supported at the moment.**
5	Set	DSCP Scheduled	USINT	47: Scheduled messages (Can only be used for "exclusive owner" connections.) (C13022/4)
6	Set	DSCP High	USINT	43: Messages with high priority (Can only be used for "input only" and "listen only" connections.) (C13022/5)

Attribute ID	Service	Name	Data type	Value
7	Set	DSCP Low	USINT	31: Messages with low priority **Is not supported at the moment.**
8	Set	DSCP Explicit	USINT	27: "Explicit messages" (parameter data) (C13022/7)

Supported service codes

Service code [hex]	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.2.3 TCP/IP Interface Object (245 / 0xF5)

The "TCP/IP Interface Object" serves to configure the TCP/IP network interface of the device.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)
4	Get	Optional Attribute List:	STRUCT of:	
		Number Attributes	UINT	4 (0x0004)
		Optional Attributes	ARRAY of UINT	8 ... 11 (0x0008.0009.000A.000B)
6	Get	Max. ID Number Class Attributes	UINT	0x0007
7	Get	Max. ID Number Instance Attributes	UINT	0x000B

Instance attributes

Note!

Write access to attribute 3 (Configuration Control) permanently saves the TCP/IP configuration defined in attribute 5.

If the TCP/IP configuration defined in attribute 5 is to be used as a "static IP" during the start-up, "0 = use static TCP/IP configuration" must be set in attribute 3.

Attribute ID	Service	Name	Data type	Value
1	Get	Status	DWORD	Current status of the TCP/IP network interface ▸ Instance attribute "Status" (attribute 1) (📖 143)
2	Get	Configuration Capability	DWORD	Possible options for TCP/IP configuration ▸ Instance attribute "Configuration Capability" (attribute 2) (📖 143)
3	Get / Set	Configuration Control	DWORD	Selection of how the TCP/IP configuration is to be made (C13005): Possible values for bits 0 ... 3: • 0000: Use static TCP/IP config. • 0001: TCP/IP config. via BOOTP • 0010: TCP/IP config. via DHCP Bits 4 ... 31 are reserved (0).
4	Get	Physical Link Object:	STRUCT of:	Path to "Physical Link Object"
		Path Size	UINT	2 (0x0002)
		Path	padded EPATH	• 32 (0x0020) • 246 (0x00F6) • 36 (0x0024) • 1 (0x0001)

Attribute ID	Service	Name	Data type	Value
5	Get	Interface Configuration:	STRUCT of:	Current TCP/IP configuration
		IP Address	UDINT	C13010 (active IP address)
		Network Mask	UDINT	C13011 (active subnet mask)
		Gateway Address	UDINT	C13012 (active gateway address)
		Name Server	UDINT	
		Name Server 2	UDINT	
		Domain Name 1	STRING	
6	Get / Set	Host Name	STRING	Host name of the device (C13899, max. 64 ASCII characters)
8	Get / Set	TTL Value	USINT	TTL value (C13019) for EtherNet/IP multicast data packets (value range: 1 ... 255)
9	Get / Set	Mcast Config:	STRUCT of:	Configuration of the multicast IP addressing
		Alloc Control	USINT	Control word (C13018) for addressing: • 0: The multicast IP addresses are generated with the standard assignment algorithm. • 1: The multicast IP addresses are assigned via the values in "Num Mcast" and "Mcast Start Addr" (C13006) • 2: Reserved
		Reserved	USINT	0 (0x0000)
		Num Mcast	UINT	Total number of multicast IP addresses assigned (C13020)
		Mcast Start Addr	UDINT	Active multicast IP start address (C13016)
10	Set	SelectAcd	BOOL	Activation of address conflict detection (ACD, C13846) • 0: Disable ACD • 1: Enable ACD Changing this value requires a reset of the device ("power off/on" or "type 0 reset").
11	Get / Set	LastConflictDetected:	STRUCT of:	ACD diagnostics information about the last occurred address conflict
		AcdActivity	USINT	Status of the ACD algorithm when the last address conflict occurred: • 0: NoConflictDetected (default) • 1: ProbeIpv4Address • 2: OngoingDetection • 3: SemiActiveProbe
		RemoteMAC	ARRAY of USINT[6]	MAC address of the device with the last address conflict
		ArpPdu	ARRAY of USINT[28]	Reproduction of the ARP message with information about the address conflict ▶ Structure of the ARP message (attribute 11, "ArpPdu") (📖 144)

Instance attribute "Status" (attribute 1)

Bits	Name	Description
0 ... 3	Interface Configuration Status	• 0000: No TCP/IP configuration available (attribute 5) • 0001: Valid TCP/IP configuration (attribute 5) via BOOTP, DHCP or static/permanent storage • 0010 ... 1111: Reserved
4	Mcast Pending	This bit indicates a pending change of the multicast configuration in attribute 9 (Mcast Config) and/or the TTL value (C13019). It is set to ’1’ if either a multicast attribute or the TTL value is set. The pending change will only take effect after a reset of the device ("power off/on" or "type 0 reset"). This bit is then reset to ’0’.
5	Interface Configuration Pending	This bit indicates a pending change of the TCP/IP configuration in attribute 5 (Interface Configuration). It is set to ’1’ if an attribute is set. The pending change will only take effect after a reset of the device ("power off/on" or "type 0 reset").
6	AcdStatus	Display of the status of address conflict detection (ACD, C13847): • 0: No address conflict detected • 1: Address conflict detected
7 ... 31	-	Reserved (0)

Instance attribute "Configuration Capability" (attribute 2)

Bits	Name	Description
0	BOOTP Client	The state ’1’ indicates that BOOTP is used for the TCP/IP configuration of the device.
1	DNS Client	Is not supported (state ’0’).
2	DHCP Client	The state ’1’ indicates that DHCP is used for the TCP/IP configuration of the device.
3	DHCP-DNS Update	Is not supported (state ’0’).
4	Configuration Settable	The state ’1’ indicates that the TCP/IP configuration can be set in attribute 5 (Interface Configuration).
5	Hardware Configurable	Is not supported (state ’0’).
6	Interface Configuration Change Requires Reset	The state ’1’ indicates that changes of the TCP/IP configuration in attribute 5 (Interface Configuration) will only take effect after a reset of the device ("power off/on" or "type 0 reset"). The state ’0’ is not supported (changes becoming effective immediately).
7	AcdCapable	The state ’1’ indicates that the device is provided with address conflict detection (ACD).
8 ... 31	-	Reserved (0)

Structure of the ARP message (attribute 11, "ArpPdu")

Field size [Bytes]	Field name	Value
2	Hardware Address Type	1: Ethernet H/W
2	Protocol Address Type	0x0800: IP
1	HADDR LEN	6: Ethernet H/W
1	PADDR LEN	4: IP
2	OPERATION	1: Request 2: Response
6	SENDER HADDR	H/W address of the sender
4	SENDER PADDR	Protocol address of the sender
6	TARGET HADDR	H/W address of the target
4	TARGET PADDR	Protocol address of the target

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.2.4 Ethernet Link Object (246 / 0xF6)

The "Ethernet Link Object" provides general information and status information about the Ethernet interfaces (IEEE 802.3).

Note!

Write accesses to writable attributes become effective immediately.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	3 (0x0003)
2	Get	Max. Instance	UINT	2 (0x0002)
3	Get	Number of Instances	UINT	2 (0x0002)
4	Get	Optional Attribute List:	STRUCT of:	
		Number Attributes	UINT	4 (0x0004)
		Optional Attributes	ARRAY of UINT	7 ... 10 (0x0007.0008.0009.000A)
6	Get	Max. ID Number Class Attributes	UINT	0x0007
7	Get	Max. ID Number Instance Attributes	UINT	0x000A

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Interface Speed	UDINT	Current baud rate • 10 Mbps • 100 Mbps
2	Get	Interface Flags	DWORD	Status bits of the Ethernet interface ▸ Instance attribute "Interface Flags" (attribute 2) (📖 146)
3	Get	Physical Address	ARRAY of USINT[6]	MAC address of the Ethernet interface
6	Set	Interface Control	STRUCT of:	
		Control Bits	WORD	Control bits for the Ethernet interface ▸ Instance attribute "Control Bits" (attribute 6, Interface Control) (📖 147)
		Forced Interface Speed	UINT	Baud rate [in Mbps] at which the Ethernet interface is to be operated (C13017). Example values: • 10 = 10 Mbps • 100 = 100 Mbps
7	Get	Interface Type	USINT	Interface type (transmission medium) • 0: Unknown interface type • 1: Device-internal interface (e.g. embedded switch) • 2: Twisted pair (e.g. 100Base-TX), Lenze setting • 3: Optical fibre (e.g. 100Base-FX) • 4 ... 255: Reserved

Attribute ID	Service	Name	Data type	Value
8	Get	Interface State	USINT	Current operating status of the Ethernet interface • 0: Unknown status • 1: Enable (The interface can send and receive data.) • 2: Disable • 3: Testing • 4 ... 255: Reserved
9	Set	Admin State	USINT	Administrative status • 0: Reserved • 1: Enable • 2: Disable • 3 ... 255: Reserved
10	Get	Interface Label	SHORT_ STRING	Text for the identification/designation of the Ethernet interface

Instance attribute "Interface Flags" (attribute 2)

Bits	Name	Description
0	Link Status	This bit indicates whether the Ethernet interface is connected to an active network. • 0: No Ethernet connection available • 1: Ethernet connection available
1	Half/Full Duplex	This bit indicates the current transmission mode of the Ethernet interface. • 0: Half duplex • 1: Full duplex **Note:** If "Link Status" bit = 0, it is not possible to determine the value of the "Half/Full Duplex" bit.

2 ... 4	Negotiation Status	These bits indicate the status of "Link Auto-Negotiation".
		• 000: "Link Auto-Negotiation" is being processed.
		• 001: "Link Auto-Negotiation" and speed detection have failed.
		  • Use the standard baud rate and transmission mode values.
		  • The standard values depend on the product; recommended values are ’10 Mbps’ and ’Half Duplex’.
		• 010: "Link Auto-Negotiation" has failed, but a baud rate has been detected.
		  • Use the recommended value (’Half Duplex’) for the transmission mode .
		• 011: "Link Auto-Negotiation" and speed detection have been successful.
		• 100: No "Link Auto-Negotiation" active.

5	Manual Setting Requires Reset	Reset after changes in the link parameters • 0: The Ethernet interface can activate changes in the link parameters (auto-negotiation, transmission mode, baud rate) automatically. • 1: If the link parameters (auto-negotiation, transmission mode, baud rate) are changed, the device must be reset ("power off/on" or "type 0 reset").
6	Local Hardware Fault	Hardware fault detection • 0: No hardware fault has been detected on the Ethernet interface. • 1: A hardware fault has been detected on the Ethernet interface.
7 ... 31	-	Reserved (0)

Instance attribute "Control Bits" (attribute 6, Interface Control)

Bits	Name	Description
0	Auto-negotiate	Activation of "Link Auto-Negotiation" • 0: "Link Auto-Negotiation" is not active. The device uses the settings of the bits "Forced Duplex Mode" (bit 1) and "Forced Interface Speed" (see attribute 6, Interface Control). • 1: "Link Auto-Negotiation" is active.
1	Forced Duplex Mode	If "Auto-negotiate" bit = 0, this bit indicates the transmission mode to be used. • 0: Half duplex • 1: Full duplex
2 ... 15	-	Reserved (0)

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.3 "AC Drive Profile" objects

From inverter drive version V13.00 and communication module version V01.02, the EtherNet/IP specified "AC Drive Profile" is supported.

The standard device code **C00005 = "1100: AC Drive Profile"** serves to select the "AC Drive Profile" application.

The "AC Drive Profile" contains ...

- the data basis for motor parameters,
- management functions of the motor control devices,
- Device-specific functions of the drive, e.g. speed ramps, torque control etc.

For using the "AC Drive Profile", the following assembly object instances in the host (scanner) have to be used:

Instance ID		**Assembly object instance**	
[dec]	**[hex]**		
20	0x14	Basic Speed Control Output	Outputs: From the scanner to the adapter
21	0x15	Extended Speed Control Output	
22	0x16	Speed and Torque Control Output	
23	0x17	Extended Speed and Torque Control Output	
70	0x46	Basic Speed Control Input	Inputs: From the adapter to the scanner
71	0x47	Extended Speed Control Input	
72	0x48	Speed and Torque Control Input	
73	0x49	Extended Speed and Torque Control Input	

See also Assembly Object (4 / 0x04) (📖 129)

Software manual / »Engineer« online help for the Inverter Drive 8400

Here you can find detailed information on how to use the "AC Drive Profile".

14.3.1 Motor Data Object (40 / 0x28)

The "Motor Data Object" provides a data basis for motor parameters.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	NumAttr	USINT	Number of supported attributes
2	Get	Attributes	ARRAY of USINT	Listing of the supported attributes
3	Get / Set	MotorType	USINT	AC motor type • 6: Wound-rotor induction motor • 7: Squirrel-cage induction motor

Instance attributes for AC motor types

Attribute ID	Service	Name	Data type	Value
6	Get / Set	RatedCurrent	UINT	Rated stator current [100mA]
7	Get / Set	RatedVoltage	UINT	Rated base voltage [V]

For a write access to the attributes *RatedCurrent* and *RatedVoltage*, the controller enable (RFR = 0) has to be deactivated.

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.3.2 Control Supervisor Object (41 / 0x29)

The "Control Supervisor Object" describes all management functions of the devices used to control the motor.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	NumAttr	USINT	Number of supported attributes
2	Get	Attributes	ARRAY of USINT	Listing of the supported attributes
3	Set	Run1	BOOL	Run/stop can be controlled via a local setting on the device or terminal, or via the network (see attribute "NetCtrl"). The relationship between Run1, Run2 and the trigger events is described in the Run/stop event (📖 151) section.
4	Set	Run2	BOOL	
5	Set	NetCtrl	BOOL	Run/Stop control • 0: Run/stop control via local setting on the device or terminal • 1: Run/stop control via network (e.g. from the scanner)
6	Get	State	USINT	• 0: Manufacturer-specific • 1: Startup • 2: Not_Ready • 3: Ready • 4: Enabled • 5: Stopping • 6: Fault_Stop • 7: Faulted
7	Get	Running1	BOOL	• 0: Other status than listed under '1' • 1: [Enabled and Run1] or [Stopping and Running1] or [Fault_Stop and Running1]
8	Get	Running2	BOOL	• 0: Other status than listed under '1' • 1: [Enabled and Run2] or [Stopping and Running2] or [Fault_Stop and Running2]
9	Get	Ready	BOOL	• 0: Other status than listed under '1' • 1: Ready or Enabled or Stopping
10	Get	Faulted	BOOL	• 0: No errors • 1: Errors have occurred
11	Get	Warning	BOOL	• 0: No warnings • 1: Warnings have occurred
12	Set	FaultRst	BOOL	• 0 → **1: Reset error** • 0: No Response

Attribute ID	Service	Name	Data type	Value
13	Get	FaultCode	UINT	DRIVECOM error code of the error causing the Faulted status. ▶ Mapping of Lenze device errors to DRIVECOM errors (📖 102)
15	Get	CtrlFromNet	BOOL	Status of run/stop control • 0: Run/stop control via local setting on the device or terminal • 1: Run/stop control via network (e.g. from the scanner)

Run/stop event

Relationships between Run1 and Run2:

Run1 / Run2	Starter					Drive
	Contactor	Starter	Reverser	Speed	Soft start	
Run1	Close	Run	RunFwd	RunLow	RunRamp1	RunFwd
Run2	No Action	No Action	RunRev	RunHigh	RunRamp2	RunRev

Run1 and Run2 trigger:

Run1	Run2	Trigger event	Run type
0	0	Stop	No Action
0 ➝ **1**	0	Run	Run1
0	0 ➝ **1**	Run	Run2
0 ➝ **1**	0 ➝ **1**	No Action	No Action
1	1	No Action	No Action
1 ➝ **0**	1	Run	Run2
1	1 ➝ **0**	Run	Run1

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.3.3 AC Drive Object (42 / 0x2A)

The "AC Drive Object" describes the device-specific functions of the drive, e.g. speed ramps, torque control etc.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	NumAttr	USINT	Number of supported attributes
2	Get	Attributes	ARRAY of USINT	Listing of the supported attributes
3	Get	AtReference	BOOL	1: The drive currently runs at reference speed or reference torque (depending on attribute 6, DriveMode).
4	Get / Set	NetRef	BOOL	• 0: Reference via local setting on the device or terminal • 1: Reference via network (e.g. from the scanner)
6	Get / Set	DriveMode	USINT	Drive mode: • 1: Idle speed (frequency) • 3: Torque control In order to be able to use the torque control for the **assembly object instances 22 (0x16), 23 (0x17), 72 (0x48), 73 (0x49)**, this attribute has to be written. ▸ Write "DriveMode" attribute (📖 154)
7	Get	SpeedActual	INT	Current speed [rpm/$2^{SpeedScale}$]
8	Get / Set	SpeedRef	INT	Reference speed [rpm/$2^{SpeedScale}$]
11	Get	TorqueActual	INT	Current torque [Nm/$2^{TorqueScale}$]
12	Get / Set	TorqueRef	INT	Reference torque [Nm/$2^{TorqueScale}$]
22	Get / Set	SpeedScale	SINT	From inverter drive version V13.00 and communication module version V01.02! Speed scaling factor [Nm/$2^{SpeedScale}$] Value range: -128 ... 127

Attribute ID	Service	Name	Data type	Value
24	Get / Set	TorqueScale	SINT	Torque scaling factor [Nm/$2^{TorqueScale}$] Value range: -128 ... 127
29	Get / Set	RefFromNet	BOOL	Status of reference speed / reference torque • 0: Reference via local setting on the device or terminal • 1: Reference via network (e.g. from the scanner)

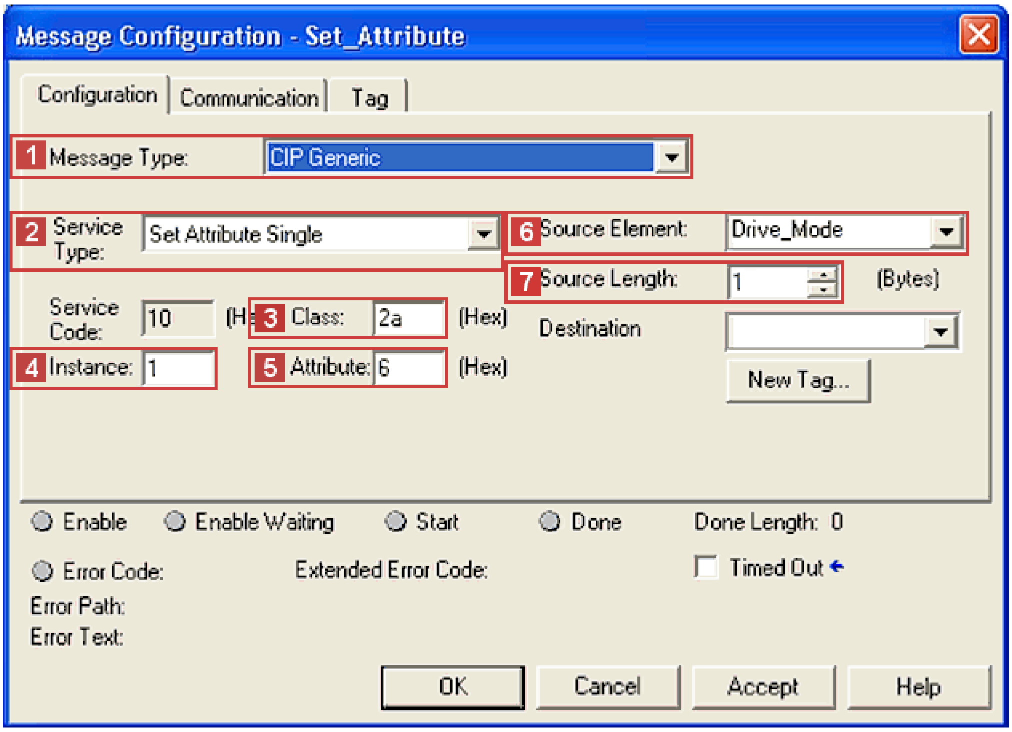
Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.3.4 Write "DriveMode" attribute

In order to be able to use the torque control for the **Assembly-Objektinstanzen 22 (0x16), 23 (0x17), 72 (0x48), 73 (0x49)**, the "DriveMode" attribute has to be written by means of explicit message transfer.

In order to write the "DriveMode" attribute by means of explicit message transmission, the following settings are required:



Settings		Value / description
1	**Message Type**	"CIP Generic"
2	**Service Type**	"Set Attribute Single" (service code "0x10")
3	**Class**	"2A" (AC Drive Object)
4	**Instance**	"1"
5	**Attribute**	"6" ("DriveMode" attribute)
6	**Source Element**	"Drive_Mode" (variable in the PLC program used as data source for writing.)
7	**Source Length**	"1 byte" (The variable data type is SINT.)

14.4 Lenze objects

14.4.1 Lenze Class (101 / 0x65)

The "Lenze Class (101 / 0x65)" enables the access to the adjustable error responses to EtherNet/IP errors which can be set in code C13880.

Note

The attributes of this class are described in the EDS file. Using the »RSNetWorx« Rockwell software, the attributes can thus be directly set under "Parameters" in the properties dialog of the EtherNet/IP node.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	No. of supported Attributes	USINT	6 (0x0006)
2	Get	Attribute List	USINT	1 (0x0001) ... 6 (0x0006)
3	Get / Set	Reaction on Idle Mode	USINT	Corresponds to the value in C13880/1: • 0 = no response • 1 = error • 4 = warning locked
4	Get / Set	Reaction on Fault Mode	USINT	Corresponds to the value in C13880/2: • 0 = no response • 1 = error • 4 = warning locked
5	Get / Set	Reaction on Expl. Msg. TO	USINT	Corresponds to the value in C13880/3: • 0 = no response • 1 = error • 4 = warning locked
6	Get / Set	Reaction on I/O Timeout	USINT	Corresponds to the value in C13880/4: • 0 = no response • 1 = error • 4 = warning locked

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.4.2 Lenze Class (103 / 0x67)

The "Lenze Class (103 / 0x67)" provides the image of the scanner input data.

The input data for the scanner are provided at the **MCI_OUT** interface of the communication module and are sent to the scanner by means of the assembly object instance **111 (0xE6)**.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	No. of supported Attributes	USINT	3 (0x0003)
2	Get	Attribute List	USINT	1 (0x0001) ... 3 (0x0003)
3	Get	I/O image of produced data	USINT	Image of the scanner input data

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.4.3 Lenze Class (104 / 0x68)

The "Lenze Class (104 / 0x68)" provides the image of the scanner output data.

The output data of the scanner are sent by means of the assembly object instance **110 (0xE5, Custom Output)** and are provided at the **MCI_IN** interface of the communication module.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	No. of supported Attributes	USINT	3 (0x0003)
2	Get	Attribute List	USINT	1 (0x0001) ... 3 (0x0003)
3	Get	I/O image of consumed data	USINT	Image of the scanner output data

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.4.4 Lenze Class (110 / 0x6E)

The "Lenze Class (110 / 0x6E)" enables read or write access to Lenze codes.

The Lenze code must be specified as an "instance" (corresponding code no. 1 ... 65535) and its subcodes must be specified as "attributes".

Note!

- If the corresponding Lenze code does not have a subcode, the value '0' must be entered in the attribute. If '0' is not supported as attribute value by the engineering tool used, the value '1' must be entered.
- A display code cannot be configured using "Set_Attribute_Single".

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
0 ... 255	Get / Set	Lenze Subcode number	Data type of the subcode	Value of the subcode

Supported service codes

Service code	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

E84AYCEO communication module (EtherNet/IP™) · Communication Manual · EDS84AYCEO · 13443009 · DMS 3.0 EN · 10/2013 · TD17

Lenze Drives GmbH
Postfach 10 13 52
D-31763 Hameln
Germany
+49 (0)51 54 / 82-0
+49 (0)51 54 / 82-28 00
Lenze@Lenze.de
www.Lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
00 80 00 / 24 4 68 77 (24 h helpline)
+49 (0)51 54 / 82-11 12
Service@Lenze.de

Accessories

PROFINET

E84AYCER

Inverter Drives 8400 ________________________

Communication manual EN

13564915

Lenze

Contents

1 About this documentation

Contents

This documentation exclusively describes the E84AYCER communication module (PROFINET).

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **hardware manual "Inverter Drives 8400"**.

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

This documentation also contains...

- Safety instructions that must be observed
- The most important technical data of the communication module
- Information about the versions of the Lenze standard devices to be used
- Notes on troubleshooting and fault elimination

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

Depending on the software version of the inverter and the »Engineer« software installed, the screenshots in this documentation may deviate from the »Engineer« representation.

This documentation does not describe the software of other manufacturers. No responsibility is taken for corresponding information given in this documentation. Information on how to use the software can be obtained from the documents of the host system (master).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information on PROFINET can be found on the homepage of the PROFIBUS user organisation which also develops the PROFINET communication technology:

www.profibus.com

Target group

This documentation addresses to persons who configure, install, commission, and maintain the networking and remote maintenance of a machine.

Tip!

Information and software updates for Lenze products can be found in the download area at:

www.Lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

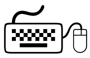
Extension module	Type designation	From hardware version	From software version
PROFINET communication module	E84AYCER	VA	02.00

1.1 Document history

Version			Description
3.0	02/2019	TD23	General revision
2.0	11/2010	TD17	• General revision • Update for SW version 02.00
1.0	04/2010	TD17	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal	Normal spelling	Example: 1234
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Text		
Program name	» «	PC software Example: Lenze »Engineer«
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Hyperlink	Underlined	Optically highlighted reference to another topic. Can be activated with a mouse-click in this documentation.
Icons		
Page reference	(🕮 8)	Optically highlighted reference to another page. Can be activated with a mouse-click in this documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Inverter	Lenze inverters of the "Inverter Drives 8400" product series for which the communication module can be used.
Standard device	Application as directed (12)
»Engineer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
Code	Parameter which serves to parameterise and monitor the inverter. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). This term is also referred to as "subindex" in common parlance.
Lenze setting	This setting is the default factory setting of the device.
Basic setting	
HW	Hardware
SW	Software
I/O controller	PROFINET master The I/O controller takes over the master function for data communication of the decentralised field devices. The I/O controller usually is the communication interface of a PLC.
I/O device	PROFINET slave
IO supervisor	Engineering and diagnostics tools The IO supervisor can access process data, diagnostic data, and alarm data.

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

- Lenze drive and automation components ...
  - must only be used as directed.
    ▶ Application as directed (📖 12)
  - must never be commissioned if they display signs of damage.
  - must never be technically modified.
  - must never be commissioned if they are not fully mounted.
  - must never be operated without required covers.
  - during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.
- The following applies to Lenze drive components ...
  - only use the accessories approved.
  - Only use original manufacturer spare parts.
- Observe all specifications contained in the enclosed documentation and related documentation.
  - This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
    ▶ Product features (📖 13)
  - The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.
- All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...
  - are familiar with installing, mounting, commissioning, and operating the product.
  - who have the corresponding qualifications for their work.
  - who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- Always use a separate power supply unit safely separated according to EN 61800-5-1 ("SELV/PELV") in every control cabinet for external voltage supply.
- Only use cables corresponding to the given specifications.
  ▶ Ethernet cable specification (📖 30)

Documentation for the standard device, control system, system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

- If the Inverter Drives 8400 are operated on a phase-earthed mains with a rated mains voltage of ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.
  ▶ Protective insulation (📖 16)

Device protection

- The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
  ▶ Installation (📖 22)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

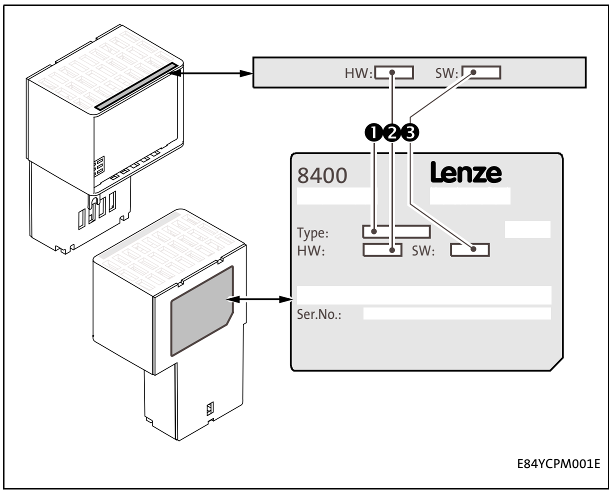
Product series	Type designation	From software version
Inverter Drives 8400 StateLine	E84AVSCxxxxx	05.00
Inverter Drives 8400 HighLine	E84AVHCxxxxx	05.00
Inverter Drives 8400 TopLine	E84AVTCxxxxx	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- should only be used in PROFINET networks.

Any other use shall be deemed inappropriate!

3.2 Identification

Type designation and hardware and software version of the communication module are indicated on the nameplate:



1 Type designation (type)

E84	Product series
A	Version
Y	Module identification: Extension module
C	Module type: Communication module
ER	PROFINET
V/S	V: Coated version S: Standard version

2 Hardware version (HW)

3 Software version (SW)

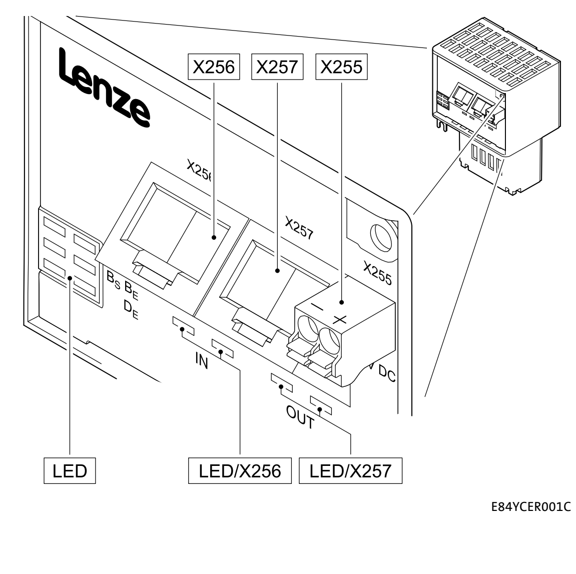
[3-1] Identification data

3.3 Product features

- Interface module for the PROFINET IO communication system to the expansion slots of the Inverter Drives 8400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- Support of the I&M0...4 functionality for the identification of the standard device
- Automatic detection of the baud rate 100 Mbps
- A line topology is enabled by the integrated 2-port switch.
- Support of the LLDP protocol for the topology recognition
- Support of the SNMP protocol for diagnostic purposes
- Support of the MRP (Media Redundancy Protocol)
  The Inverter Drive 8400 can be integrated into a ring topology as a client node.
- Access to all Lenze parameters
- An online connection via PROFINET can be established using the Lenze »Engineer«.

3.4 Connections and interfaces

- 2 RJ45 sockets for the PROFINET connection
- 2-pole plug connector for the external voltage supply of the communication module.
- Front LEDs for diagnosing the ...
  - voltage supply of the communication module;
  - connection to the standard device;
  - PROFINET connection;
  - PROFINET activity.



X255	External voltage supply of the communication module • 2-pin plug connector with spring connection ▸ External voltage supply (📖 32)
X256	PROFINET input (IN)
X257	PROFINET output (OUT) • RJ45-sockets • each with 2 LED status displays for diagnostics ▸ Network topology (📖 26) ▸ PROFINET connection (📖 28) ▸ Status displays at X256 and X257 (📖 76)
MS **ME** **BS** **BE** **EN**	5 LED status displays for diagnostics ▸ Module status displays (📖 74) ▸ Fieldbus status displays (📖 75)

[3-2] E84AYCER communication module (PROFINET)

4 Technical data

4.1 General data and operating conditions

Range	Values	
Order designation	E84AYCER	
Communication profile	PROFINET	
Communication medium	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e	
Interface for communication	RJ45: Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)	
Network topology	Tree, star, and line	
Type of node	I/O device with real time (RT) communication properties	
Number of device nodes	Max. 255 in the subnetwork	
Max. cable length	100 m	
PNO identification number	0x0106	
Device identification (Device ID)	0x8400	
TCP port	8410	
Baud rate	100 Mbps	
Switching method	"Store and forward"	
Switch latency	Approx. 125 µs at max. frame length	
Voltage supply	External supply via separate power supply unit • "+": U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %), I = 140 mA • "-": Reference potential for external voltage supply	
Conformities, approvals	CE	2004/108/EC, EMC Directive 2006/95/EC, Low-Voltage Directive
	EAC	Eurasian conformity TR CU: Technical Regulations of Customs Union
		TP TC 004/2011 (TR CU 004/2011) About the safety of low voltage equipment
		TP TC 020/2011 (TR CU 020/2011) Electromagnetic compatibility of technical means
	UL	UL 508C / C22.2 No. 14, Power Conversion Equipment (File-No. E132659)

Hardware manual "Inverter Drives 8400"

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 Protective insulation

Danger!

Dangerous voltage

If the Inverter Drives 8400 are operated on a phase-earthed mains with a rated mains voltage of ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

- Death or severe injuries

Protective measures:

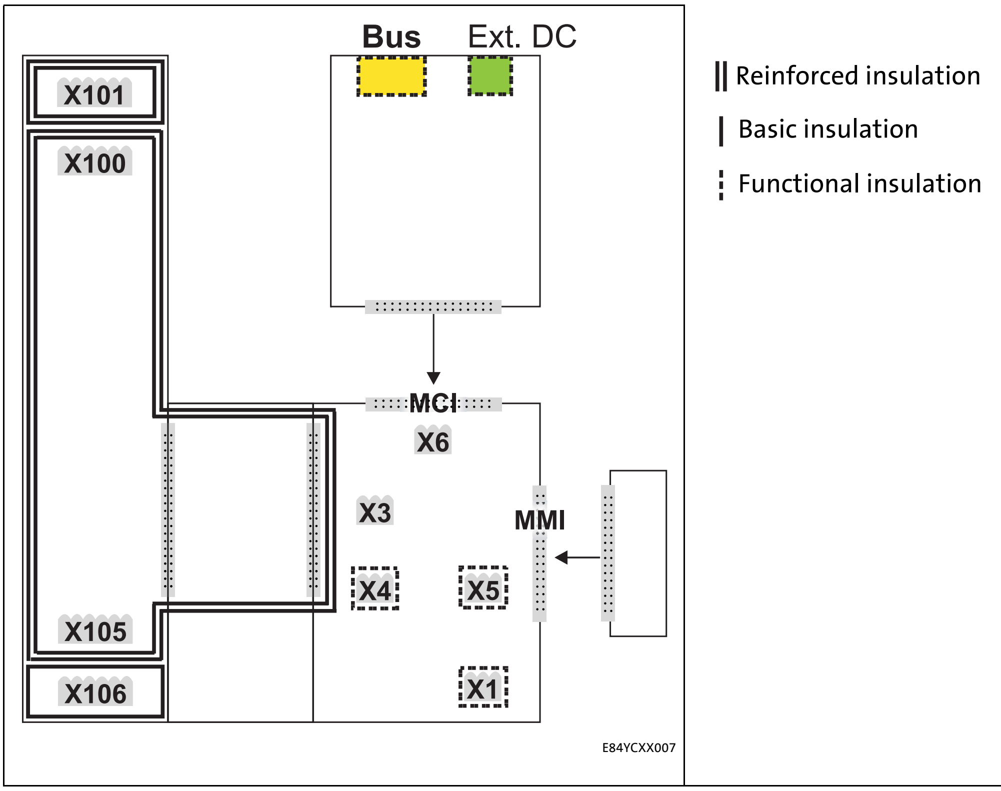
- If protection against accidental contact is required for the control terminals of the inverter and the connections of the device modules plugged, ...
  - a double isolating distance must exist.
  - the components to be connected must be provided with the second isolating distance.

Note!

The existing protective insulation in the Inverter Drives 8400 is implemented according to EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Inverter Drive 8400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection
X100	Mains/DC-bus connection
X101	Relay contact
X105	Motor/brake resistor
X106	Motor PTC
X1	System bus (CANopen)
X3	Analog inputs/outputs
X4	Digital outputs
X5	Digital inputs
X6	Diagnostics
MCI	Slot for the communication module
MMI	Slot for the memory module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MCI and the X100 mains terminal?

- The separate potential area with the better protective insulation is decisive.
  - The separate potential area of the bus terminal of the device module has a "functional insulation".
  - The separate potential area of the mains terminal has a "reinforced insulation".
- Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.3 Protocol data

Range	Values
Process data words (PCD)	Max. 16 process data words (max. 32 bytes)
Acyclic parameter data channel	Limited by the PROFINET frame size

4.4 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in the PROFINET network depend on the ...

- processing time in the inverter;
- frame runtime (baud rate / frame length);
- nesting depth of the network.

Processing time inside the inverter

Data	Processing time	
Process data	Approx. 2 ms + 0 ... 1 ms + 1 ... x ms	Update cycle Processing time in the module Runtime of the application task of the technology application used (tolerance)
Parameter data	Approx. 30 ms + a tolerance of 20 ms (typically) • For some codes, the processing time may be longer (see software manual/»Engineer« online help for the Inverter Drive 8400).	

There are no interdependencies between parameter data and process data.

4.5 Internal switch latency

The integrated 2-port switch causes runtime delays which can be calculated as follows:

Runtime delay = ((36 permanent bytes + process data in bytes) x 8 x 10 nsec) + 4 µsec

Example :

20 process data words + 4 PROFIsafe words => 48 bytes

- ((36 permanent bytes + 48 bytes) x 8 x 10 nsec) + 4 µsec
- (84 bytes x 8 x 10 nsec) + 4 µsec
- 6.72 µsec + 4 µsec = **10.72 µsec**

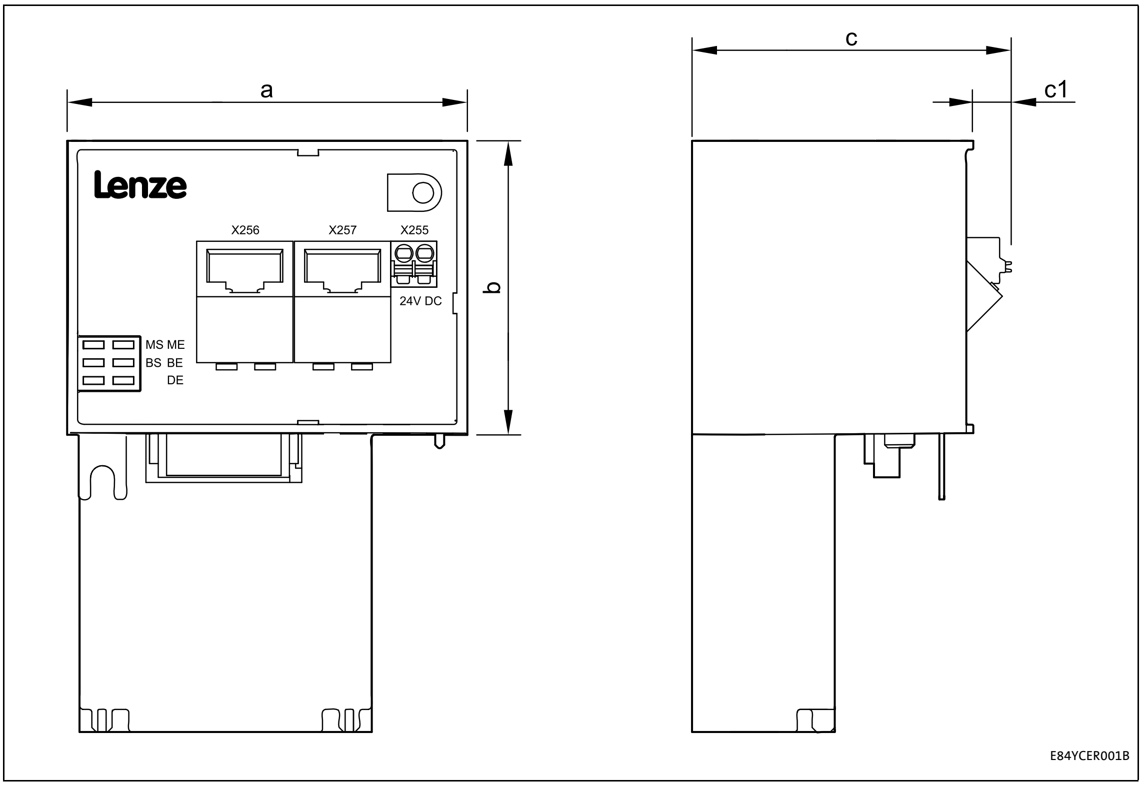
In accordance with the PROFINET specification, the shortest PROFINET I/O frame must have a data length of 72 bytes. If the 36 permanent bytes are subtracted from the 72 bytes, 36 bytes are available for process data. If now less than 36 bytes of process data are used, the PROFINET I/O frame is filled with "zero bytes" until it can be transmitted. As a consequence for the calculation formula, the shortest PROFINET I/O frame with 18 process data words (36 bytes) has always the same length and thus the runtime delay is the same, too.

Note!

The use of external switches can also lead to runtime delays. Depending on the system constellation, it may be useful to create a star topology or a line/mix topology.

▸ Network topology (📖 26)

4.6 Dimensions



[4-2] Dimensions

Type	Dimensions [mm]			
	a	b	c	c1
E84AYCER	67	50	57	8

5 Installation

Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Fieldbus communication is not possible or faulty.

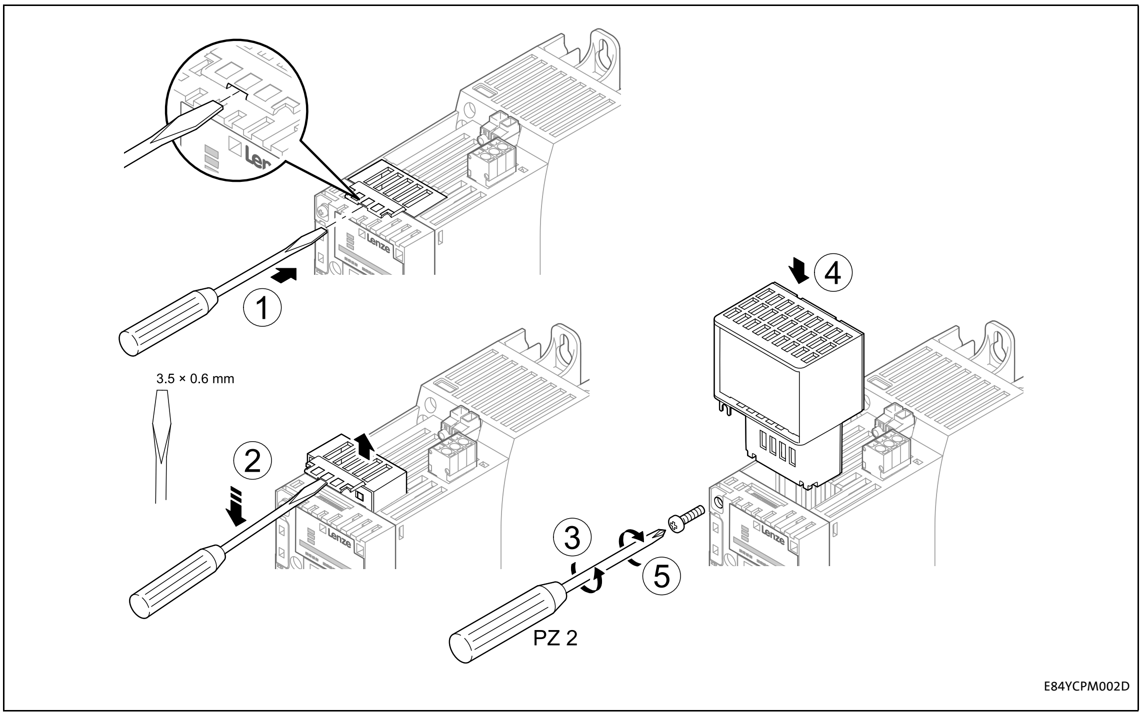
Protective measures

- Before touching the module, be sure that you are free of electrostatic charge.

5.1 Mechanical installation

The communication module can be plugged in or unplugged from the MCI slot when the inverter is switched on. When the module is plugged in, it is detected automatically, and a function and version plausibility check is executed.

5.1.1 Mounting for standard devices of 0.25 kW and 0.37 kW

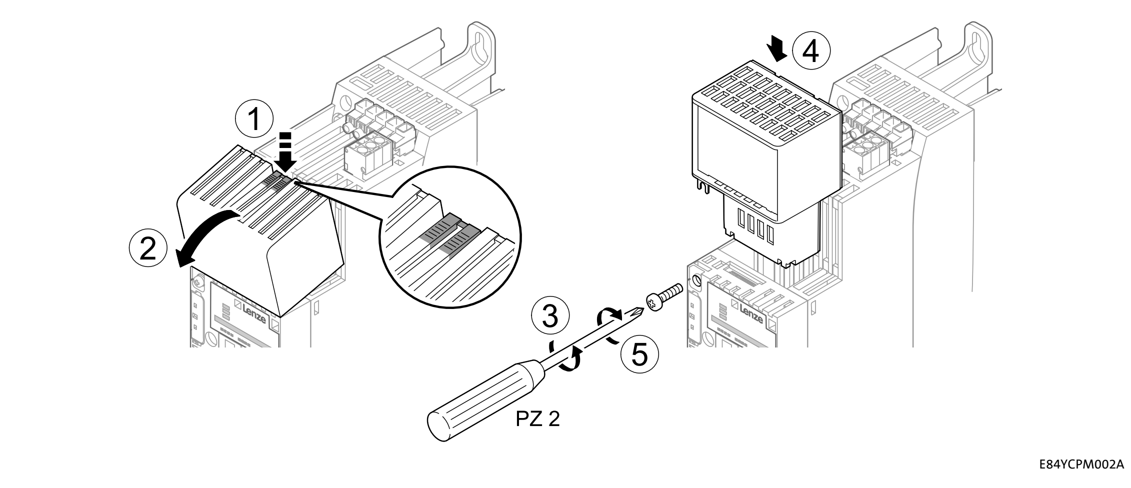


[5-1] Mounting for standard devices of 0.25 kW and 0.37 kW

Mounting steps

1. Use a screwdriver to lever out the cover of the MCI slot of the standard device and remove it (1, 2).
2. Loosen the securing screw for the communication module at the standard device (3).
3. Insert the communication module into the MCI slot of the standard device (4).
4. Tighten the securing screw again (5).

5.1.2 Mounting for standard devices of 0.55 kW or more

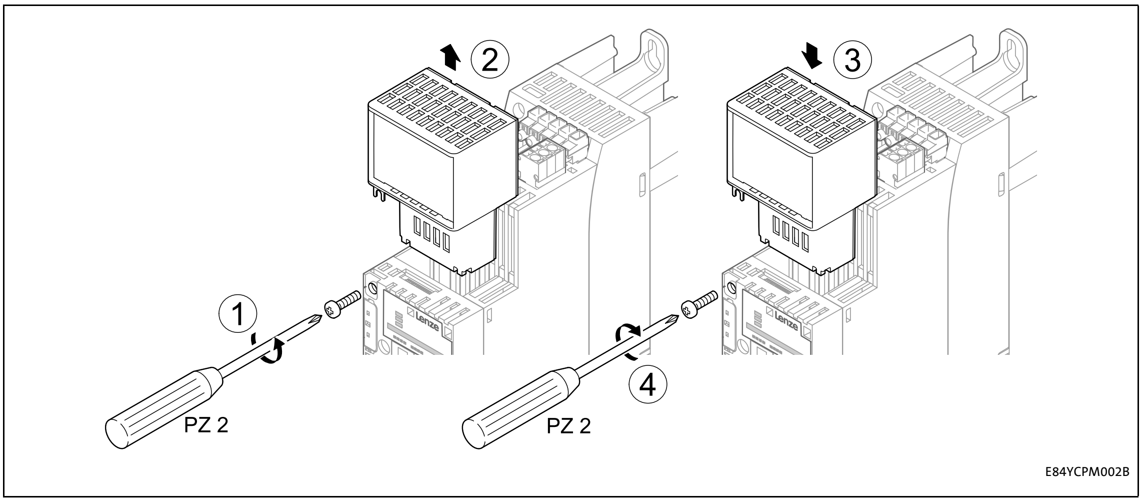


[5-2] Mounting for standard devices of 0.55 kW or more

Mounting steps

1. Slightly press on the marked spot on the top of the MCI slot cover of the standard device (1).
2. Tilt the cover forward and remove it from the standard device (2).
3. Loosen the securing screw for the communication module at the standard device (3).
4. Insert the communication module into the MCI slot of the standard device (4).
5. Tighten the securing screw again (5).

5.1.3 Exchanging the communication module



[5-3] Exchanging the communication module

Mounting steps

1. Loosen the securing screw for the communication module at the standard device (1).
2. Pull the communication module out of the MCI slot of the standard device (2).
3. Insert the new communication module into the MCI slot of the standard device (3).
4. Tighten the securing screw again (4).

5.2 Electrical installation

Documentation for the standard device, control system, system/machine
Observe the notes and wiring instructions contained in this documentation.

5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. Remove the plastic sheath of the cable at a length of 2 cm.
2. Fasten the cable shield to the shield support of the standard device.

5.2.2 Network topology

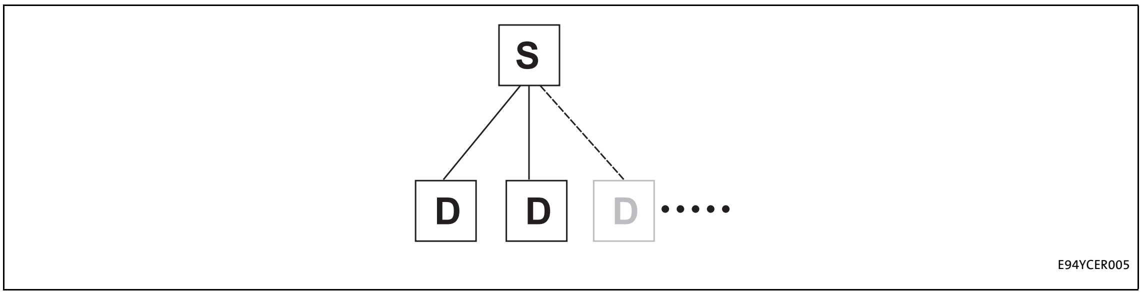
It is typical of PROFINET to have a rather free topology the limiting factor of which is large message latencies due to e.g. switches connected in series.

▸ Internal switch latency (📖 20)

The combination of a line and a stub is useful for system wiring.

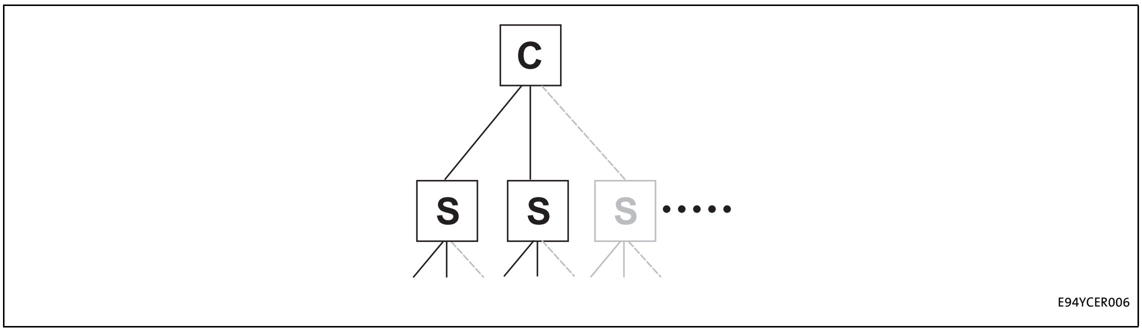
PROFINET supports the following topologies:

- Switch / star



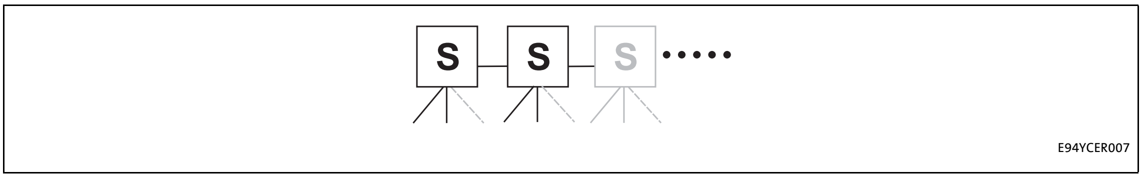
[5-4] Switch / star topology (S = switch, D = I/O device)

- Tree via switches



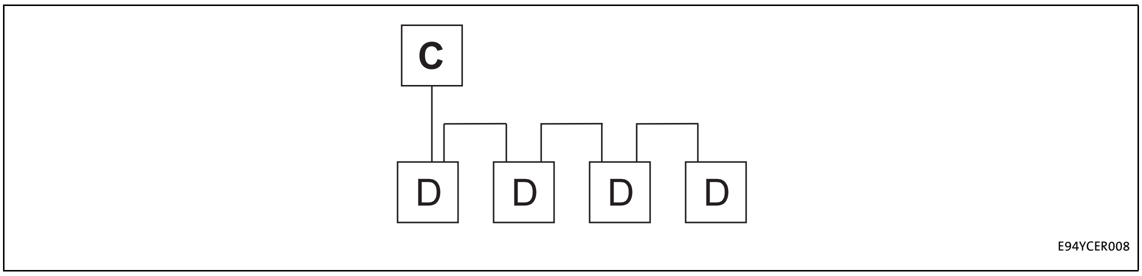
[5-5] Tree topology (C =I/O controller, S = switch)

- Switch / switch



[5-6] Switch/switch topology (S = switch)

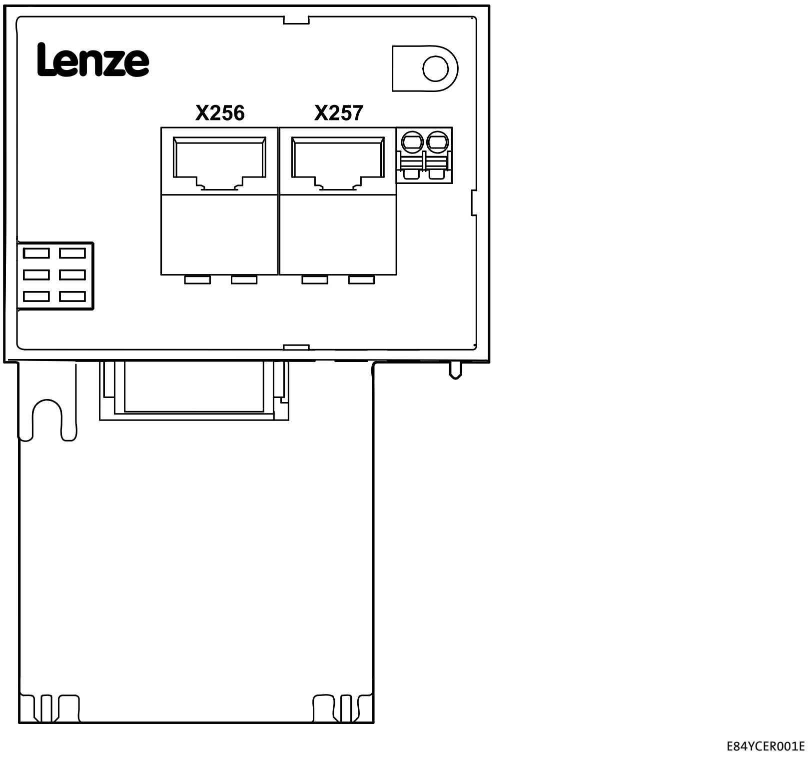
- I/O controller / I/O device



[5-7] Line topology (C = I/O controller, D = I/O device)

5.2.3 PROFINET connection

PROFINET is connected via the RJ45 sockets **X256** (IN) and **X257** (OUT).



[5-8] PROFINET connections X256 (IN) and X257 (OUT)

For connection of the communication module to the PROFINET fieldbus, a standard Ethernet patch cable is suitable.

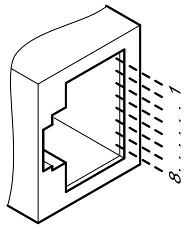
▶ Ethernet cable specification (📖 30)

The installation and removal of the Ethernet cables is optimised for the use of connectors in accordance with the "Automation Initiative of German Domestic Automobile Manufacturers" (AIDA).

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment of the RJ45 sockets

RJ45 socket	Pin	Signal
	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Tip!

The PROFINET interfaces feature an auto-MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite PROFINET interface and irrespective of the type of cable used (standard patch cable or crossover cable).

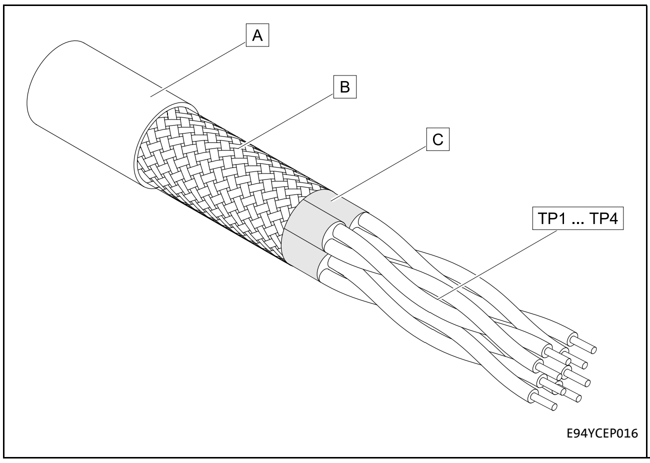
5.2.4 Ethernet cable specification

Note!

Only use cables that meet the listed specifications.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair), ISO/IEC 11801 or EN 50173, CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 ... TP4 Twisted core pairs 1 ... 4
▸ Colour code of the Ethernet cable (📖 31)

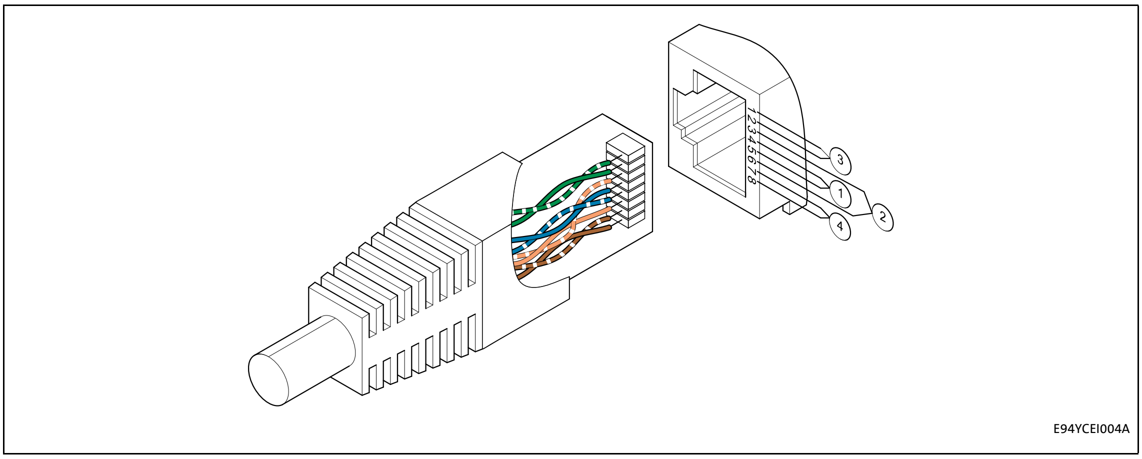
[5-9] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-10] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	White / Green	White / Orange
	2	Tx -	green	orange
2	3	Rx +	White / Orange	White / Green
1	4		blue	blue
	5		White / Blue	Blue / White
2	6	Rx -	orange	green
4	7		White / Brown	White / Brown
	8		brown	brown

5.2.5 External voltage supply

The communication module can be externally supplied with voltage via separate supply cables at the 2-pin plug connector **X255**.

Note!

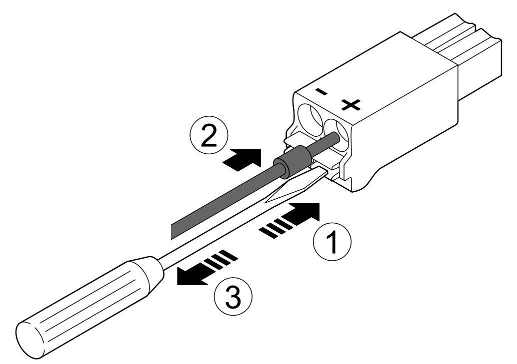
Always use a separate power supply unit safely separated according to EN 61800-5-1 ("SELV/PELV") in every control cabinet for external voltage supply.

- External voltage supply of the communication module is required if the communication via the bus should be maintained when the supply of the standard device fails.
- Access to parameters of a standard device disconnected from the mains is not possible.

Wiring of the X255 plug connector

Stop!

Only wire the plug connector if the standard device is disconnected from the mains.



[5-11] Wiring of the 2-pin plug connector with spring connection

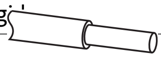
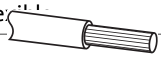
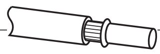
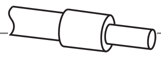
How to wire the plug connector with spring connection:

1. Press a screwdriver into the notch below the terminal and keep it pressed.
2. Place the supply cable in the terminal.
3. Remove the screwdriver from the notch.

Assignment of the X255 plug connector

Name	Description
+	U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %) I = 140 mA
-	Reference potential for the external voltage supply

Terminal data

Range	Values	
Electrical connection	2-pin plug connector with spring connection	
Possible connections	Rigid:	
		0.2 ... 1.5 mm^2 (AWG 24 ... 16)
	Flexible:	
		Without wire end ferrule 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, without plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, with plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
Stripping length	10 mm	

6 Commissioning

During commissioning, system-related data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. For Lenze devices, this is done via the codes.

The codes of the inverter and communication are saved non-volatilely as a data set in the memory module.

In addition, there are codes for diagnosing and monitoring the stations.

▶ Parameter reference (📖 85)

6.1 Before initial switch-on

Stop!

Before switching on the standard device together with the communication module, check the entire wiring for completeness, short-circuit, and earth fault.

6.2 Configuring the PROFINET IO controller

To enable communication with the inverter, you have to configure the IO controller first.

Configuration for device control

For the configuration of PROFINET, the current PROFINET device description file (XML) of the E84AYCER communication module (PROFINET) has to be imported in the IO controller.

You can find the **GSDML-Vx.z-Lenze-8400PNabb-yyyymmdd.xml** device description file in the Download area at:

www.Lenze.com

Wildcards in the "GSDML-Vx.z-Lenze-8400PN100-yyyymmdd.xml" file name	
x	Main version of the GSDML scheme used
z	Subversion of the GSDML scheme used
a	Major version of the software version
bb	Minor version of the software version
yyyy	Year
mm	Month
dd	Day

Defining the user data length

The user data length is defined during the initialisation phase of the I/O controller.

The Inverter Drives 8400 support the configuration of max. 16 process data words (max. 32 bytes).

Description of the device data base file

Selection text	Parameter data (with consistency)	Process data (with consistency)	Assigned I/O memory
PCD (**n**W) AR cons.	-	**n** words	**n** words
n = 1 ... 16 process data words			

Example of selecting the device data base file

- "PCD (8W) AR cons." = 8 process data words (only in slot 1 of the PROFINET telegram)

Tip!

A detailed description of consistency is given in the chapter "Consistent parameter data" (📖 70).

6.3 Setting the station name

Note!

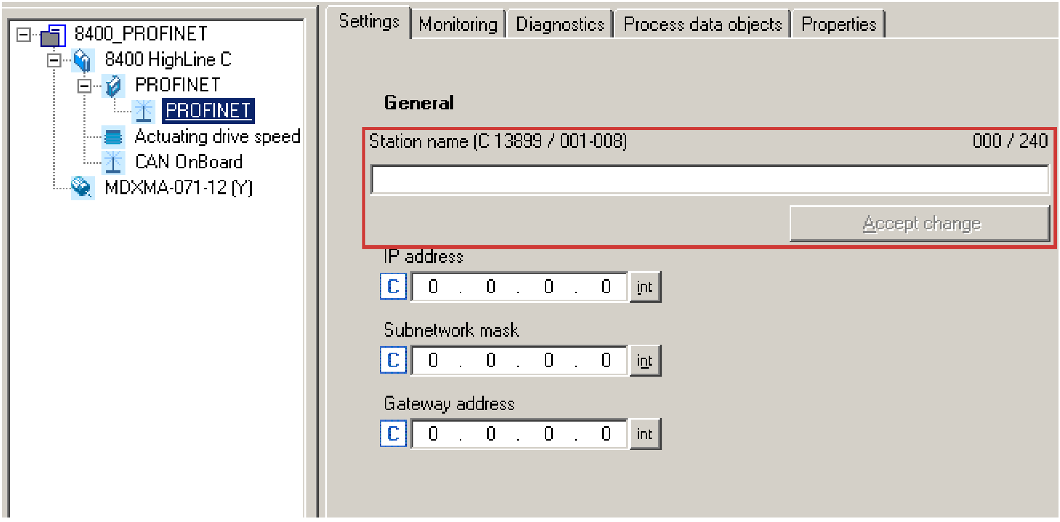
- The "Node blinking test" PROFINET function by means of which an accessible device can be identified is supported. During execution of the function, the red LED **ME** (Module status displays (📖 74)) blinks.
- Operation on the PROFINET requires a valid station name.
- In the case of impermissible settings, the red **BE** (Fieldbus status displays (📖 75)) blinks and the error message PROFINET: Station name error [0x01bc6532] (📖 83) is output. The communication module then continues to work internally with the name deleted.
- If the station name is assigned by the IO controller via PROFINET or the PROFINET configurator of the »Engineer«, changes will be effective immediately.

The station name currently used is shown in code C13864.

The station name ...

- is required for unambiguous addressing of the Inverter Drive 8400 by the I/O controller.
- can either be assigned by the I/O controller via PROFINET or set manually in the »Engineer«.
- has to be allocated in accordance with the PROFINET specification:
  - 1 or several labels separated by ".".
  - Max. length per label: 63 characters
  - Max. total length: 240 characters
  - Permissible characters: [a ... z], [0 ... 9], [.], [-]
  - Labels must not begin or end with [-].
- Prohibited syntax:
  - "n.n.n.n" (n = 0 ... 999)
  - "port-xyz" (x, y, z = 0 ... 9)
  - "port-xyz-abcde" (a, b, c, d, e, x, y, z = 0 ... 9)

In the »Engineer« the station name is set under the **Settings** tab.



- Then click **Accept change**. The station name is saved and written to code C13899.
- In the Lenze setting a deleted name is displayed. The name is also deleted if the "Reset to factory defaults" command is executed by an IO supervisor or an I/O controller.

How to activate changed settings in the »Engineer«:

1. Execute device command **C00002 = "11: Save all parameter sets"**.
2. Carry out a "reset node" of the node, or switch the voltage supply of the communication module off and on again

6.4 Setting the IP configuration

The IP configuration is required for addressing the Inverter Drive 8400 if communication between the PC/»Engineer« or the IO controller and the inverter is to be established via PROFINET. This requires the allocation of an IP address, subnet mask, and gateway address.

If no PROFINET network or IO controller is available yet, you can allocate the IP address, subnet mask, and gateway address for the communication module by using the following options:

- Settings via the PROFINET configurator of the »Engineer« (📖 40)
- Setting via codes in the »Engineer« (📖 42)

Note!

- If the IP parameters are assigned by the IO controller via PROFINET or the PROFINET configurator of the »Engineer«, changes become effective immediately and are saved with mains failure protection.
- The assignment of invalid combinations of IP address, subnet mask, and gateway address can have the consequence that no connection to the PROFINET can be established.
- In the case of impermissible settings, the red LED **BE** (Fieldbus status displays (📖 75)) blinks and the error message PROFINET: IP address error [0x01bc6533] (📖 84) is output.

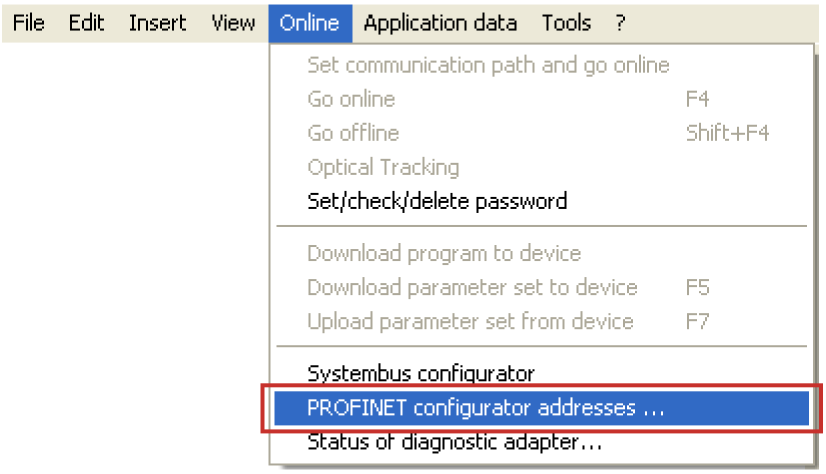
6.4.1 Settings via the PROFINET configurator of the »Engineer«

Note!

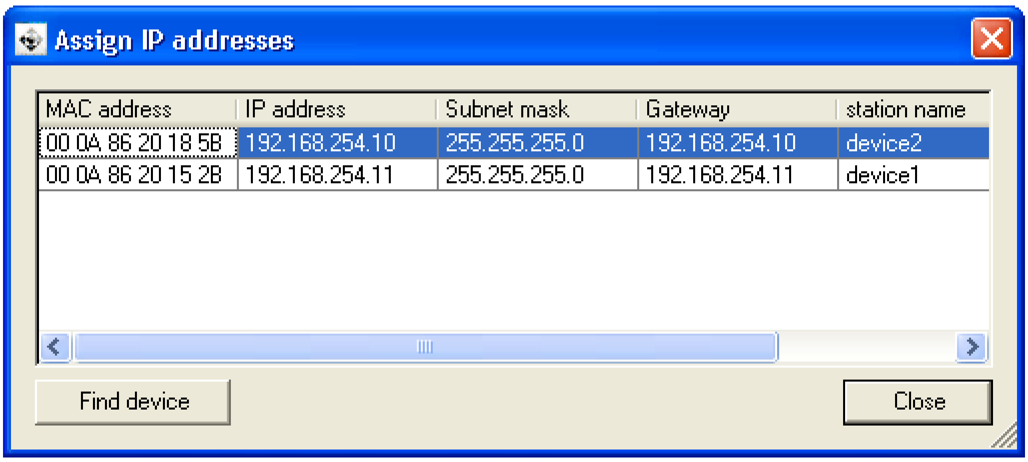
- The IP address must only be allocated manually in the »Engineer« if the PROFINET network is not actuated on the IO controller yet (IP address was not allocated by the IO controller yet).
- During the IP parameters are set in the »Engineer«, PROFINET communication with the IO controller must not take place at the same time.
- Changes will be effective immediately and are saved with mains failure protection.
- The parameter values currently used are shown in codes C13010 (IP address), C13011 (subnet mask), and C13012 (gateway address).

How to set the IP parameters via the PROFINET configurator:

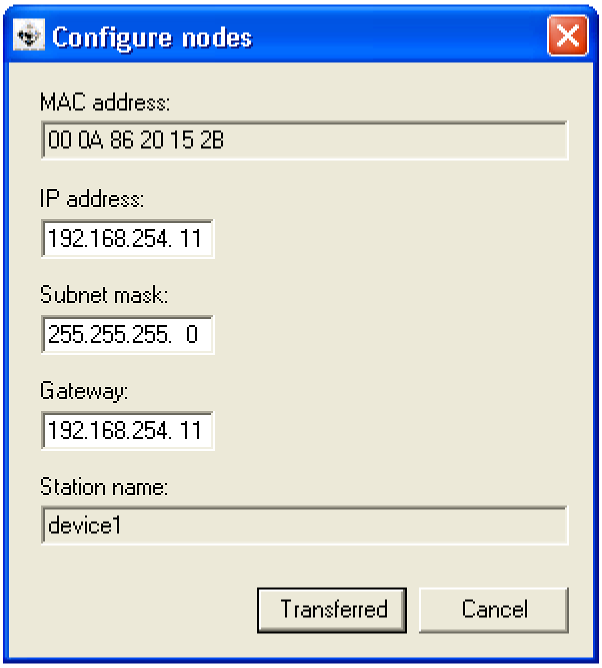
1. Execute the menu command **Online → Profinet configurator addresses....**



The **Assign IP addresses** dialog window is opened, and all Lenze PROFINET devices connected are listed.



2. By double-clicking on the individual IP parameters, you can set the IP configuration for each PROFINET node in the **Configure nodes** dialog window.



3. Click **Transferred**.
   - The IP configuration is transferred to the corresponding PROFINET node.
   - Changes in the IP parameters will become effective immediately.
   - The IP parameters are written to codes C13000 (IP address), C13001 (subnet mask) and C13002 (gateway address) of the communication module.

Tip!

By clicking the **Find device** button in the **Assign IP addresses** dialog window (see step 1), you can check whether the configuration was transferred successfully.

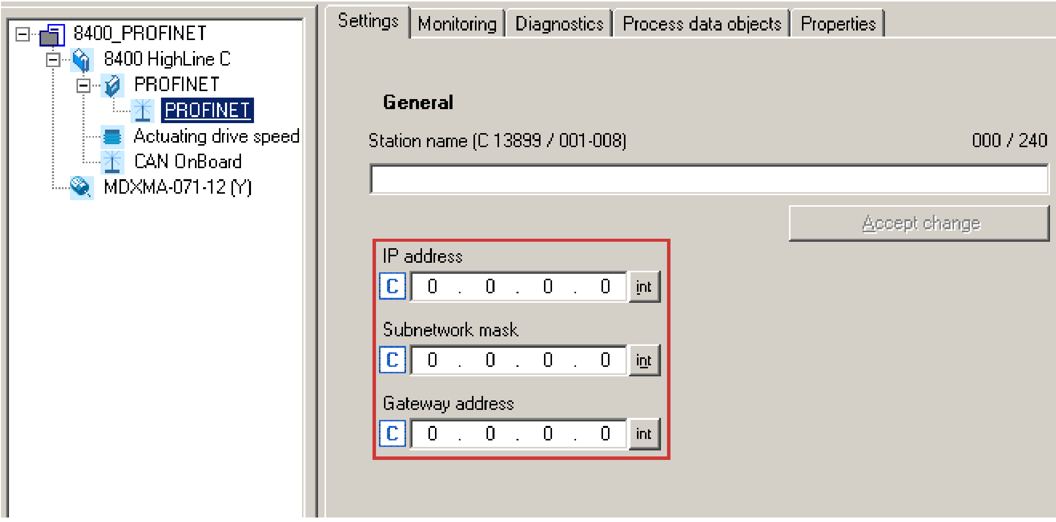
With device command **C00002 = "11: Save all parameter sets",** the current IP configuration is saved non-volatilely in the memory module.

6.4.2 Setting via codes in the »Engineer«

Note!

- The IP address must only be allocated manually in the »Engineer« if the PROFINET network is not operated on the IO controller yet (IP address has not been allocated by the IO controller yet).
- While setting the IP parameters in the »Engineer«, PROFINET communication with the IO controller must not take place at the same time.
- The parameter values currently used are shown in codes C13010 (IP address), C13011 (subnet mask), and C13012 (gateway address).

In the »Engineer« under the **Settings** tab you can set the IP parameters manually via code.



The IP parameters are written to codes C13000 (IP address), C13001 (subnet mask), and C13002 (gateway address).

How to activate changed settings in the »Engineer«:

1. Execute the device command **C00002 = "11: Save start parameters"**.
2. Carry out a "reset node" of the node, or switch the voltage supply of the communication module off and on again

Decimal representation of the IP parameters

By clicking the **[int]** buttons on the right next to the input fields, the IP parameters are represented as decimal values.

In the case of the decimal representation, the byte sequence is inverted.

Example: IP address 192.168.0.1

- C13000 = 16820416 [00000001.00000000.10101000.11000000$_{bin}$]

Byte 3								Byte 2								Byte 1								Byte 0							
1								0								168								192							
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	1	0	0	0	0	0	0
C13010/4								C13010/3								C13010/2								C13010/1							

IP address

- Valid IP addresses are defined in accordance with RFC 3330.
- The IP address is set/changed in C13000.
- In C13010/1...4 the IP address currently used is shown.

Example: Display of the IP address 192.168.0.1				
Code	C13010/1	C13010/2	C13010/3	C13010/4
Value	192	168	0	1

Subnet Mask

- The subnet mask indicates which part of the IP address is evaluated as net ID or host ID.
- Valid subnet masks are defined in accordance with RFC 1878
- The subnet mask is set/changed in C13001.
- In C13011/1...4 the subnet mask currently used is shown.

Example: Display of the subnet mask 255.255.255.0				
Code	C13011/1	C13011/2	C13011/3	C13011/4
Value	255	255	255	0

Gateway address

- The gateway address is valid if the network address of the IP address and the gateway address are identical.
- If the gateway address and the IP address are identical, gateway functionality is not used.
- DHCP is not supported.
- The gateway address is set/changed in C13002.
- In C13012/1...4 the gateway address currently used is shown.

Example: Display of the gateway address 192.168.0.1				
Code	C13012/1	C13012/2	C13012/3	C13012/4
Value	192	168	0	1

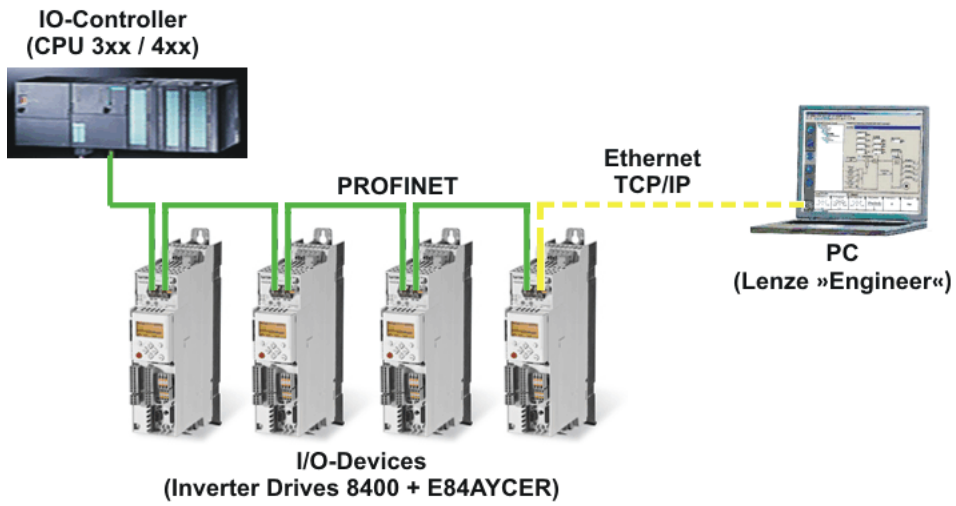
6.5 Establishing an online connection via PROFINET with the Lenze »Engineer«

This functionality is only supported **from software version V01.30.05**.

Note!

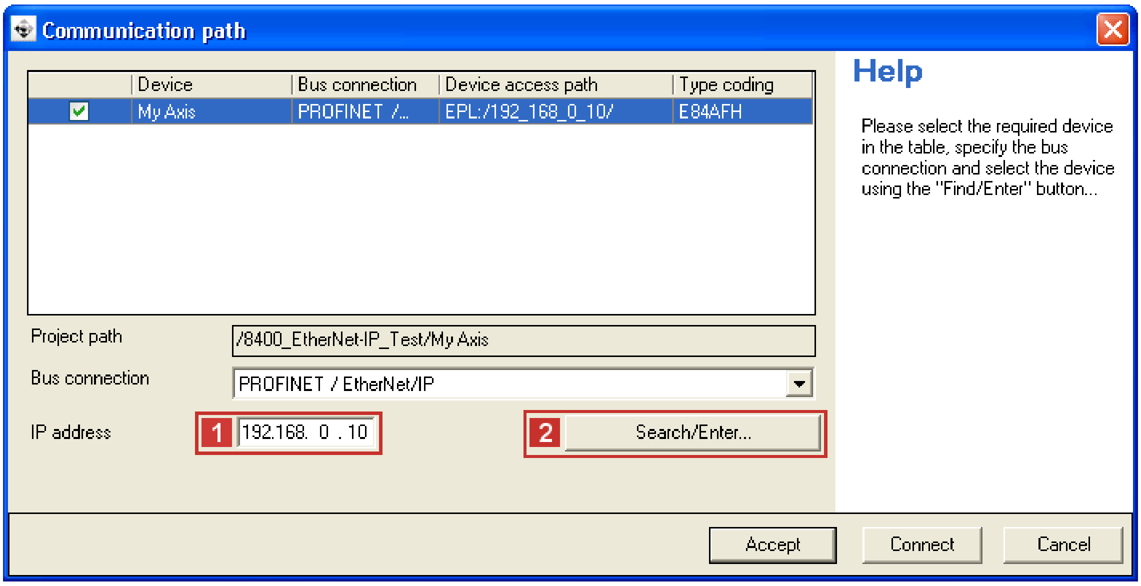
To ensure perfect operation of cyclic PROFINET communication, online access with the »Engineer« must be effected via a PROFINET switch.

The PROFINET switch integrated in the communication module can execute cyclic PROFINET communication prior to normal TCP/IP communication. In the case of PROFINET this is effected via the VLAN identification in the Ethernet frame.



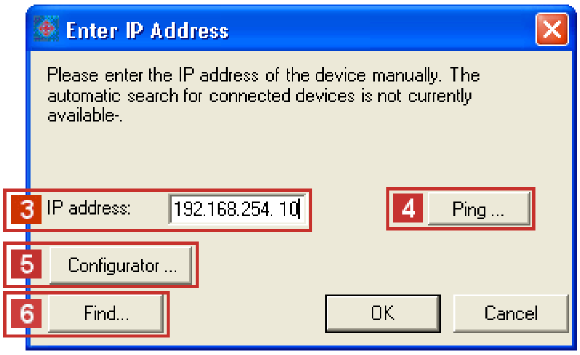
For an online connection between the »Engineer« and the inverter, the inverter must have an IP address (see Setting the IP configuration (📖 39)).

In the »Engineer« via the menu command **Online → Set communication path and go online**, you can select the PROFINET communication path. The PROFINET nodes previously configured are shown in the *Communication path* dialog window:



If the device access path is not configured correctly, the 1 IP address of the inverter selected in the display field can be entered manually here.

Via 2 **Search/Enter** you can establish a connection to devices which have not appeared in the display field. Corresponding settings for this can be made in the dialog window *Enter IP address*, which is shown:



Here you can enter an 3 **IP address** manually or execute the following actions using the buttons:

- Execute the console command 4 **Ping**.
- Assign the IP address via the 5 **Configurator**.
  - Settings via the PROFINET configurator of the »Engineer« (40)
- Select the device access path to the desired inverter by clicking 6 **Find**.

After having established the online connection, you can continue work with the »Engineer« as usual.

6.6 Initial switch-on

Documentation for the standard device

Observe the safety instructions and information on residual hazards.

Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

The restart protection is activated in the Lenze setting of the Inverter Drives 8400.

The restart behaviour of the inverter can be set via **C00142** ("auto-start option"):

- **C00142 = 9** (Lenze setting)
  - The inverter remains inhibited (even if the fault is no longer active).
  - Bit 0 (inhibit at power-on) and bit 3 (inhibit in the case of undervoltage) are set.
  - The drive restarts in a controlled mode through explicit controller enable: LOW-HIGH edge at digital input X4/RFR.
- **C00142 = 8** (Enabled)
  - In order to enable the device directly when switching it on, set bit 0 to zero (FALSE).
  - An uncontrolled restart of the drive is possible.

7 Data transfer

PROFINET transmits parameter data, configuration data, diagnostic data, alarm messages, and process data between the host system (IO controller) and the inverters that are part of the fieldbus (I/O devices). As a function of their time-critical behaviour, the data are transmitted via corresponding communication channels.

Communication channels

- The process data channel transmits process data.
  - The process data serve to control the inverter.
  - The transmission of process data is time-critical.
  - Process data are transmitted cyclically between the I/O controller and the I/O devices that are part of the fieldbus according to the Provider/Consumer model (continuous exchange of current input and output data).
  - The I/O controller can directly access the process data. In the PLC, for instance, the data are directly assigned to the I/O area.
  - Inverter Drives 8400 allow for a maximum of 16 process data words (16 bits/word) to be exchanged for each direction.
  - Process data are not saved in the inverter.
  - Process data are e.g. setpoints, actual values, control words, and status words.

Note!

Please observe the direction of the flow of information!
- Process input data (Rx data):
  - Process data from the inverter (IO device) to the IO controller
- Process output data (Tx data):
  - Process data from the IO controller to the inverter (IO device)

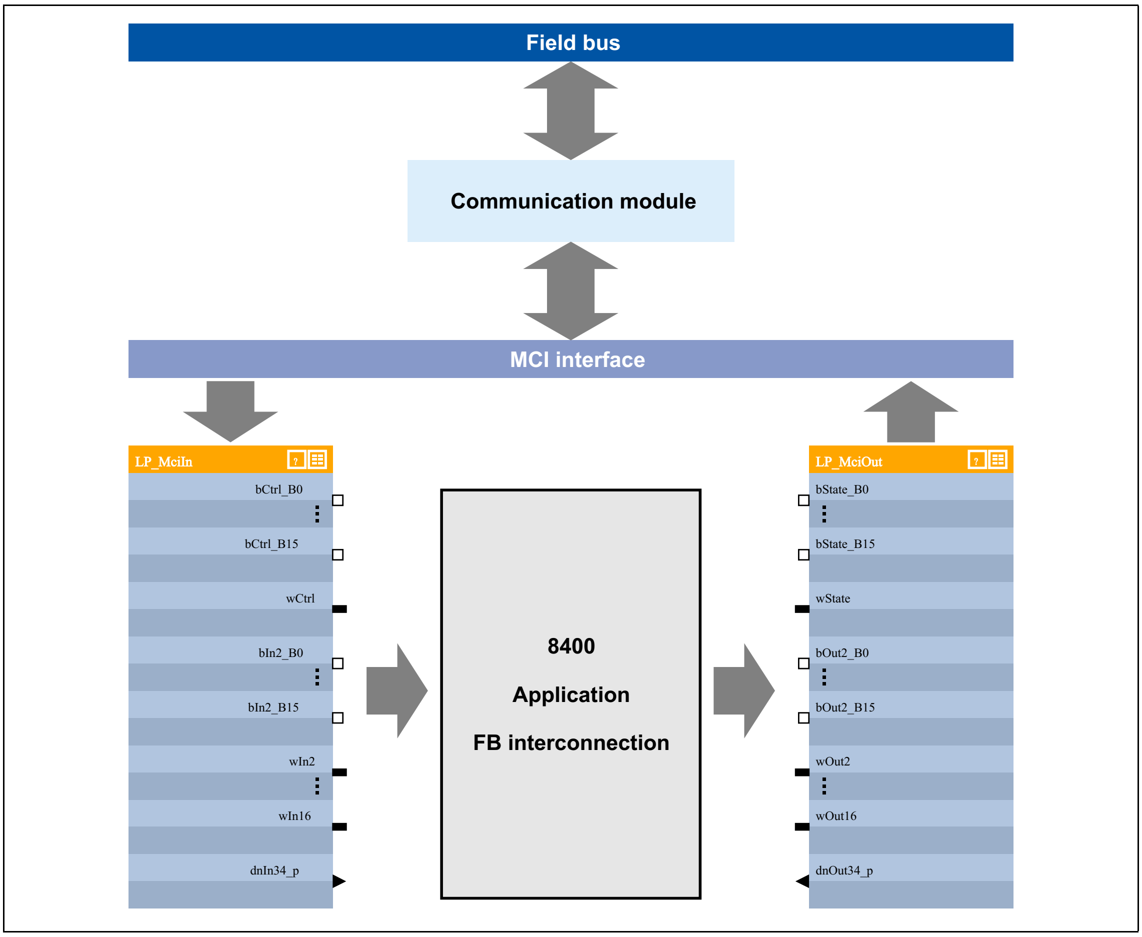
- Parameter data are transmitted via the acyclic channel.
  - The transmission of parameter data is usually not time-critical.
  - The access to the parameter data depends on the PROFIdrive profile.
  - Examples of parameter data are operating parameters, motor data, and diagnostic information.
  - The acyclic channel provides access to all Lenze codes.
  - Parameter data changes must be saved via code **C00002** of the Inverter Drives 8400.

8 Process data transfer

8.1 Access to process data / PDO mapping

Process data (MCI-PDOs) are transferred via the MCI interface.

- A maximum of 16 words are exchanged per direction.
- The process data are accessed via the port blocks **LP_MciIn** and **LP_MciOut**. The port blocks are also called process data channels.
- The port block **LP_MciIn** maps the received MCI-PDOs.
- The port block **LP_MciOut** maps the MCI-PDOs to be sent.
- The port/function block interconnection of the process data objects (PDO) takes place via the Lenze »Engineer«.



[8-1] External and internal data transfer between the bus system, inverter, and application

Software manual / »Engineer« online help for the Inverter Drive 8400

Here you will find detailed information on the port/function block interconnection in the »Engineer« and the port blocks.

Note!

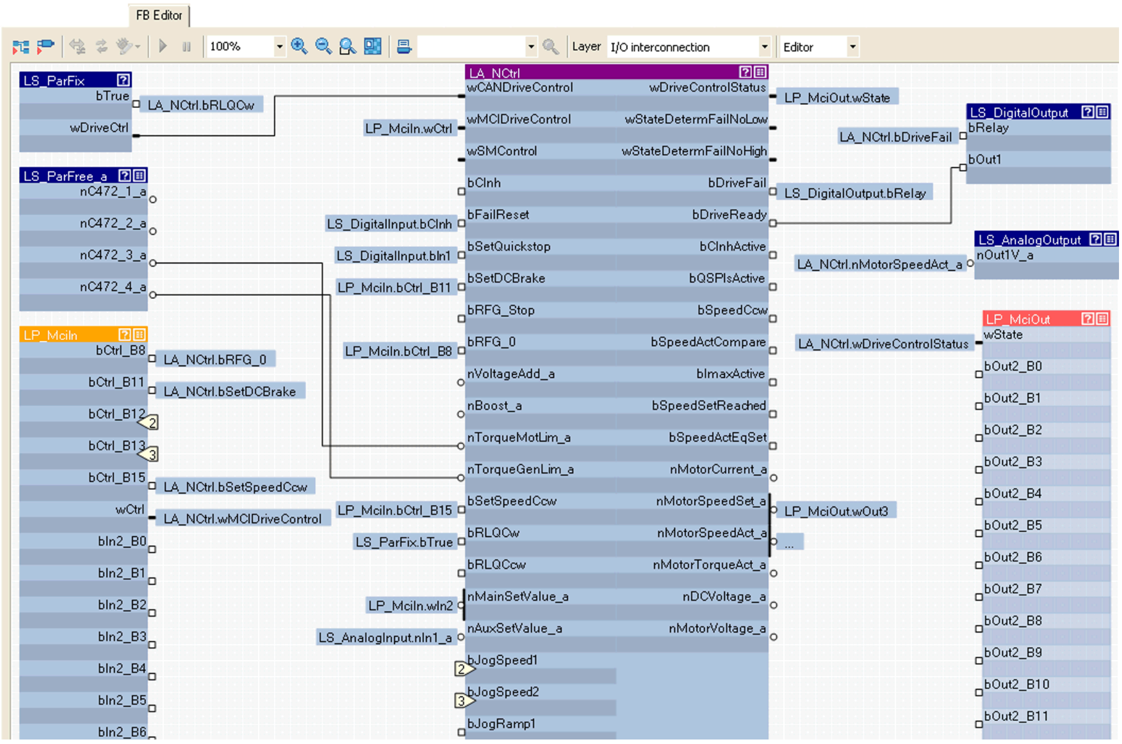
The »Engineer« screenshots shown on the following pages are only examples of the setting sequence and the resulting screens.

The data in the display fields may differ from the ones of your project.

8.2 Preconfigured port interconnection of the process data objects (PDO)

The preconfigured port interconnection of the process data objects can be activated by setting standard device code **C00007 = "40: MCI"**.

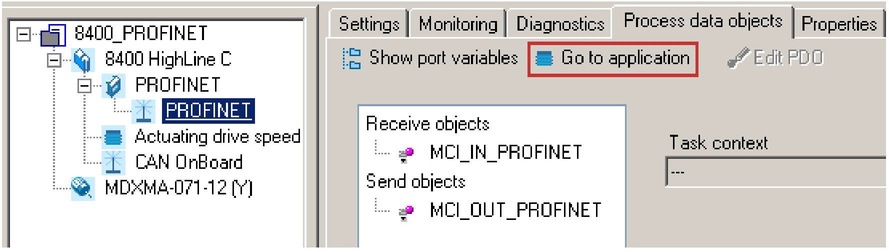
The function block editor (FB Editor) serves to display the port blocks "LP_MciIn" and "LP_MciOut" with the preconfigured interconnections:



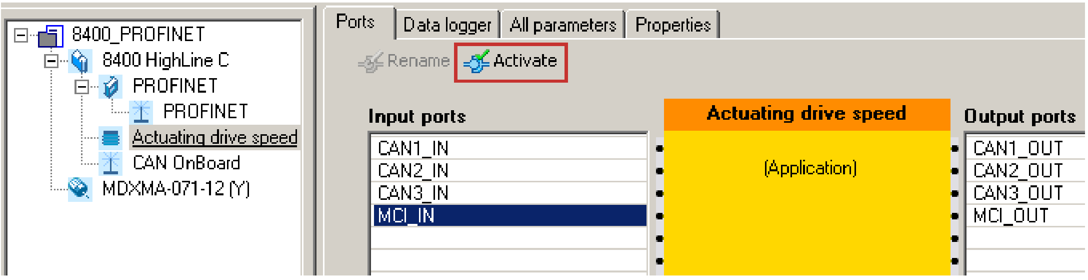
8.3 Freely configuring the port interconnection of the process data objects (PDO)

How to freely configure the port interconnection in the »Engineer«:

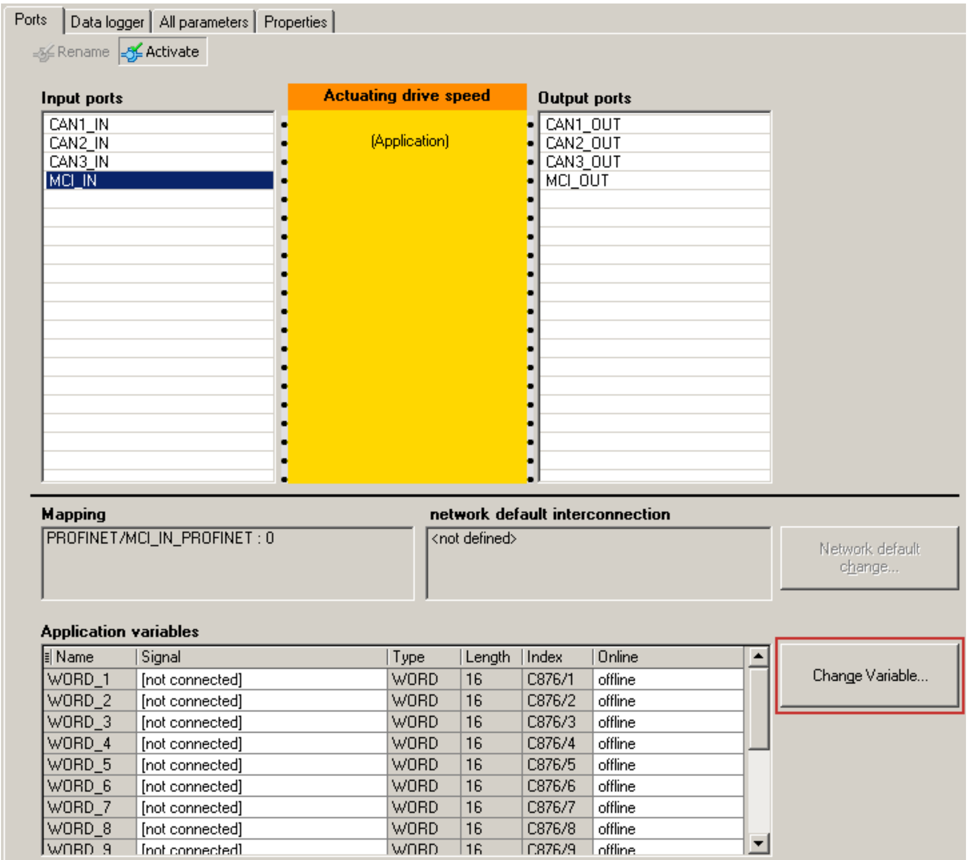
1. Go to the **Process data objects** tab and click **Go to application**.



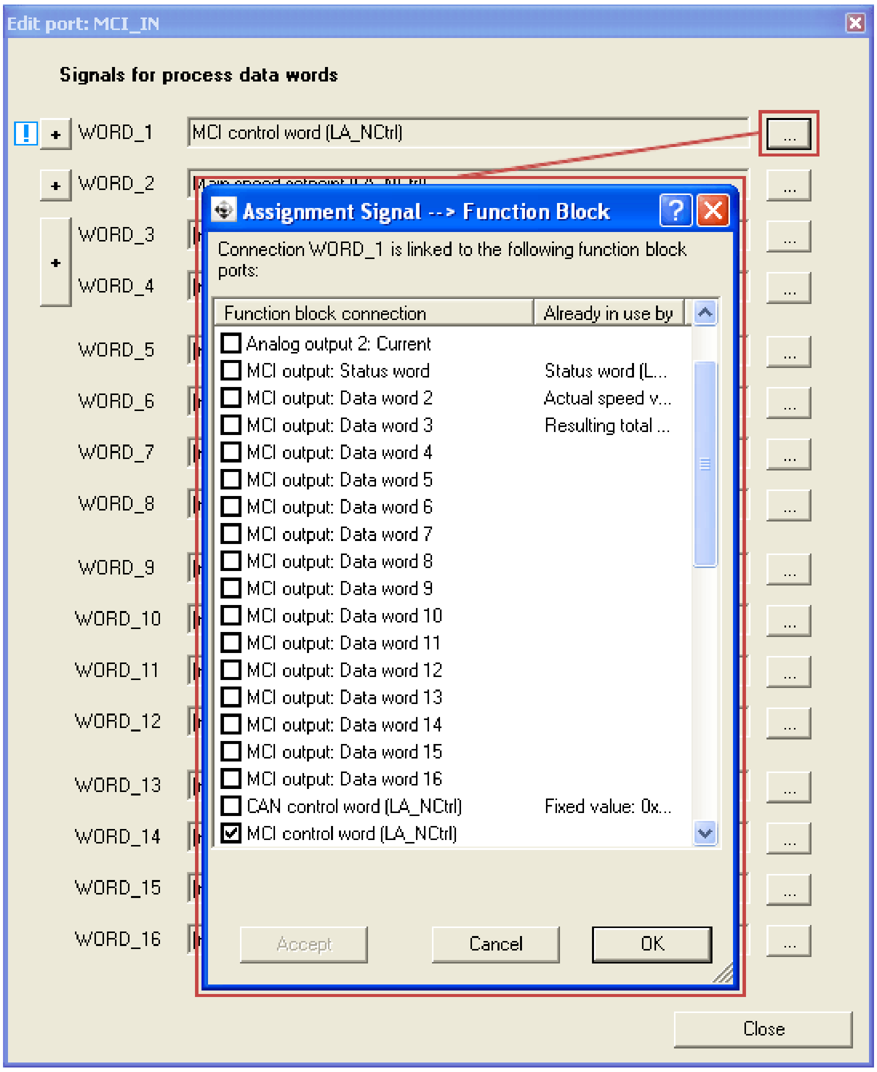
2. Go to the **Ports** tab, select the port blocks "MCI_IN" or "MCI_OUT" and click **Activate** to activate them.



3. Click the **Change Variable ...** button.

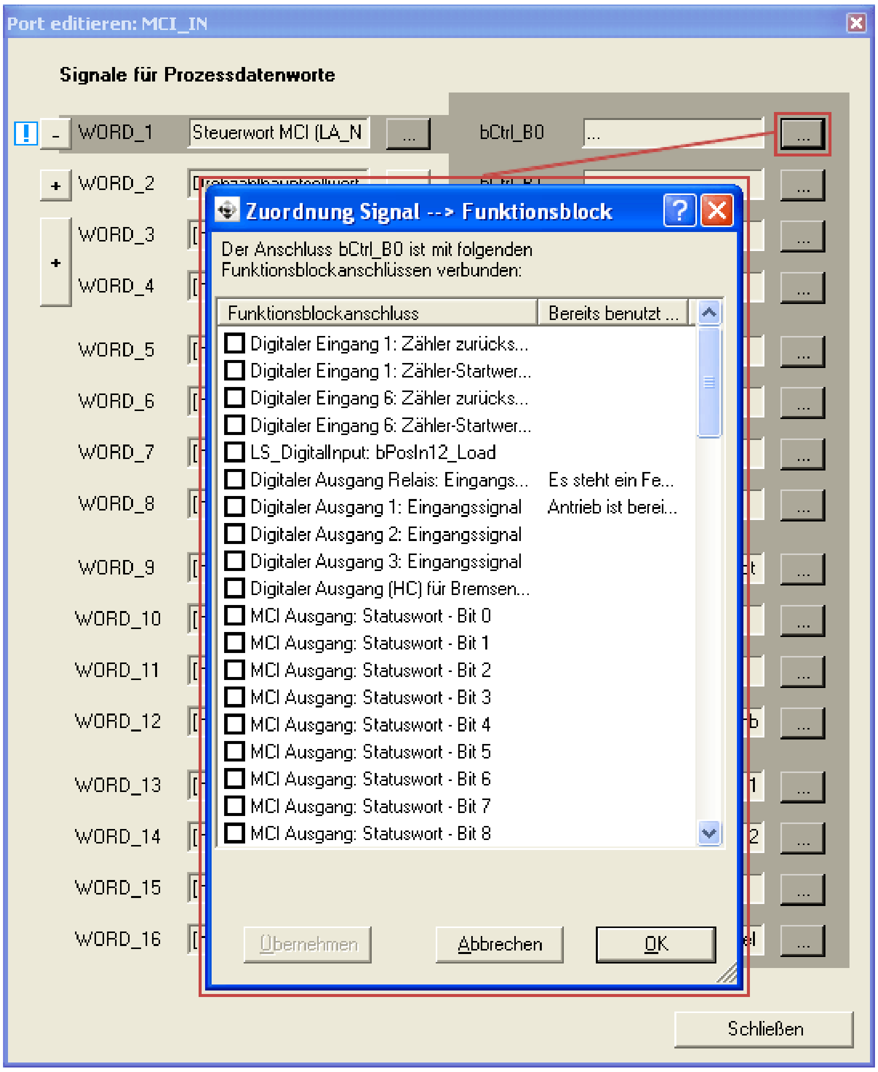


4. Via the [...] button, you can assign signals to the process data words in the *Assignment Signal --> Function Block* dialog window.
   ➞ Select the signals and then confirm the selection with **OK**.



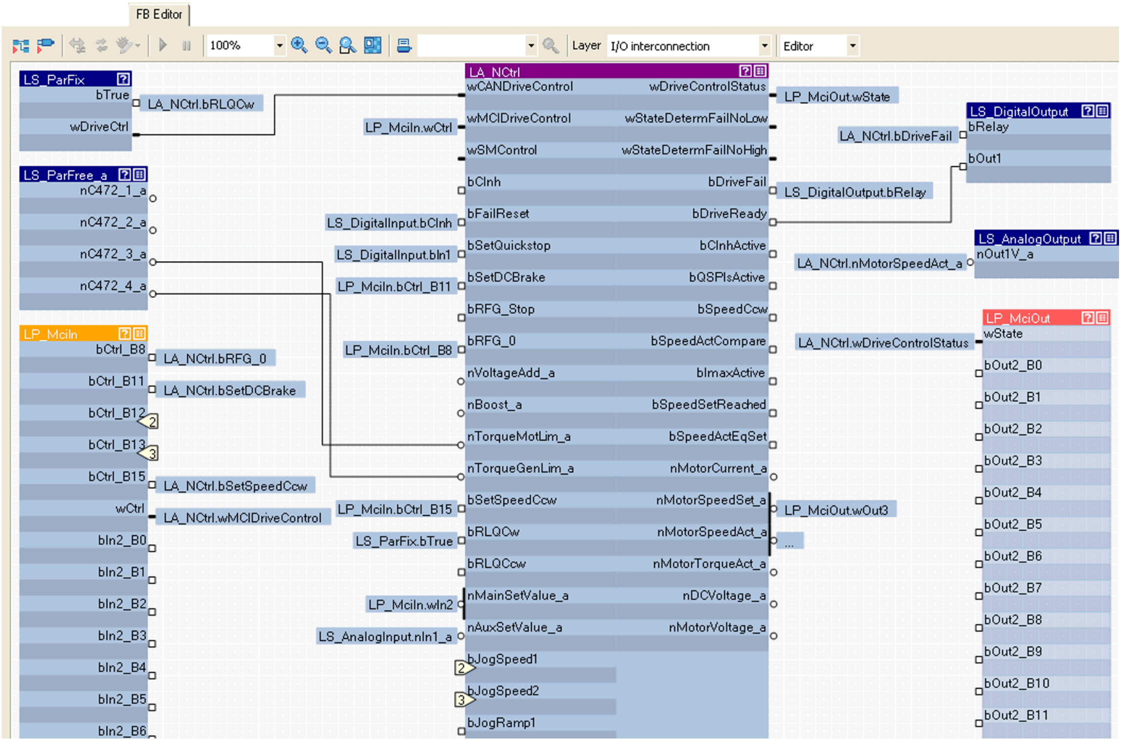
Moreover you can assign signals to the individual control and status bits at the WORD_1 and WORD_2 process data words via the [+] and [...] buttons.

→ Select the signals and then click **OK**.



Tip!

When the port blocks "LP_MciIn" and "LP_MciOut" are activated (see 1.), they will be visible in the FB Editor. Here you can also assign signals to the process data words.



9 Parameter data transfer

9.1 The acyclic channel (PROFIdrive profile)

An optional service extension is the acyclic parameter data transfer.

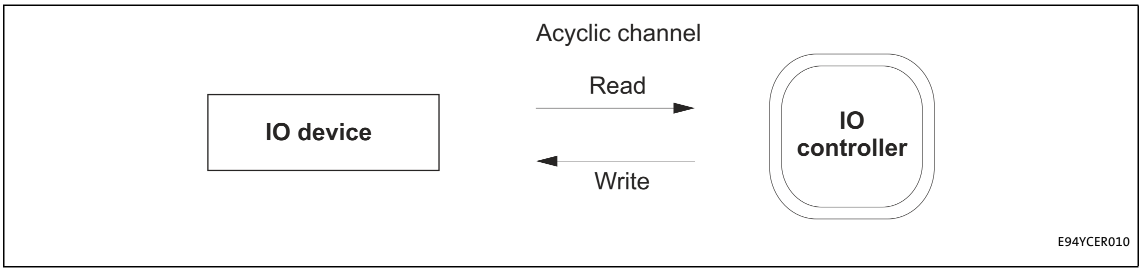
Cyclic and acyclic PROFINET services can be operated simultaneously in the network.

Product features

- There is always only one parameter request in process (no pipelining).
- No spontaneous messages are transferred.
- There are only acyclic parameter requests.
- Profile-specific parameters can be read independently of the I/O device state.

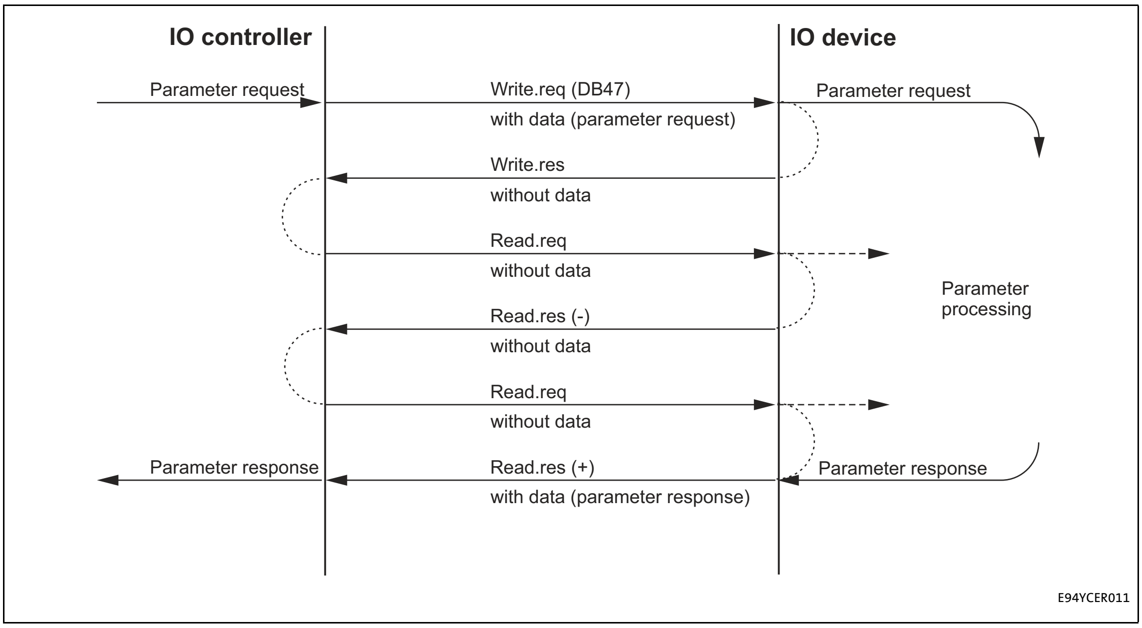
9.1.1 Connection establishment of an I/O controller to an I/O device

An I/O controller can always be used to request parameters from an I/O device if the I/O device is in the "DATA_EXCHANGE" state.



[9-1] Data communication via the acyclic channel

9.1.2 Acyclic data transmission process



[9-2] Data communication via the acyclic channel

- A "Write.req" is used to transmit the data set (DB47) in the form of a parameter request to the I/O device.
- "Write.res" confirms the receipt of the message by the I/O controller.
- The I/O controller requests the response of the I/O device with "Read.req".
- The I/O device responds with a "Read.res (-)" if processing is not yet completed.
- After parameter processing, the parameter request is completed by transmitting the parameter response in the form of a "Read.res (+)" to the I/O controller.

9.1.3 Structure of the PROFINET data frame

Dest Addr	Scr Addr	VLAN Day	Type 0800H	RPC	NDR	Read/Write Block	Data	FSC
6 bytes	6 bytes	4 bytes	4 bytes	80 bytes	64 bytes	64 bytes	0 ... 240 bytes	4 bytes

[9-3] PROFINET data telegram

In the "Read/Write Block", the initiator specifies the access to data set "DB47". The data which are written to this index or read by it contain a header and the parameter request or the parameter response. The read data or the data to be written are contained in the "Data" field.

The following subchapters describe the parameter request and the parameter response in detail.

PROFINET specification

Here you will find detailed information on the PROFINET data telegram.

Assignment of the user data depending on the data type

Depending on the data type used, the user data are assigned as follows:

Data type	Length	User data assignment				
		Byte 1	Byte 2	Byte 3	Byte 4	Byte ...
String	x bytes					
U8	1 byte		00			
U16	2 bytes	High byte	Low byte			
U32	4 bytes	High word		Low word		
		High byte	Low byte	High byte	Low byte	

9.2 Reading parameters from the inverter

Note!

- When a read request is processed, no parameter value is written to the I/O device.
- In the case of a multi-parameter read request, parameter attribute, index, and subindex are repeated "n" times, "n" being the number of parameters requested.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is specified by the I/O controller.
Request identification	U8	0x01: Request parameters for reading
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 or 0x01

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

9.2.1 Response to a correctly executed read request

Note!

- Responses to read requests do not contain parameter attributes and indices/subindices.
- When a multi-parameter read request is transmitted, the parameter format and the parameter value are repeated "n" times, "n" being the number of parameters requested.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x01: Parameter has been read
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	• 0x01 • Number of requested subindices. (If there is more than one subindex, only the parameter value is repeated.) • In the case of string codes, the number of characters is entered here.

Parameter value

Byte 7	Byte 8	Byte 9	Byte 10
Value			

Field	Data type	Values
Value	String	Any
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.2.2 Response to a read error

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x81: Parameter has not been read • The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7	Byte 8	Byte 9	Byte 10
Error code		Additional information (if available)	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF Error information (error) (68)
Additional information (if available)	U16	

9.2.3 Frame example: Read request

The heatsink temperature of the inverter is to be read.

- Code to be read: C00061
- Heatsink temperature: 43 °C

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x01	0x00	0x01
	Request parameters for reading		

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0xC2	0x00	0x00
Index = 24575 - code no. = 24575 - 61 = 24514 = 0x5F C2		No subindex	

Parameter response to a correctly executed read request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Response identification	Axis	Number of indices
0xXX	0x01	0x00	0x01
(mirrored)	Parameter has been read	(mirrored)	

Byte 5	Byte 6
Format	Number of values
0x43	0x01
Double word	1 value

Byte 7	Byte 8	Byte 9	Byte 10
Value			
High word: high byte	High word: low byte	Low- word: high byte	Low word: low byte
0x00	0x00	0x00	0x2B
Read value = 0x00 00 00 2B = 43 x 1 (internal factor) = 43 [°C]			

Parameter response to a read error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Response identification	Axis	Number of indices
0xXX	0x81	0x00	0x01
(mirrored)	Parameters not read	(mirrored)	

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Error	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning see the chapter "Error information (error)" (📖 68)	

9.3 Writing parameters to the inverter

Note!

- When a multi-parameter write request is processed, the parameter attribute, index, subindex, and then the parameter format and parameter value are repeated "n" times, "n" being the number of parameters requested.
- A parameter request must not exceed the maximum data length of 240 bytes.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is specified by the I/O controller.
Request identification	U8	0x02: Write parameter
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 or 0x01

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

Parameter format

Byte 11	Byte 12
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	• 0x01 • Number of requested subindices. (If there is more than one subindex, only the parameter value is repeated.) • In the case of string codes, the number of characters is entered here.

Parameter value

Byte 13	Byte 14	Byte 15	Byte 16
Value			

Field	Data type	Values
Value	String	Any
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.3.1 Response to a correctly executed write request

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x01: Parameter written
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

9.3.2 Response to a write error

Note!

In the case of a multi-parameter request, correct and possible faulty messages are summarised in one frame. They have the following data contents:

- Correct message
  - Format: 0x40
  - Number of values: 0x00
- Faulty message
  - Format: 0x44
  - Number of values: 0x01 or 0x02
  - Error code without additional information (number of values = 0x01) *or*
  - Error code with additional information (number of values = 0x02)

A faulty access to a parameter "n" is indicated at the nth position in the response frame of a multi-parameter request.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x82: Parameter has not been written • The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7	Byte 8	Byte 9	Byte 10
Error code		Additional information (if available)	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF ▶ Error information (error) (📖 68)
Additional information (if available)	U16	

9.3.3 Frame example: Write request

In the inverter, the deceleration time for quick stop is to be set to 50 ms.

- Code to be written: C00105

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x02	0x00	0x01
	Write parameters	Axis 0	1 index

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0x96	0x00	0x00
Index = 24575 - code no. = 24575 - 105 = 24470 = 0x5F 96		No subindex	

Byte 11	Byte 12
Format	Number of values
0x43	0x01
Double word	1 value

Byte 13	Byte 14	Byte 15	Byte 16
Value			
High word: high byte	High word: low byte	Low- word: high byte	Low word: low byte
0x00	0x00	0x00	0x32
Value to be written = 0.05 [s] x 1000 (internal factor) = 50 = 0x00 00 00 32			

Parameter response to a correctly executed write request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Response identification	Axis	Number of indices
0xXX	0x02	0x00	0x01
(mirrored)	Parameter has been written	(mirrored)	1 index

Parameter response after write error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Response identification	Axis	Number of indices
0xXX	0x82	0x00	0x01
(mirrored)	Parameter has not been written	(mirrored)	1 index

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Error	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning see the chapter "Error information (error)" (📖 68)	

9.4 Error information (error)

Error code	Meaning	Description	Additional information
0x0000	Impermissible parameter number	Access to unavailable parameter	-
0x0001	Parameter value cannot be changed	Change access to a parameter value that cannot be changed	Subindex
0x0002	Lower or upper value limit exceeded	Change access with value beyond the value limits	Subindex
0x0003	Faulty subindex	Access to unavailable subindex	Subindex
0x0004	No array	Access with subindex to non-indicated parameter	-
0x0005	Wrong data type	Change access with value that does not match the data type of the parameter	-
0x0006	No setting permitted (only resettable)	Change access with value unequal to 0 where this is not permitted	Subindex
0x0007	Description element cannot be changed	Change access to a description element that cannot be changed	Subindex
0x0008	Reserved	(PROFIdrive profile V2: PPO-Write requested in the IR is not available)	-
0x0009	Description data not available	Access to unavailable description (parameter value is available)	-
0x000A	Reserved	(PROFIdrive profile V2: Wrong access group)	-
0x000B	No parameter change rights	Change access without parameter change rights	-
0x000C	Reserved	(PROFIdrive profile V2: Wrong password)	-
0x000D	Reserved	(PROFIdrive profile V2: Text in the cyclic traffic cannot be read)	-
0x000E	Reserved	(PROFIdrive profile V2: Name in the cyclic traffic cannot be read)	-
0x000F	No text array available	Access to unavailable text array (parameter value is available)	-
0x0010	Reserved	(PROFIdrive profile V2: Missing PPO-Write)	-
0x0011	Request cannot be executed due to the operating state	Access is not possible due to temporary reasons not specified here	-
0x0012	Reserved	(PROFIdrive profile V2: Other error)	-
0x0013	Reserved	(PROFIdrive profile V2: date in the cyclic traffic cannot be read)	-
0x0014	Value impermissible	Change access with the value that is inside the value limits but not permissible for other permanent reasons (parameters with defined individual values)	Subindex
0x0015	Response too long	The length of the current response exceeds the maximum transmittable length	-
0x0016	Parameter address impermissible	Impermissible or non-supported value for attribute, number of subindices, parameter number, or subindex, or a combination	-
0x0017	Format impermissible	Write request: Impermissible or non-supported format of parameter data	-
0x0018	Number of values not consistent	Write request: Number of values of the parameter data do not match the number of subindices in the parameter address	-
0x0019	Reserved	-	-
...			
0x0064			

Error code	Meaning	Description	Additional information
0x0065	Manufacturer-specific	-	-
...			
0x00FF			

9.5 Consistent parameter data

In the PROFINET communication system, data are permanently exchanged between the host (CPU + I/O controller) and the standard device via the plugged-on I/O device interface module. The I/O controller and the CPU (central processing unit) of the host access a joint memory: the dual port memory (DPM).

- The DPM permits a data exchange in both directions (write/read):

It could happen that a slower I/O controller writing would be overtaken by a faster CPU reading within a cycle time without any further data organisation.

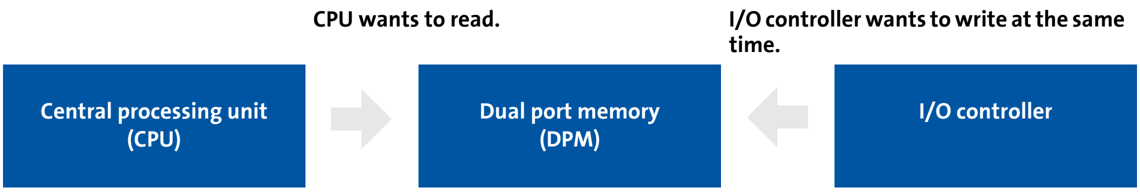
In order to avoid such an impermissible state, the parameter data to be transmitted must be marked as "consistent".

Data communication with consistent data

With consistency, either "reading" or "writing" is possible when the I/O controller and the CPU simultaneously access the memory:

- The I/O controller transfers data only as a complete data set.
- The CPU can only access completely updated data sets.
- The I/O controller cannot read or write data as long as the CPU accesses consistent data.

The result becomes clear from the example below:



1. As the I/O controller can only write when the CPU does not read, the I/O controller has to wait until the data are completely read by the CPU.
2. The I/O controller only writes a complete data set into the DPM.

Configuring consistent data

Note!

Consistency is achieved by an appropriate I/O controller configuration (see documentation for the configuring software).

10 Monitoring

10.1 Interruption of PROFINET communication

An interruption of PROFINET communication in the "Data_Exchange" state, e.g. by cable break or failure of the I/O controller is recognised by the I/O device.

The response to the interruption of communication is controlled via the following settings:

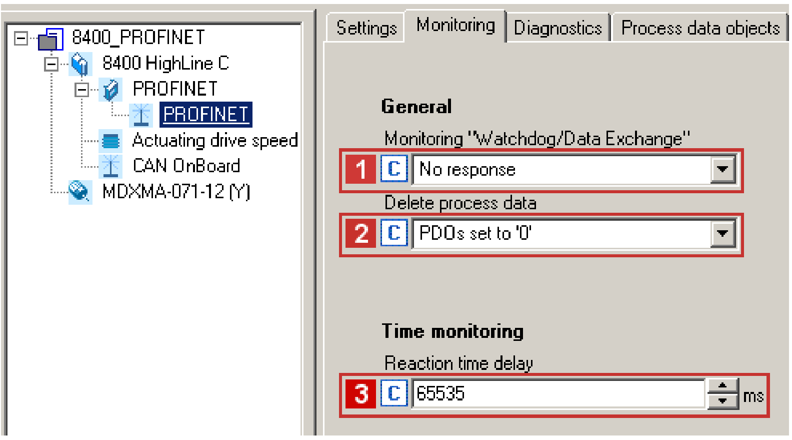
1. During the initialisation of PROFINET communication the watchdog monitoring time specified in the I/O controller (C13882) is transferred to the I/O device.

   If the I/O device does not receive any valid process data in the "Data_Exchange" state, the process data are treated according to the setting in 2 C13885. (Like this the data that were sent last by the I/O controller can be used or set to zero.)

   After the watchdog monitoring time has elapsed, the I/O device changes to the "No_Data_Exchange" status (see C13861), and the red LED **BE** is activated (Fieldbus status displays (📖 75)).

   There is no response in the I/O device.

2. To trigger a response in the IO device, you additionally have to set a **Response of the Inverter Drive 8400** 1 (C13880) under the **Monitoring** tab in the »Engineer«.



By setting a **Reaction time delay** 3 (C13881) you can decelerate this response.

- In the Lenze setting "0 ms", this monitoring is activated.
- With the setting "65535 ms", this monitoring is deactivated.
- A change of monitoring will be effective immediately.
- The monitoring time elapses when the "Data_Exchange" status is exited.

After this response delay has elapsed, the response set is executed with the error message "PROFINET: Data_Exchange status quit [0x01bc6531]" (📖 83).

10.2 Interruption of internal communication

- The response in the event of a communication error between the communication module and the Inverter Drive 8400 can be set via code **C01501**.
- The communication module reports a connection interruption to the IO controller and changes to the "No_Data_Exchange" state.

- The error message "PROFINET: Exist. conn. to 8400 lost [0x01bc3100]" (📖 81) is output.

11 Diagnostics

For purposes of fault diagnostics, the communication module is provided with the LEDs on the front. Furthermore you can carry out the Diagnostics with the »Engineer« (📖 77).

11.1 LED status displays

Note!

During normal operation ...

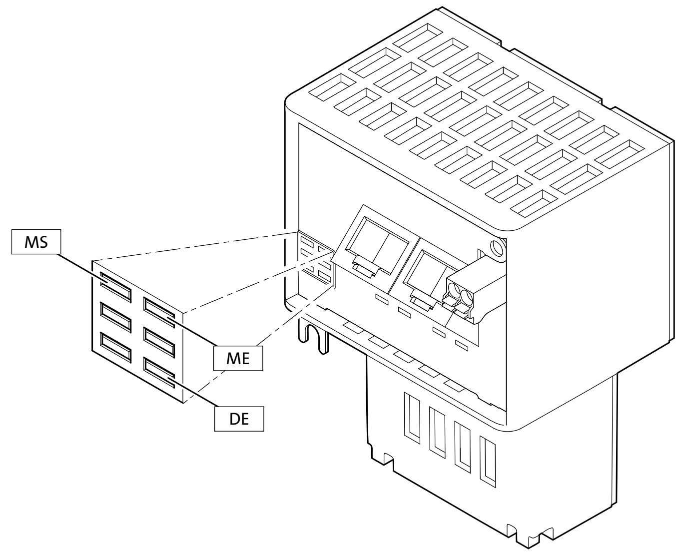
- only the LEDs **MS** (📖 74) and **BS** (📖 75) should be lit permanently.
- the green LEDs on the RJ45 sockets X256/X257 must be lit or blinking (📖 76).

The following status displays are distinguished:

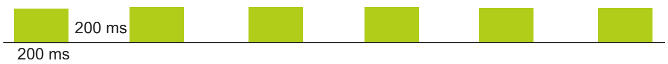
- Module status displays (📖 74)
- Fieldbus status displays (📖 75)
- Status displays at X256 and X257 (📖 76)

11.1.1 Module status displays

The LEDs **MS**, **ME** and **DE** indicate the module status.

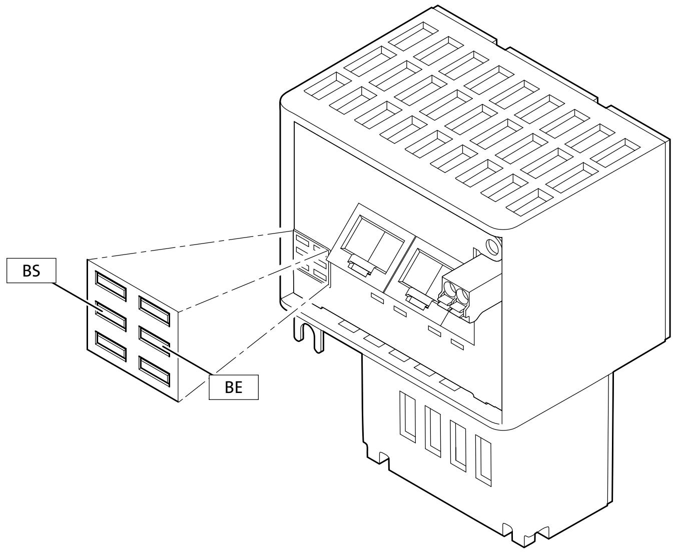


E84YCER006

LED	Colour	Status	Description
MS	green	On	The communication module is supplied with voltage and is connected to the standard device.
		blinking	 The communication module is supplied with voltage, but is not connected to the standard device. (Standard device is switched off, in the initialisation phase, or not available.)
ME	red	On	An error has occurred in the communication module.
		blinking	 The "Node blinking test" PROFINET function is activated by the I/O controller. The blinking LED serves to identify/localise accessible I/O devices.
EN	red	On	The communication module is not accepted by the basic device or the basic device is not active (see notes in the documentation relating to the basic device.)

11.1.2 Fieldbus status displays

The LEDs **BS** and **BE** indicate the fieldbus status.

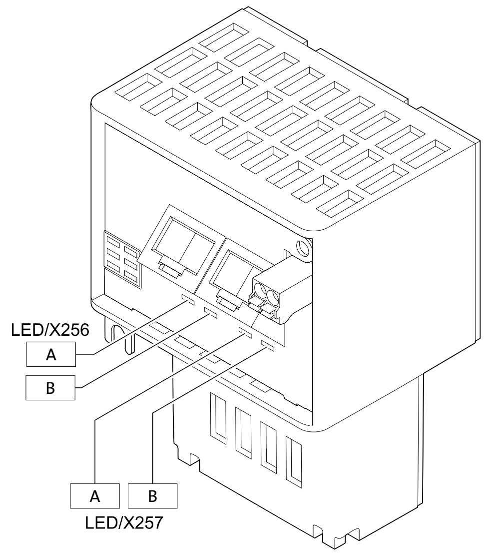


E84YCER006

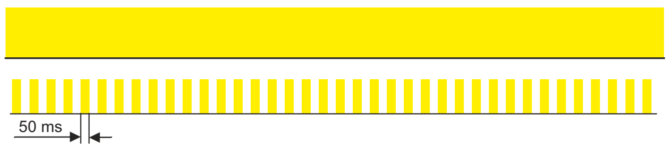
LED	Colour	Status	Description
BS	green	Off	The communication module is not active on the fieldbus or is in the INIT state.
		blinking	200 ms 200 ms The communication module is in the DATA _EXCHANGE status.
BE	red	On	Bus error/fault is active (e.g. Ethernet cable not connected). The communication module is in the NO_DATA_EXCHANGE status.
		blinking	200 ms 200 ms Impermissible settings: • Invalid station name • Invalid IP parameters The communication module has been initialised and continues to work internally with the respective standard values.

11.1.3 Status displays at X256 and X257

The LEDs below the RJ45 sockets X256 and X257 indicate the PROFINET connection status.

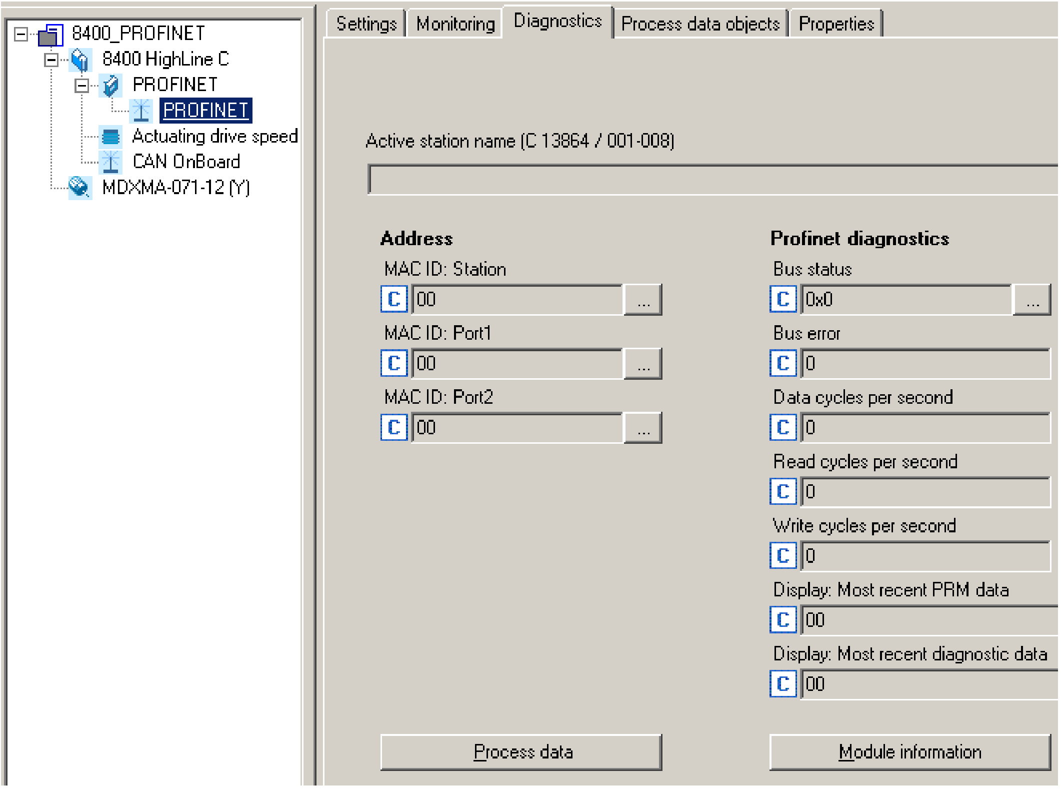


E84YCER005

LED	Colour	Status	Description
A	green	Off	No connection to PROFINET
		On	 Connection to PROFINET has been established.
B	Yellow	Off	No communication
		On *or* Flickers	 Communication active

11.2 Diagnostics with the »Engineer«

In the »Engineer« under the **Diagnostics** tab, you will find PROFINET diagnostics information.

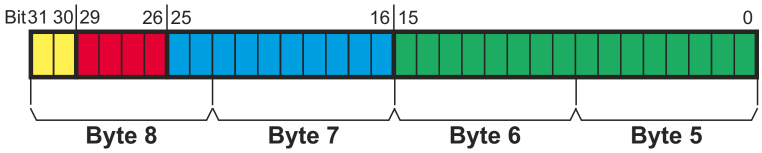


11.3 Diagnostic data

- The I/O device sends an alarm message to the I/O controller to signalise the diagnostic data below.
- Errors and warnings of the Inverter Drive 8400 and the module connected to it are sent to the IO controller as extended diagnostic messages.
- With code C13887, the transmission of an alarm message can be suppressed for specific error messages.
- The diagnostic data can be displayed using the hexadecimal representation of the Siemens S7 engineering tool.

Bytes	Meaning	Value [hex]
1 ... 6	Diag. block header	0x0010 001C 0100
7 ... 8	Alarm type	0x0001 (diagnosis)
9 ... 12	API	0x0000 0000
13, 14	Slot number	0x0001 / 0x0002
15, 16	Subslot number	0x0001
17 ... 20	Module ID	ID according to module
21 ... 24	Submodule ID	ID according to module
25, 26	Alarm specifier	0xB000
27, 28	User structure identifier	0x0001
29 ... 32	Error code of the Inverter Drive 8400 / E84AYCER	

Error code of Inverter Drive 8400



- The error code can be found in bytes 29 ... 32 of the diagnostic message.
- In the logbook and in code **C00165**, the error number is shown in the following syntax in order to facilitate the readability:
  [error type].[error subject area no.].[error ID]

Example: error message "PROFINET: Data_Exchange status quit [0x01bc6531]"

Byte 32								Byte 31								Byte 30								Byte 29							
0x01								0xbc								0x65								0x31							
0	0	0	0	0	0	0	1	1	0	1	1	1	1	0	0	0	1	1	0	0	1	0	1	0	0	1	1	0	0	0	1
Response		Instance ID				Module ID										Error ID															

Software manual/»Engineer« online help for Inverter Drive 8400
Detailed information on the error codes is provided here.

12 Error messages

[ALinkKeywords] Error messages

This chapter supplements the error list in the software manual and the »Engineer« online help for the Inverter Drive 8400 by the error messages of the communication module.

Software manual/»Engineer« online help for Inverter Drive 8400

Here you will find general information on diagnostics & fault analysis and on error messages.

[Search-Keywords] Error messages

12.1 Short overview of the PROFINET error messages

The following table contains all PROFINET error messages in numerical order of the error number. Furthermore the preset error response and - if applicable – the parameter for setting the error response is specified.

Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error no. [hex]	Subject area no. [dec]	Error no. [dec]	Error text	Error type (Error response)	Adjustable in
0x01bc3100	444	12544	PROFINET: Exist. connect. to 8400 lost	1: No Response	-
0x01bc5531	444	21809	PROFINET: Memory: No access	1: No Response	-
0x01bc5532	444	21810	PROFINET: Memory: Read error	1: No Response	-
0x01bc5533	444	21811	PROFINET: Memory: Write error	1: No Response	-
0x01bc6010	444	24592	PROFINET: Restart by watchdog reset	1: No Response	-
0x01bc6011	444	24593	PROFINET: Internal error	1: No Response	-
0x01bc6100	444	24832	PROFINET: Internal error	1: No Response	-
0x01bc6101	444	24833	PROFINET: Internal error	1: No Response	-
0x01bc641f	444	25631	PROFINET: Invalid parameter set	1: No Response	-
0x01bc6420	444	25632	PROFINET: Error: Lenze setting loaded	1: No Response	-
0x01bc6430	444	25648	PROFINET: Invalid module configuration	1: No Response	-
0x01bc6501	444	25857	PROFINET: Record parameter: Invalid read	4: Warning locked	-
0x01bc6502	444	25858	PROFINET: Record parameter: Invalid write	4: Warning locked	-
0x01bc6503	444	25859	PROFINET: Data output status bad	4: Warning locked	-
0x01bc6531	444	25905	PROFINET: Data_Exchange status quit	0: None	C13880/1
0x01bc6532	444	25906	PROFINET: Station name error	1: No Response	-
0x01bc6533	444	25907	PROFINET: IIP address error	1: No Response	-
0x01bc6534	444	25908	PROFINET: Stack init error	1: No Response	-
0x01bc6650	444	26192	PROFINET: Internal error	1: No Response	-

12.2 Possible causes and remedies

In this chapter, all error messages of the communication module are listed in numerical order of the error numbers. Possible causes and remedies and responses to the error messages are described in detail.

Tip!

You will find a list of all error messages of the communication module in alphabetical order in the previous chapter "Short overview of the PROFINET error messages" (📖 80).

PROFINET: Exist. conn. to 8400 lost [0x01bc3100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
MCI communication to Inverter Drive 8400 is interrupted. • Inverter Drive 8400 is switched off. • The communication module is not connected correctly at the MCI slot of the Inverter Drive 8400.	• Switch on Inverter Drive 8400. • Check screwed connection of the communication module at the MCI slot of the Inverter Drive 8400. • Send the communication module and Inverter Drive 8400 together with a description of the fault to Lenze.

PROFINET: Memory: No access [0x01bc5531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Access to memory was not possible.	Send communication module together with a description of the fault to Lenze.

PROFINET: Memory: Read error [0x01bc5532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Parameter could not be read.	Send communication module together with a description of the fault to Lenze.

PROFINET: Memory: Write error [0x01bc5533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Parameter could not be written.	Send communication module together with a description of the fault to Lenze.

PROFINET: Restart by watchdog reset [0x01bc6010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

PROFINET: Internal error [0x01bc6011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

PROFINET: Internal error [0x01bc6100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Internal error	Send communication module together with a description of the fault to Lenze.

PROFINET: Internal error [0x01bc6101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Internal error	Send communication module together with a description of the fault to Lenze.

PROFINET: Invalid parameter set [0x01bc641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

PROFINET: Error: Lenze setting loaded [0x01bc6420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set in the memory module was not possible.	• Download application again (including module). • Send device with error description to Lenze.

PROFINET: Invalid module configuration [0x01bc6430]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
A module or submodule does not comply with the configuration of the Siemens S7 engineering tool.	Check configuration.

PROFINET: Record Parameter: Invalid read [0x01bc6501]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
Invalid parameter read access	Check configuration.

PROFINET: Record Parameter: Invalid write [0x01bc6502]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
Invalid parameter write access When reading back the data, the IO Controller has requested a data length (number of data bytes) that is to small.	Check configuration.

PROFINET: Data output status bad [0x01bc6503]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
• Output data invalid. • Connection to Siemens S7 has been interrupted.	• Check cables and terminals. • Connect network cable to PROFINET connections.

PROFINET: Data_Exchange status quit [0x01bc6531]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
The data exchange via PROFINET has been terminated. • Also see the chapter "Interruption of PROFINET communication" (📖 71).	• Check cables and terminals. • Connect network cable to PROFINET connections.

PROFINET: Station name error [0x01bc6532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Station name is not DNS-conform.	Use a DNS-compliant station name. ▶ Setting the station name (📖 36)

PROFINET: IP address error [0x01bc6533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
An invalid IP address has been assigned by the I/O controller via PROFINET or has been set in code C13000.	• Make sure that the I/O controller has assigned a valid IP address via PROFINET. • Set a valid IP address. ▸ Setting the IP configuration (📖 38)

PROFINET: Stack init error [0x01bc6534]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
The stack cannot be initialised with the parameters selected by the user. This may be due to a station name which does not comply with the PROFINET specification.	Check and, if necessary, adapt PROFINET parameters: ▸ Setting the IP configuration (📖 38) ▸ Setting the station name (📖 36)

PROFINET: Internal error [0x01bc6650]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Internal error	Send communication module together with a description of the fault to Lenze.

13 Parameter reference

[ALinkKeywords] Parameters

This chapter supplements the parameter list and the table of attributes for the Inverter Drive 8400 contained in the software manual and in the »Engineer« online help by the parameters of the communication module E84AYCER (PROFINET).

Software manual/»Engineer« online help for the Inverter Drive 8400

Here you will find general information on parameters.

[Search-Keywords] Hlp_Para

13.1 Parameters of the communication module

This chapter lists the parameters of the E84AYCER communication module (PROFINET) in numerically ascending order.

Note!

PROFINET command "Reset to Factory Defaults"

If the "Reset to factory defaults" PROFINET command is executed by an IO supervisor or an I/O controller, the PROFINET-specific parameters will be reset to their standard values:

- C13000 | IP address
- C13001 | Subnetwork mask
- C13002 | Gateway address
- C13010 | Active IP address
- C13011 | Active subnetwork mask
- C13012 | Active gateway address
- C13864 | Active station name
- C13887 | Suppress signalling diag. mess. upon
- C13899 | Station name
- C13910 | I&M1 system designation
- C13911 | I&M1 installation site
- C13912 | I&M2 installation date
- C13913 | I&M3 additional information
- C13914 | I&M4 signature code

C13000

Parameter | Name: **C13000 | IP address**			Data type: UNSIGNED_32 Index: 11575_d = $2D37_h$
Setting of the IP address ▸ Setting the IP configuration (📖 38)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13001

Parameter | Name: **C13001 | Subnetwork mask**			Data type: UNSIGNED_32 Index: 11574_d = $2D36_h$
Setting of the subnet mask ▶ Setting the IP configuration (📖 38)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13002

Parameter | Name: **C13002 | Gateway address**			Data type: UNSIGNED_32 Index: 11573_d = $2D35_h$
Setting of the gateway address ▶ Setting the IP configuration (📖 38)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13003

Parameter | Name: **C13003 | Physical address**	Data type: OCTET_STRING Index: 11572_d = $2D34_h$
Display of the MAC-ID	
Subcodes	**Info**
C13003/1	MAC ID: Station
C13003/2	MAC ID: Port1
C13003/3	MAC ID: Port2
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13010

Parameter | Name: **C13010 | Active IP address**		Data type: UNSIGNED_8 Index: 11565_d = $2D2D_h$
Display of the active IP address • The active IP address may differ from the contents of code C13000, depending on whether the station name was changed via the fieldbus or the parameter.		
Subcodes	**Info**	
C13010/1	Active IP address.1	
C13010/2	Active IP address.2	
C13010/3	Active IP address.3	
C13010/4	Active IP address.4	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13011

Parameter | Name: **C13011 | Active subnetwork mask**				Data type: UNSIGNED_8 Index: 11564_d = $2D2C_h$
Display of the active subnetwork mask • The active subnetwork mask may differ from the contents of code C13001, depending on whether the station name was changed via the fieldbus or the parameter.				
Display range (min. value | unit | max. value)				
0		255		
Subcodes			**Info**	
C13011/1			Active subnetwork mask.1	
C13011/2			Active subnetwork mask.2	
C13011/3			Active subnetwork mask.3	
C13011/4			Active subnetwork mask.4	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

C13012

Parameter | Name: **C13012 | Active gateway address**				Data type: UNSIGNED_8 Index: 11563_d = $2D2B_h$
Display of the active gateway address • The active gateway address may differ from the contents of code C13002, depending on whether the station name was changed via the fieldbus or the parameter.				
Display range (min. value | unit | max. value)				
0		255		
Subcodes			**Info**	
C13012/1			Active gateway address.1	
C13012/2			Active gateway address.2	
C13012/3			Active gateway address.3	
C13012/4			Active gateway address.4	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

C13850

Parameter | Name: **C13850 | All words to master**			Data type: UNSIGNED_16 Index: 10725_d = $29E5_h$
Display of the process data words which are transmitted from the inverter to the IO controller. In the subcodes 1 to 16, all process data words to the I/O controller are displayed. However, only the configured process data words are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master**			Data type: UNSIGNED_16 Index: 10724_d = $29E4_h$
Display of the process data words which are transmitted from the IO controller to the inverter. In the subcodes 1 to 16, all process data words from the I/O controller are displayed. However, only the configured process data words are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to standard device**			Data type: UNSIGNED_16 Index: 10723_d = $29E3_h$
Display of the process data words 1 ... 16, which are transmitted from the IO controller to the inverter.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words from standard device**				Data type: UNSIGNED_16 Index: 10722_d = $29E2_h$
Display of the process data words 1 ... 16, which are transmitted from the inverter to the IO controller.				
Display range (min. value | unit | max. value)				
0		65535		
Subcodes			**Info**	
C13853/1				
...				
C13853/16				
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

C13860

Parameter | Name: **C13860 | Settings**				Data type: UNSIGNED_8 Index: 10715_d = $29DB_h$
Display range (min. value | unit | max. value)				
0		255		
Subcodes			**Info**	
C13860/1			Reserved	
C13860/2			Number of process data words	
C13860/3			Reserved	
C13860/4			Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

C13861

Parameter | Name: **C13861 | Bus status**		Data type: BITFIELD_16 Index: 10714_d = $29DA_h$
Bit-coded display of current bus status		
Value is bit-coded:		
Bit 0	Initialised	
Bit 1	Online	
Bit 2	Connected	
Bit 3	Address conflict	
Bit 4	Hardware error	
Bit 5	EEPROM error	
Bit 6	Watchdog error	
Bit 7	Protocol error	
Bit 8	Profinet stack ok	
Bit 9	Profinet stack not configured	
Bit 10	Ethernet controller error	
Bit 11	UDP stack error	
Bit 12	Reserved	
Bit 13	Reserved	
Bit 14	Reserved	
Bit 15	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13862

Parameter | Name: **C13862 | Bus counter**				Data type: UNSIGNED_32 Index: 10713_d = $29D9_h$
Display of the data cycles per second (independent of data changes)				
Display range (min. value | unit | max. value)				
0		4294967295		
Subcodes			**Info**	
C13862/1			Data cycles per second	
C13862/2			Read cycles per second	
C13862/3			Write cycles per second	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

C13864

Parameter | Name: **C13864 | Active station name**		Data type: VISIBLE_STRING Index: 10711_d = $29D7_h$
Displays the active station name used by the inverter. It may differ from the contents of code C13899, depending on whether the station name has been changed via the fieldbus or via C13899. ▸ Setting the station name (📖 36)		
Subcodes	**Info**	
C13864/1	1st ... 30th character	
C13864/2	31th ... 60th character	
C13864/3	61th ... 90th character	
C13864/4	91th ... 120th character	
C13864/5	121th ... 150th character	
C13864/6	151th ... 180th character	
C13864/7	181th ... 210th character	
C13864/8	211th ... 240th character	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13867

Parameter | Name: **C13867 | Display: Most recent diagnostic data**	Data type: OCTET_STRING Index: 10708_d = $29D4_h$
Display of the diagnostic data sent by the inverter most recently.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Bytes	Information
0	Slot
1	
2	Subslot
3	
4	Error code
...	
7	
8	Slot
9	
10	Subslot
11	
12	Error code
...	
15	

C13877

Parameter | Name: **C13877 | Bus error(1)**		Data type: UNSIGNED_16 Index: 10698_d = $29CA_h$
The code contains the error currently detected on the fieldbus. • The error values can occur in combination with the error values from code C13878.		
Selection list (read only)		**Info**
0	No error	
1	Internal error	
2	Unit ID unknown	
3	Max. units exceeded	
4	Invalid size	
5	Unit type unknown	
6	Runtime plug	
7	Invalid argument	
8	Service pending	
9	Stack not ready	
10	Command unknown	
11	Invalid address descriptor	
12	Watchdog expired	
13	Protocol not supported	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13878

Parameter | Name: **C13878 | Bus error(2)**		Data type: BITFIELD_16 Index: 10697_d = $29C9_h$
The code contains the error currently detected on the fieldbus. • The error values can occur in combination with the error values from code C13877.		
Value is bit-coded:		**Info**
Bit 0	Reserved	
...	...	
Bit 6	Reserved	
Bit 7	IP address error	
Bit 8	Station name error	
Bit 9	DataExch left	
Bit 10	Stack boot error	
Bit 11	Stack online error	
Bit 12	Stack state error	
Bit 13	Stack revision error	
Bit 14	Stack init error	
Bit 15	Stack CPU boot error	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13880

Parameter | Name: **C13880 | Monitoring Reaction**		Data type: UNSIGNED_8 Index: 10695_d = $29C7_h$
The action set in subcode 1 of the code is carried out when the node recognises that it is no longer in the DATA_EXCHANGE status. • Please also observe the notes provided in code C13881. • A change in the monitoring response becomes immediately effective.		
Selection list		
0	No response	
1	Error	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
Subcodes	**Lenze setting**	**Info**
C13880/1	0: No Response	"Watchdog/Data Exchange" monitoring
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13881

Parameter | Name: **C13881 | Reaction time delay**			Data type: UNSIGNED_16 Index: 10694_d = $29C6_h$
If the DATA_EXCHANGE status is exited, the response parameterised in C13880 is activated after the time set here has elapsed. • A value of "65535" in this code deactivates the monitoring function. • A change of monitoring will be effective immediately.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Clear process data**		Data type: UNSIGNED_8 Index: 10690_d = $29C2_h$
This code is used to set the process data that are to be processed by the I/O device in order to maintain the internal communication when the PROFINET has left the DATA_EXCHANGE status.		
Selection list (Lenze setting printed in bold)		**Info**
0	Use of most recent master PDOs	The last data sent by the I/O controller are used.
1	**PDOs are set to the value 0'**	The contents of the process data are set to the value "0".
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13887

Parameter | Name: **C13887 | Suppress signalling diag. mess. upon**		Data type: BITFIELD_8 Index: 10688_d = $29C0_h$
This code serves to inhibit the transmission of alarm messages to the IO controller. By this, errors of a specific type can be systematically suppressed. All errors are furthermore entered in the logbook. • A change will only be effective immediately if no error number with the error type selected here is active in standard device code **C00165**.		
Value is bit-coded:		**Info**
Bit 0	Error	
Bit 1	Fault	
Bit 2	Quick stop by trouble	
Bit 3	Warning Locked	
Bit 4	Warning	
Bit 5	Information	
Bit 6	Reserved	
Bit 7	Connection to 8400 lost	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13899

Parameter | Name: **C13899 | Station name**		Data type: VISIBLE_STRING Index: 10676_d = $29B4_h$
The name with a max. length of 240 characters is distributed to the subindices. The name can be entered starting with subindex 1. The following unused subindices are not relevant. • The station name must be assigned in accordance with the PROFINET specification. In the standard setting a deleted name is displayed. The name is also deleted if the "Reset to factory defaults" command is executed by an IO supervisor or an I/O controller. • A change of the station name will only become effective by switching the mains of the inverter. ▸ Setting the station name (🕮 36)		
Subcodes	**Lenze setting**	**Info**
C13899/1		1st ... 30th character
C13899/2		31th ... 60th character
C13899/3		61th ... 90th character
C13899/4		91th ... 120th character
C13899/5		121th ... 150th character
C13899/6		151th ... 180th character
C13899/7		181th ... 210th character
C13899/8		211th ... 240th character
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13900

Parameter | Name: **C13900 | Firmware product type**	Data type: VISIBLE_STRING Index: 10675_d = $29B3_h$
The code contains a string with a length of 8 characters. The "E84DFYER" identification code is output.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 10674_d = $29B2_h$
The code contains a string with a length of 20 characters. The creation date ("MMM DD YYYY") and time ("hh:mm:ss") of the software are displayed, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware version**	Data type: VISIBLE_STRING Index: 10673_d = $29B1_h$
The code contains a string with a length of 11 characters. The identification code is displayed (e.g. "01.00.00.00").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13910

Parameter | Name: **C13910 | I&M1 System designation**	Data type: VISIBLE_STRING Index: 10665_d = $29A9_h$
Input/output of the I&M1 plant identification code • The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13911

Parameter | Name: **C13911 | I&M1 Installation site**	Data type: VISIBLE_STRING Index: 10664_d = $29A8_h$
Input/output of the I&M1 location identification code • The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13912

Parameter | Name: **C13912 | I&M2 Installation date**	Data type: VISIBLE_STRING Index: 10663_d = $29A7_h$
Input/output of the I&M2 date of installation • The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13913

Parameter | Name: **C13913 | I&M3 additional information**		Data type: VISIBLE_STRING Index: 10662_d = $29A6_h$
Input/output if the I&M3 additional information • The Lenze setting shows an empty string.		
Subcodes	**Lenze setting**	**Info**
C13913/1		I&M3 additional information
C13913/2		I&M3 additional information
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13914

Parameter | Name: **C13914 | I&M4 signature code**		Data type: OCTET_STRING Index: 10661_d = $29A5_h$
Input/output of the I&M4 signature		
Subcodes	**Lenze setting**	**Info**
C13914/1	00000000000000000000000000000 0000000000000000000000000	I&M4 signature code
C13914/2	00000000000000000000000000000 0000000000000000000000000	I&M4 signature code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

13.2 Table of attributes

The table of attributes contains information that is required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ No decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible if controller inhibit is set	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C13000	IP address	11575	2D37	E	1	UNSIGNED_32	1	☑	☑	
C13001	Subnet Mask	11574	2D36	E	1	UNSIGNED_32	1	☑	☑	
C13002	Gateway address	11573	2D35	E	1	UNSIGNED_32	1	☑	☑	
C13003	Physical address	11572	2D34	A	3	OCTET_STRING		☑		
C13010	Active IP Address	11565	2D2D	A	4	UNSIGNED_8	1	☑		
C13011	Active Subnetwork Mask	11564	2D2C	A	4	UNSIGNED_8	1	☑		
C13012	Active IP Address	11563	2D2B	A	4	UNSIGNED_8	1	☑		
C13850	All words from drive to master	10725	29E5	A	16	UNSIGNED_16	1	☑		
C13851	All words from master to drive	10724	29E4	A	16	UNSIGNED_16	1	☑		
C13852	All words to the basic device	10723	29E3	A	16	UNSIGNED_16	1	☑		
C13853	All words to the basic device	10722	29E2	A	16	UNSIGNED_16	1	☑		
C13860	Settings	10715	29DB	A	4	UNSIGNED_8	1	☑		
C13861	Bus status	10714	29DA	E	1	BITFIELD_16		☑		
C13862	Bus counter	10713	29D9	A	3	UNSIGNED_32	1	☑		
C13864	Active station name	10711	29D7	A	8	VISIBLE_STRING		☑		
C13867	Display: Most recent diagnostic data	10708	29D4	E	1	OCTET_STRING		☑		
C13877	Bus error(1)	10698	29CA	E	1	UNSIGNED_16	1	☑		
C13878	- Bus error(2)	10697	29C9	E	1	BITFIELD_16		☑		
C13880	Monitoring Reaction	10695	29C7	A	1	UNSIGNED_8	1	☑	☑	
C13881	Reaction time delay	10694	29C6	E	1	UNSIGNED_16	1	☑	☑	
C13885	Delete process data	10690	29C2	E	1	UNSIGNED_8	1	☑	☑	
C13887	Suppress signalling diag. mess. upon	10688	29C0	E	1	BITFIELD_8		☑	☑	
C13899	Station name	10676	29B4	A	8	VISIBLE_STRING		☑	☑	
C13900	Firmware Product Type	10675	29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware Compilation Date	10674	29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware Version	10673	29B1	E	1	VISIBLE_STRING		☑		
C13910	I&M1 system designation	10665	29A9	E	1	VISIBLE_STRING		☑	☑	
C13911	I&M1 installation site	10664	29A8	E	1	VISIBLE_STRING		☑	☑	
C13912	I&M2 installation date	10663	29A7	E	1	VISIBLE_STRING		☑	☑	
C13913	I&M3 additional information	10662	29A6	A	2	VISIBLE_STRING		☑	☑	
C13914	I&M4 signature code	10661	29A5	A	2	OCTET_STRING		☑	☑	

14 Index

Lenze Drives GmbH
Postfach 10 13 52, D-31763 Hameln
Breslauer Straße 3, D-32699 Extertal
Germany
HR Lemgo B 6478
+49 5154 82-0
+49 5154 82-2800
sales.de@lenze.com
www.lenze.com

Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service.de@lenze.com

Accessories

EtherCAT®

E84AYCET

Inverter Drives 8400 ________________________

Communication Manual EN

13429553

Lenze

Contents

1 About this documentation

Contents

This documentation only contains descriptions for the E84AYCET communication module (EtherCAT®).

Note!

This documentation supplements the **Mounting Instructions** supplied with the communication module and the **"Inverter Drives 8400" Hardware Manual**.

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical concepts are only explained to the level of detail required to understand the function of the communication module.

This documentation does not describe any software provided by other manufacturers. No warranty can be given for corresponding data provided in this documentation. For information on how to use the software, please refer to the control system documents (Controller, EtherCAT master).

All product names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information about EtherCAT can be found on the website of the EtherCAT Technology Group:

www.EtherCAT.org

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.

Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.Lenze.com

Validity information

The information in this documentation applies to the following devices:

Extension module	Type designation	From hardware version	From software version
EtherCAT communication module	E84AYCET	VC	01.00...03
		1A	01.04/05 02.00

Screenshots/application examples

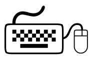
All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the engineering tools installed (e.g. »Engineer«), the screenshots in this documentation may differ from the actual screen representation.

1.1 Document history

Version			Description
1.0	08/2009	TD17	First edition
2.0	10/2009	TD17	Amendment of the information on the synchronisation via the fieldbus and general revision
3.0	04/2010	TD17	General revision
4.0	11/2010	TD17	Information about the EtherCAT register "AL Status Code" (📖 58) supplemented.
4.1	11/2012	TD17	EtherCAT® is a registered trademark by Beckhoff Automation GmbH, Germany.
5.0	05/2013	TD17	• Revision of firmware version V02.00 • New layout

1.2 Conventions used

This documentation uses the following conventions to distinguish different types of information:

Type of information	Identification	Examples/notes
Numbers		
Decimal separator	Point	In general, the decimal point is used. Example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Text colour blue	All pieces of information that only apply to or from a specific software version of the inverter are highlighted accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »Engineer«...
Window	*italics*	The *message window*... / The *Options* dialog box ...
Variable name		Setting *bEnable* to TRUE...
Control element	**Bold**	The **OK** button ... / The **Copy** command ... / The Properties tab ... / The **Name** input field ...
Sequence of menu commands		If several successive commands are required for executing a function, the individual commands are separated from each other by an arrow: Select the command **File → Open** to...
Hyperlink	Underlined	Optically highlighted reference to another topic. Can be activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 5)	Optically highlighted reference to another page. Can be activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are marked by a pictograph.

1.3 Terminology used

Term	Meaning
Inverter	Lenze inverters of the "Inverter Drives 8400" series
Standard device	
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in so-called "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). In normal usage, the term is also referred to as "Subindex".
CoE	CANopen over EtherCAT
DC	"Distributed clocks" for EtherCAT synchronisation
»Engineer«	PC software from Lenze which supports you during engineering (parameterisation, diagnostics, and configuration) throughout the entire life cycle, i.e. from planning to maintenance of the commissioned machine.
»PLC Designer«	
ESI	EtherCAT slave information (device description file in XML format)
EtherCAT	EtherCAT® is a real-time capable Ethernet system with the highest performance. EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
EtherCAT master	Control system (Controller)
HW	Hardware
I-1600.8	CoE index (hexadecimal representation) In the example: index 0x1600, subindex 8
Lenze setting	Settings with which the device is preconfigured ex works.
Standard setting	
PDO	Process data object
SDO	Service data object
SW	Software
»TwinCAT«	Beckhoff PC software for EtherCAT configuration

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and suggests how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of damage to material assets** Reference to a possible danger that may result in damage to material assets if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for easy handling
		Reference to other documentation

2 Safety instructions

Note!

Always observe the specified safety measures to avoid severe injury to persons and damage to property!

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If you ignore the following basic safety measures, severe injury to persons and damage to material assets may result.

Lenze drive and automation components ...

- shall only be used as directed.
  ▸ Application as directed (📖 12)
- must never be commissioned if they display any signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without the covers required.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

For Lenze drive components ...

- only use the accessories approved.
- only use genuine spare parts supplied by the manufacturer of the product.

Observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for a safe and trouble-free operation and for obtaining the product features specified.
  ▸ Product features (📖 13)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...

- who are familiar with the installation, assembly, commissioning and operation of the product.
- who have the corresponding qualifications for their work.
- who know all regulations for the prevention of accidents, directives and laws applicable on site and are able to apply them.

2.2 Device- and application-specific safety instructions

- During operation, the communication module must be firmly connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables that correspond to the given specifications.
  ▶ Specification of the Ethernet cable (📖 29)

Documentation for the standard device, control system, plant/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes stated in this manual.

2.3 Residual hazards

Protection of persons

If Inverter Drives 8400 are used on a phase earthed mains with a rated mains voltage of ≥ 400 V, external measures need to implemented in order to provide reliable protection against accidental contact.

▶ Protective insulation (📖 16)

Device protection

The communication module contains electronic components that can be damaged or destroyed by electrostatic discharge.

▶ Installation (📖 21)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module which can be used with the following standard devices:

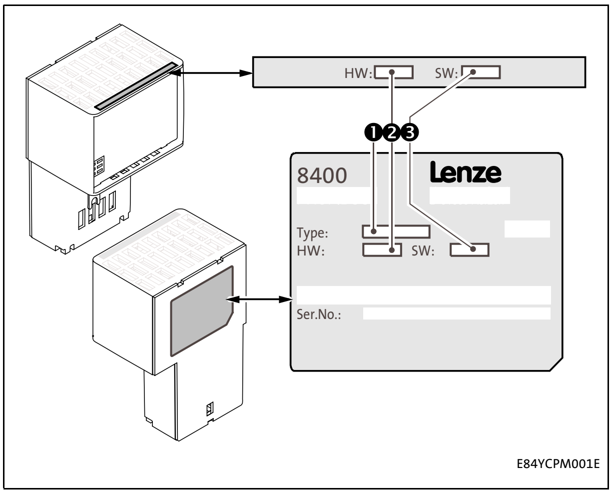
Product series	Type designation	From software version
Inverter Drives 8400 StateLine	E84AVSCxxxxx	03.00
Inverter Drives 8400 HighLine	E84AVHCxxxxx	02.00
Inverter Drives 8400 TopLine	E84AVTCxxxxx	01.00
Inverter Drives 8400 TopLine P (with CiA402)	E84AVPCxxxxx	01.00

- is an item of equipment intended for use in industrial power systems.
- may only be operated under the operating conditions specified in this documentation.
- may only be used in EtherCAT networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation as well as the hardware and software version of the communication module are specified on the nameplate:



1 Type designation (type)

E84	Product series
A	Version
Y	Module identification: Extension module
C	Module type: Communication module
ET	EtherCAT
V/S	V: Coated version
	S: Standard version

2 Hardware version (HW)

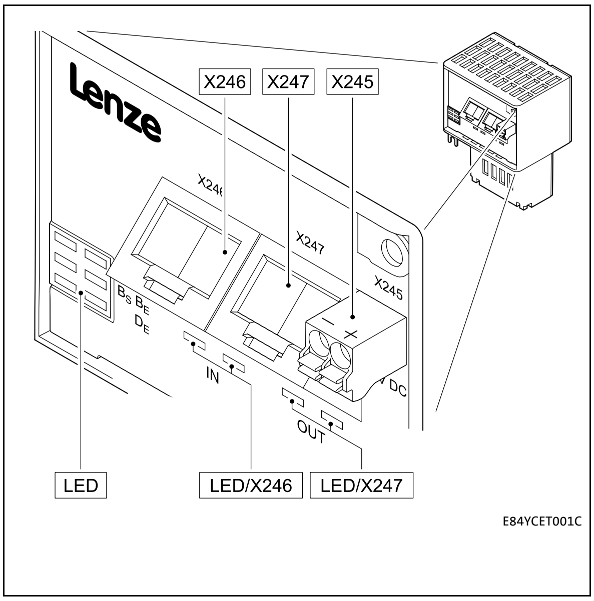
3 Software version (SW)

[3-1] Identification data

3.3 Product features

- Interface module for the EtherCAT communication system to the expansion slots of the Inverter Drives 8400
- The communication module can be supplied internally by the standard device and externally via a separate voltage source.
- Supports the "Distributed Clocks" (DC) functionality for synchronisation via the fieldbus
- Cycle times:
  - 1 ms or an integer multiple of 1 ms
  - max. 15 ms if "Distributed Clocks" (DC) are used
- PDO transfer with CoE (CANopen over EtherCAT)
- Up to 16 process data words (16 bits/word) per direction can be exchanged.
- Access to all Lenze parameters with CoE (CANopen over EtherCAT)

3.4 Terminals and interfaces



[3-2] E84AYCET communication module (EtherCAT)

X245 External voltage supply of the communication module
2-pole plug connector with spring connection
- External voltage supply (📖 31)

X246 EtherCAT input (IN)

X247 EtherCAT output (OUT)
- RJ45 sockets according to IEC/EN 60603-7
- with 2 LED status displays each for diagnostics
- Network topology (📖 26)
- EtherCAT connection (📖 27)
- Status displays at X246 and X247 (📖 88)

MS
ME
BS
BE
DE
5 LED status displays for diagnostics
- Module status displays (📖 85)
- Fieldbus status displays (📖 86)

4 Technical data

4.1 General data and operating conditions

Area	Values
Order designation	E84AYCET
Communication profile	EtherCAT
Supported device profile and mailbox protocol	CANopen over EtherCAT (CoE)
Communication medium	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e
Interface for communication	RJ45: Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Network topology	Line, switch
Bus device type	EtherCAT slave
Number of nodes	Max. 65535 (in the entire network)
Max. cable length between two EtherCAT nodes	100 m (typically)
Vendor ID [hex]	0x3B
Product ID	Depending on the standard device used (see chapter Automatic device identification (🕮 35)).
Revision ID	Depending on the software version of the communication module (see chapter Identification (🕮 12)).
Baud rate	100 Mbps, full duplex
Cycle times	1 ms or an integer multiple of 1 ms, max. 15 ms if "Distributed clocks" (DC) are used
Voltage supply	External supply via a separate power supply unit • "+": V = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %), I = 140 mA • "-": Reference potential for external voltage supply
Conformity, approvals	CE UL (see hardware manual "Inverter Drives 8400")

"Inverter Drives 8400" Hardware Manual

Here you can find the **ambient conditions** and information on the **electromagnetic compatibility (EMC)** which also apply to the communication module.

4.2 Protective insulation

Danger!

Dangerous voltage

If Inverter Drives 8400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to provide reliable protection against accidental contact.

Possible consequences:

Death or severe injury

Protective measures:

If protection against accidental contact is required for the control terminals of the inverter and the connections of the plugged device modules, ...

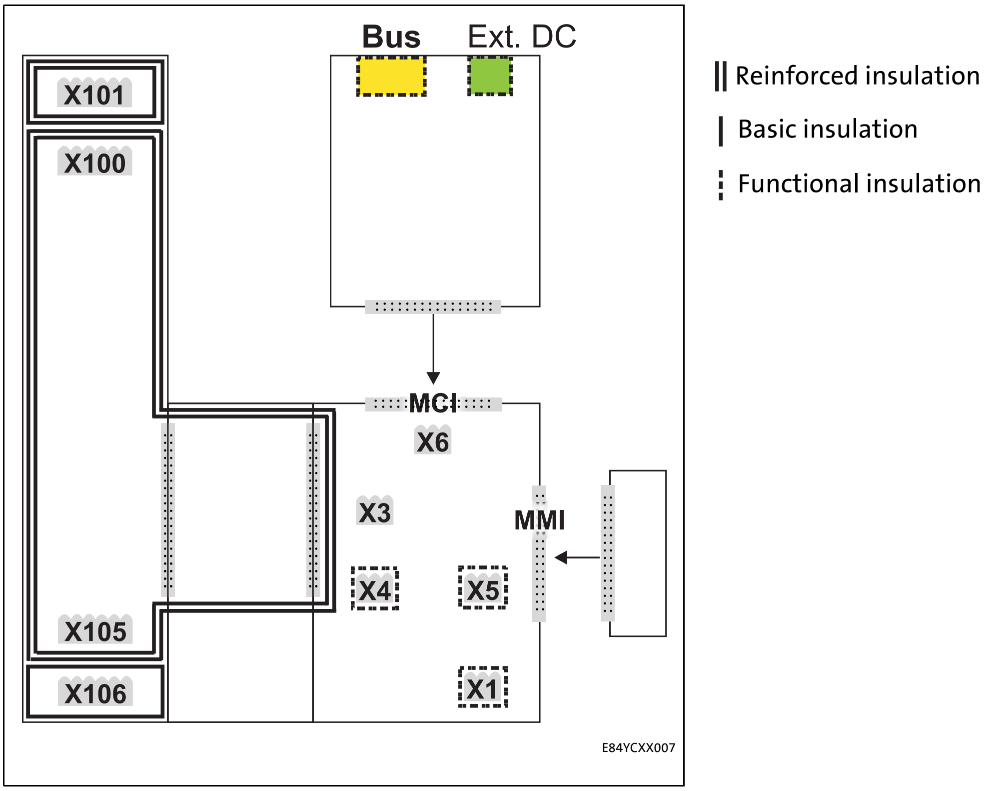
- a double isolating distance must be available.
- the components to be connected must be provided with a second isolating distance.

Note!

The protective insulation provided in Inverter Drives 8400 is realised in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the inverter.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection
X100	Mains / DC bus connection
X101	Relay contact
X105	Motor/brake resistor
X106	Motor PTC
X1	System bus (CAN)
X3	Analog inputs/outputs
X4	Digital outputs
X5	Digital inputs
X6	Diagnostics
MCI	Slot for communication module
MMI	Slot for memory module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MCI and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "basic insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between the X100 mains terminal and the bus terminal is of the "reinforced insulation" type.

4.3 Protocol data

Area	Values
Process data words	1 ... 16 process data words per direction (max. 32 bytes, 16 bits/word)
Parameter data (mailbox size for CoE transfer)	Max. 128 bytes

4.4 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

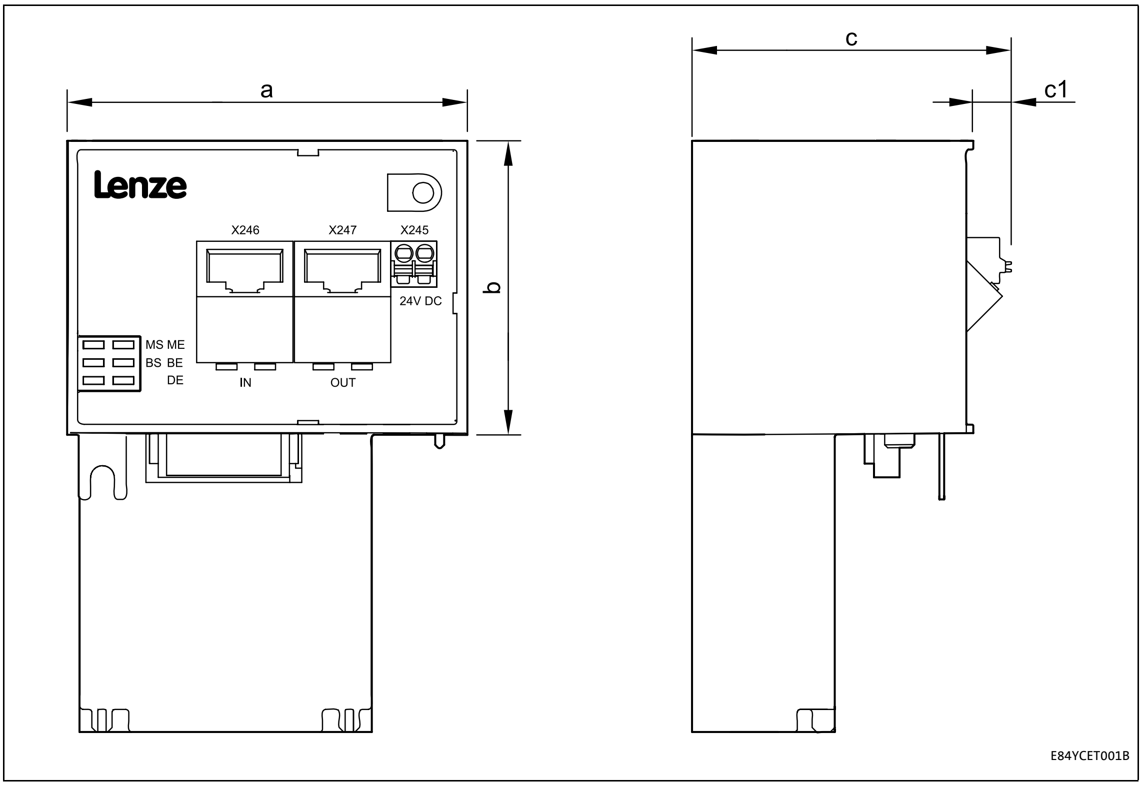
The communication times in the EtherCAT network depend on the ...

- processing time in the inverter;
- telegram runtime (baud rate / telegram length).

processing time within the inverter

Data	Processing time	
Process data for synchronised operation	0.2 ms	Processing time in the module
	+ 1 ... x ms	Runtime of the application task of the technology application used (tolerance)
Process data for non-synchronised operation	2 ms	Processing time in the module
	+ 1 ... x ms	Runtime of the application task of the technology application used (tolerance)
Parameter data	approx. 30 ms	Processing time
	+ 20 ms	Tolerance (typical) Some codes may require a longer processing time (see software manual/online help for the Inverter Drive 8400).

4.5 Dimensions



[4-2] Dimensions

Type	Dimensions [mm]			
	a	b	c	c1
E84AYCET	67	50	57	8

5 Installation

Stop!

Electrostatic discharge

Electronic components in the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Fieldbus communication is faulty or not possible.

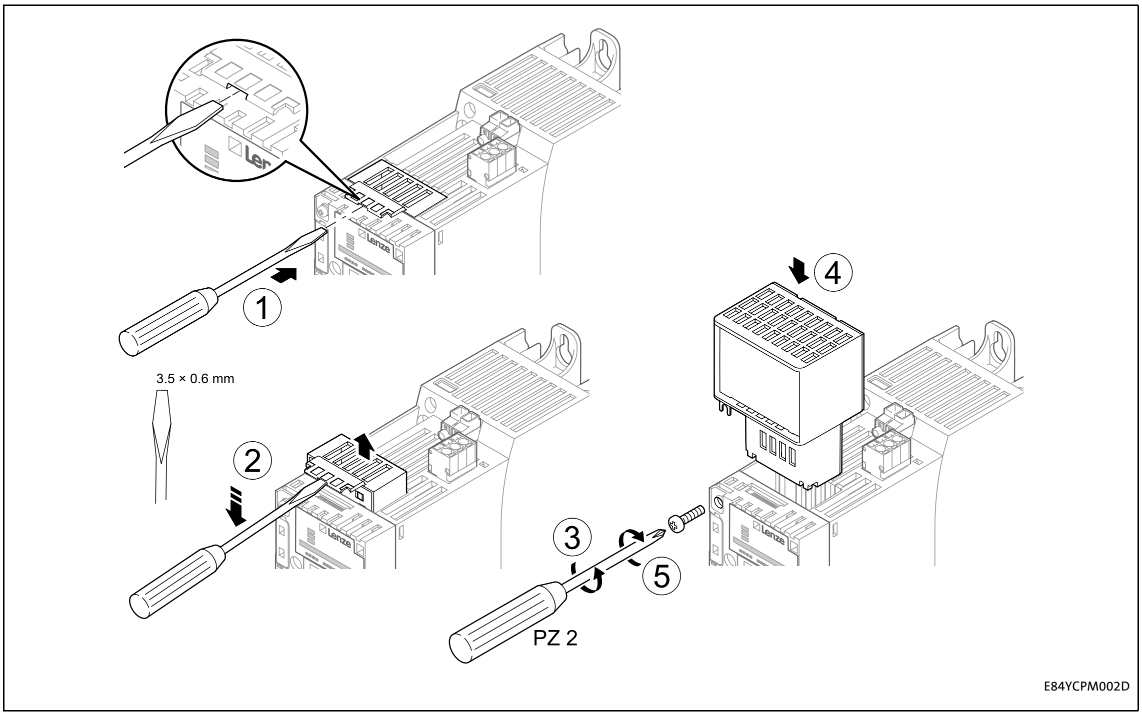
Protective measures:

Before touching the module, make sure that you are free of electrostatic charge.

5.1 Mechanical installation

The communication module can be plugged into the MCI slot or removed while the frequency inverter is switched on. When the module is plugged in, it is recognised automatically and checked for plausibility regarding its function and version.

5.1.1 Mounting for standard devices 0.25 kW and 0.37 kW

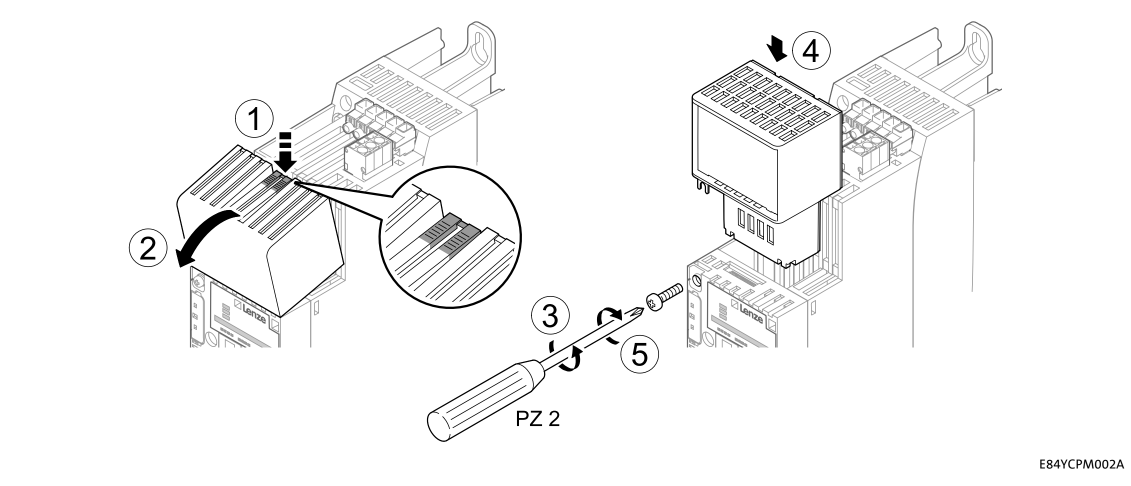


[5-1] Mounting for standard devices 0.25 kW and 0.37 kW

Mounting steps

1. Use a screwdriver to lever out the cover of the MCI slot of the standard device and remove it (1, 2).
2. Loosen the securing screw for the communication module at the standard device (3).
3. Insert the communication module into the MCI slot of the standard device (4).
4. Tighten the securing screw again (5).

5.1.2 Mounting for standard devices from 0.55 kW onwards

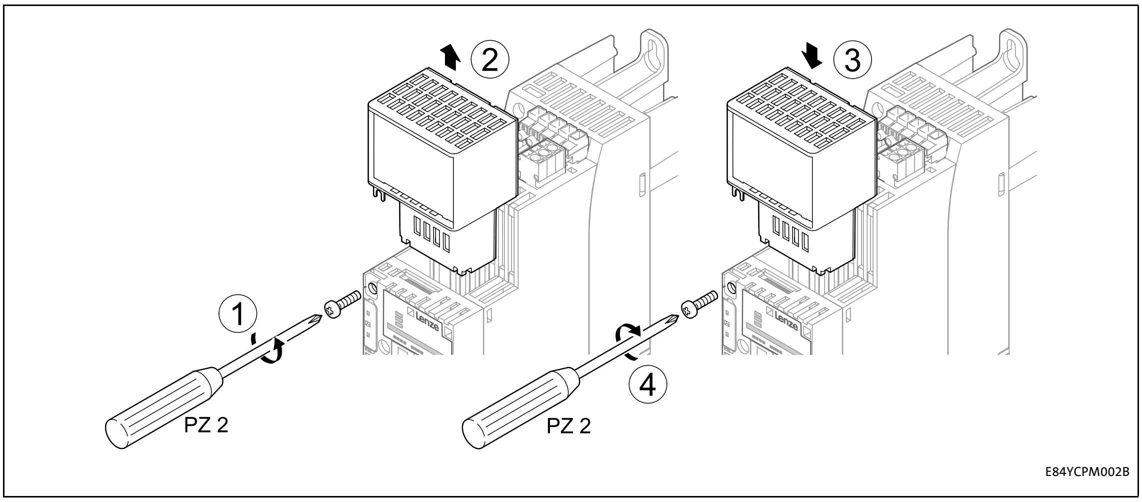


[5-2] Mounting for standard devices from 0.55 kW onwards

Mounting steps

1. Slightly impress the pressure surface of the top side of the MCI slot cover of the standard device (1).
2. Bend the cover forward and remove it from the standard device (2).
3. Loosen the securing screw for the communication module at the standard device (3).
4. Insert the communication module into the MCI slot of the standard device (4).
5. Tighten the securing screw again (5).

5.1.3 Replacing the communication module



[5-3] Replacing the communication module

Mounting steps

1. Loosen the securing screw for the communication module at the standard device (1).
2. Remove the communication module from the MCI slot of the standard device (2).
3. Insert the new communication module into the MCI slot of the standard device (3).
4. Tighten the securing screw again (4).

5.2 Electrical installation

Documentation for the standard device, control system, plant/machine
Observe the notes and wiring instructions stated.

5.2.1 EMC-compliant wiring

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by earthing both sides of the cable shield as well.

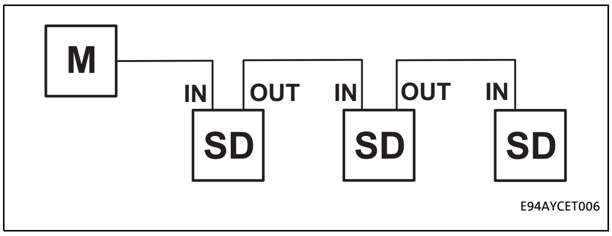
For this purpose, observe the following notes:

1. Remove the plastic sheath of the cable on a length of 2 cm.
2. Fasten the cable shield to the shield support of the basic device.

5.2.2 Network topology

An EtherCAT frame is sent through a pair of wires from the master to the slaves. The frame is forwarded from slave to slave until it has passed through all the devices. Finally, the last slave returns the frame to the master through a second pair of wires. In this way, EtherCAT always forms a logic ring topology, irrespective of the topology used.

Line topology



M = master

SD = slave device

[5-4] Line topology

The devices are interconnected successively.

In order to ensure trouble-free operation, it is required to assign and wire the EtherCAT inputs (IN) and EtherCAT outputs (OUT) correctly.

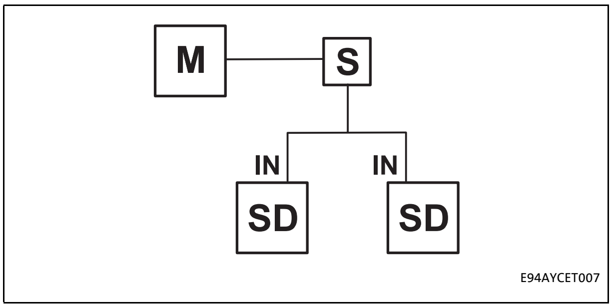
The receiving line is plugged into socket X246 (IN), the forwarding line into socket X247 (OUT).

The direction of data transmission is from the master to the slaves.

Tip!

The termination of the last EtherCAT node is effected automatically by the slave.

Switch topology



M = master

S = switch

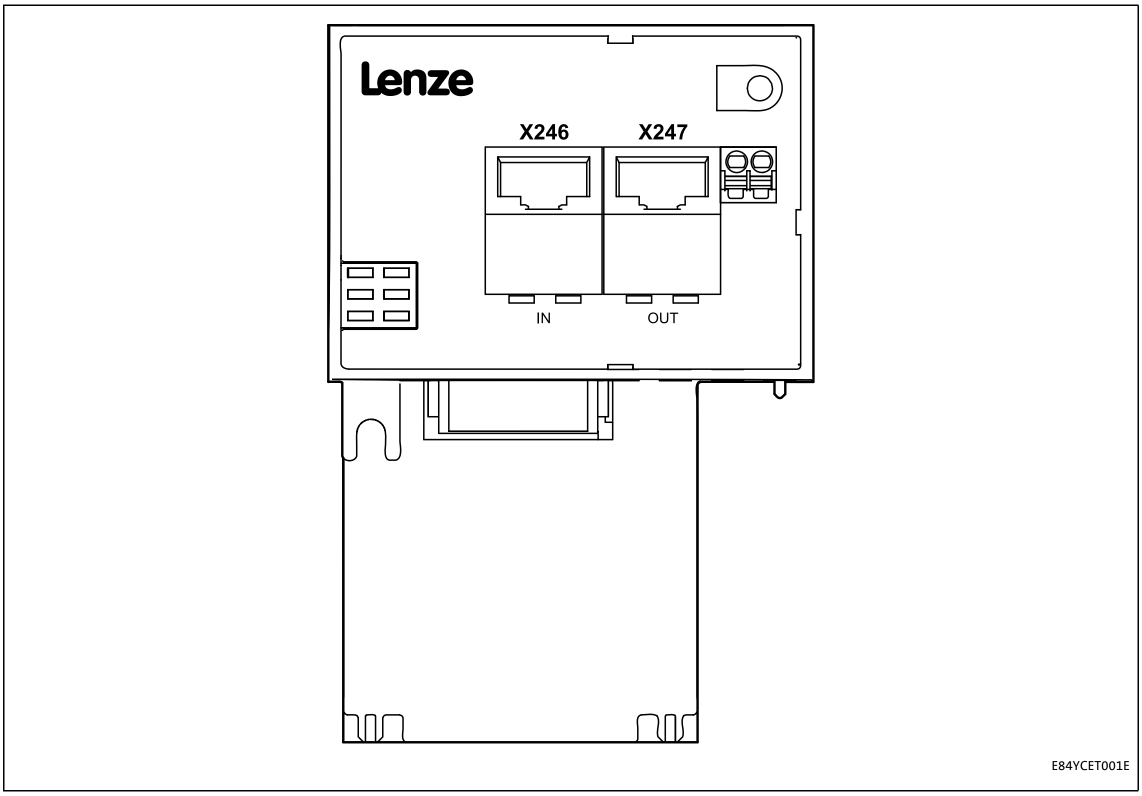
SD = slave device

[5-5] Switch topology

The wiring can also be carried out in a star structure via an appropriate switch. For this, observe the additional runtimes.

5.2.3 EtherCAT connection

EtherCAT is connected via the RJ45 sockets **X246** (IN) and **X247** (OUT).



[5-6] EtherCAT connection

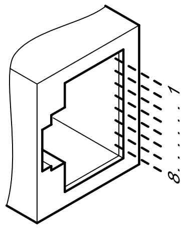
A standard Ethernet patch cable is suitable for connecting the communication module to the EtherCAT fieldbus.

▶ Specification of the Ethernet cable (📖 29)

Note!

In order to prevent damage to the RJ45 socket, plug or remove the Ethernet cable connector straight (at a right angle) into/from the socket.

Pin assignment of the RJ45 sockets

RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Tip!

The EtherCAT interfaces are provided with an auto MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite EtherCAT interface and irrespective of the type of cable used (standard patch cable or crossover cable).

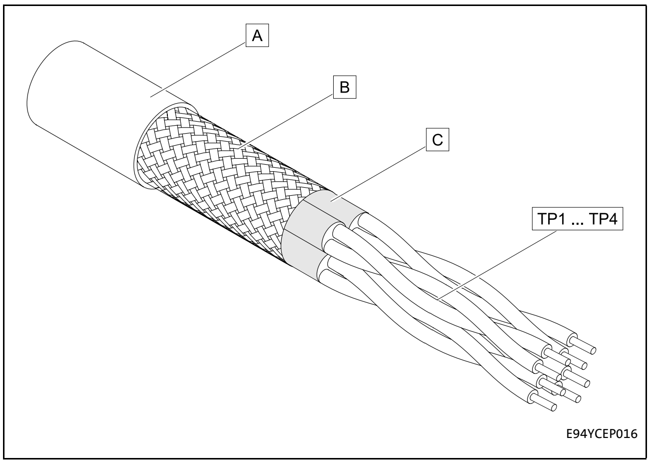
5.2.4 Specification of the Ethernet cable

Note!

Only use cables that correspond to the given specifications.

Specification of the Ethernet cable	
Ethernet standard	Standard Ethernet (acc. to IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (for 100 MHz and 100 m each)
Return loss	10 dB (100 m each)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 ... TP4 Twisted core pairs 1 ... 4

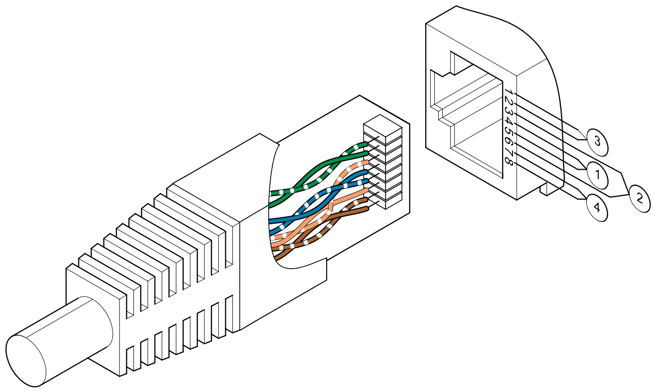
[5-7] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

The use of 4-pole Ethernet cables according to industrial standard is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6.



[5-8] Ethernet plugs in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	white / green	white / orange
	2	Tx -	green	orange
2	3	Rx +	white / orange	white / green
1	4		blue	blue
	5		white / blue	blue / white
2	6	Rx -	orange	green
4	7		white / brown	white / brown
	8		brown	brown

5.2.5 External voltage supply

The communication module can be supplied externally with voltage via separate supply cables at the 2-pole plug connector **X245**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

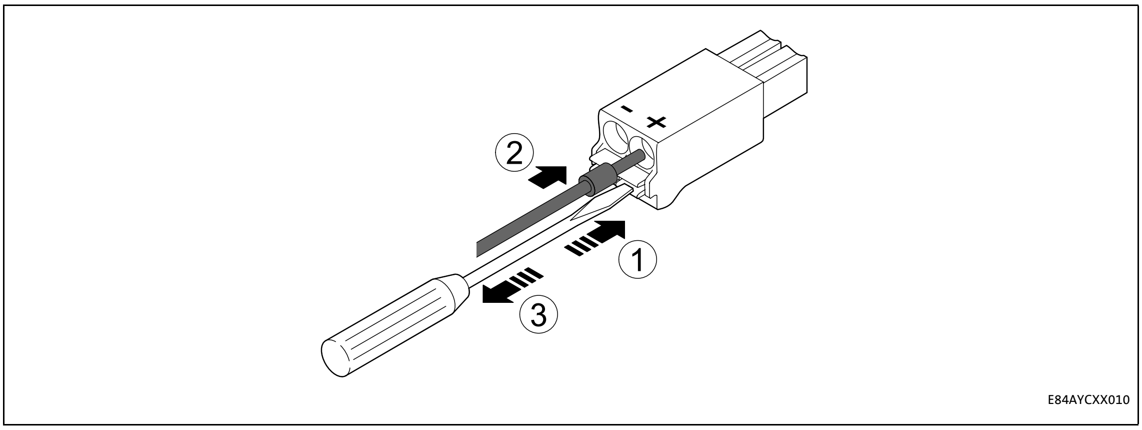
External voltage supply of the communication module is necessary if the bus communication is to be continued in the event of a failure of the supply of the standard device.

Access to parameters of a standard device disconnected from the mains is not possible.

Wiring plug connector X245

Stop!

Only wire the plug connector if the standard device is disconnected from the mains.



[5-9] Wiring of the 2-pole plug connector with spring connection

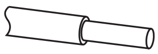
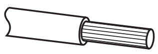
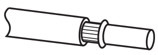
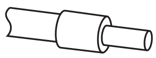
How to wire the plug connector with spring connection:

1. Place a screwdriver in the notch beneath the contact slot and keep it pressed.
2. Insert the supply cable in the contact slot.
3. Remove the screwdriver from the notch.

Assignment of the X245 plug connector

Designation	Description
+	U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %) I = 140 mA
-	Reference potential for the external voltage supply

Terminal data

Area	Values	
Electrical connection	2-pole plug connector with spring connection	
Possible connections	Fixed:	
		0.2 ... 1.5 mm^2 (AWG 24 ... 16)
	Flexible:	
		Without wire end ferrule 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, without plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, with plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
Stripping length	10 mm	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The codes of the inverter and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

▶ Parameters of the communication module (📖 97)

6.1 Before initial switch-on

Stop!

Before you switch on the Inverter Drive 8400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Configuring the Controller (EtherCAT master)

The Controller (EtherCAT master) must be configured before communication with the communication module is possible.

In order to configure EtherCAT networks, you always need a configuration software for the Controller, e.g.:

- Lenze »PLC Designer«
- Beckhoff »TwinCAT«

These are software systems for the programming of control programs, EtherCAT configuration, real-time execution, and diagnostics.

The basic parameters of the communication module are stored in the internal configuration memory and can be used by the master for the node identification.

For the node search (fieldbus scan), the corresponding device descriptions of the Lenze device family are used.

6.2.1 Installing device description files

The current XML device description file required for configuring the EtherCAT node can be found in the download area at:

www.Lenze.com

The **Lenze_Inverter_8400_IO_yyyymmdd.xml** device description file has to be installed via the EtherCAT configuration software.

Wildcards in the file name	
yyyy	Year
mm	Month
dd	Day

The description file includes all EtherCAT capable devices of the Inverter Drives 8400 series (Inverter Drives 8400 with EtherCAT V01.xx and V02.xx, 8400 motec with EtherCAT V01.xx).

6.2.2 Automatic device identification

For a faultless integration of the EtherCAT slaves into a master configuration it is necessary to select the correct Lenze device in the EtherCAT configuration software.

Each EtherCAT node is identified unambiguously by the configuration software by means of the product code (equal to the CoE object I-1018.2), the manufacturer's identification mark (0x3B), and the main software version of the communication module.

▶ Identification (📖 12)

▶ Implemented CoE objects (📖 79)

In order that the configuration software select the configuration from the device description file specific for the EtherCAT node, the product code is automatically set in the identity object. An update is made after switching off/on the voltage supply.

During initialisation, the product code is transferred to the EtherCAT master. On the basis of this identification, the master can accept the corresponding settings from the device description.

Product codes for Inverter Drives 8400

The product code defines the following Inverter Drives 8400 in the device description files:

Product code [dec]						Meaning
8	4	0	0	2	2	Inverter Drive 8400 StateLine
8	4	0	0	2	3	Inverter Drive 8400 HighLine
8	4	0	0	2	4	Inverter Drive 8400 TopLine
8	4	0	0	2	5	Inverter Drive 8400 TopLine P (with CiA402)

6.2.3 Configuring process data

Inverter Drives 8400 support the configuration of max. 16 process data words (max. 32 bytes) per direction.

The process data configuration is determined during the initialisation phase of the master (PDO mapping).

The process data configuration is predefined in the device description file for each application.

The process data length can be adjusted by the user if required.

6.2.4 Determining the cycle time

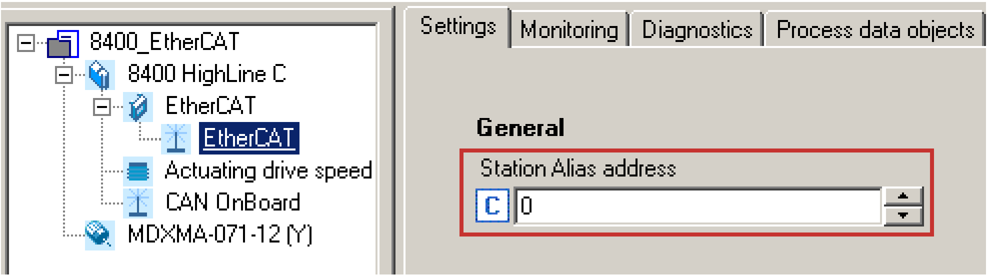
The process data objects (PDO) are transferred cyclically between the EtherCAT master and the slaves (inverters).

The cycle time is set with the EtherCAT configuration software.

6.3 Address allocation

The EtherCAT nodes are normally addressed via a fixed 16-bit address defined by the EtherCAT master. During start-up, the master assigns this address to each node, depending on the physical order in the EtherCAT network. The address is not saved and is lost when the device is switched off.

Via the **Station alias address** input field you can assign a fixed address to the EtherCAT slave.



Note!

- The station alias address must be unambiguous and may only be assigned once within the EtherCAT network.
- Use the same station alias address in the EtherCAT master and in the slave.

Valid address range: 0 ... 32767

- Address 0 means that no station alias address is assigned.
- Impermissible addresses are marked in red in the input field.
- The address is written to code C13899.

In addition, specify the use of the fixed addressing on the master.

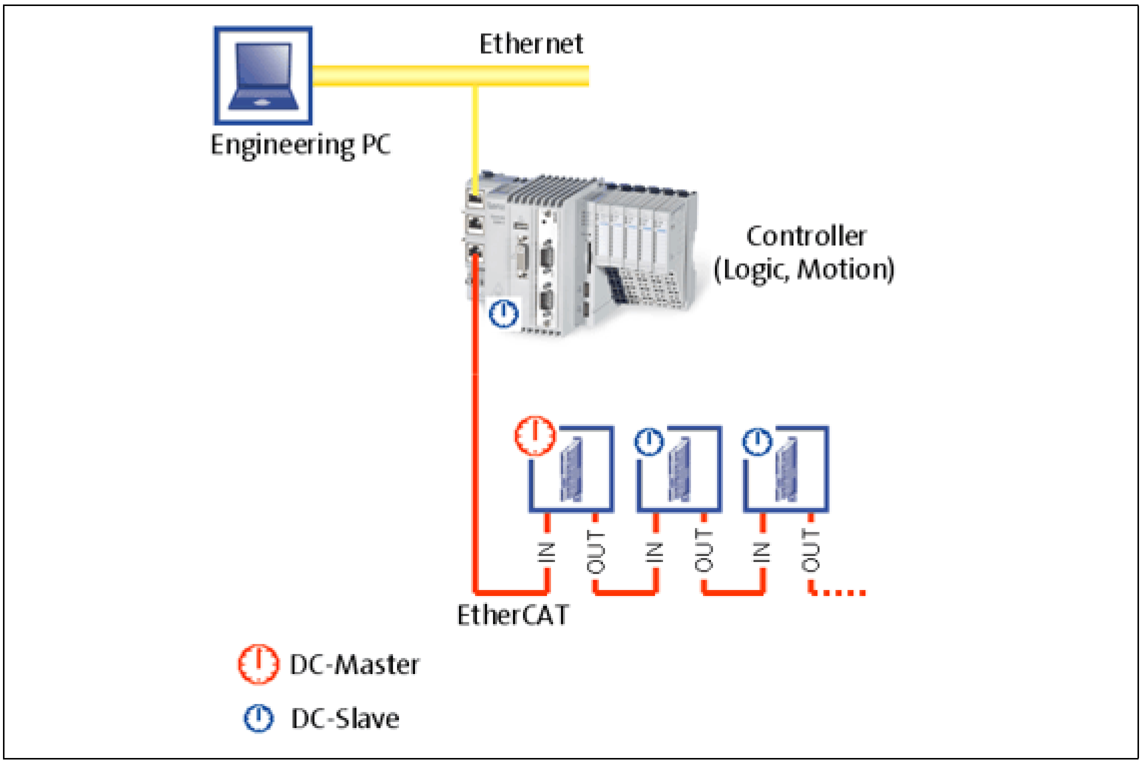
The address assigned by the master is displayed under code C13864.

6.4 Synchronisation with "Distributed Clocks" (DC)

The "Distributed clocks" (DC) functionality enables exact time synchronisation for applications in which several axes perform a coordinated movement simultaneously. Data are incorporated synchronously with the PLC program. During DC synchronisation, all slaves are synchronised with a reference clock, the so-called "DC master".

Note!

- DC synchronisation is absolutely required for Motion applications.
- DC synchronisation can also be used for Logic applications.
- Not all slaves support the DC functionality.
- On order to be able to use the DC functionality, the first slave connected to the EtherCAT master (e.g. Lenze Controller) must have **DC master capability**.
  When further slaves are connected, DC-capable and non-DC-capable devices can be mixed.
- The first EtherCAT slave after the Lenze Controller must be the **DC master** that supplies the other EtherCAT nodes (incl. Controller) with the exact time.



[6-1] Example: "Distributed clocks" in the EtherCAT bus system with Lenze Controller 3231 C

The DC synchronisation is set with the EtherCAT configuration software.

"Control technology EtherCAT" communication manual

Here you can find some detailed information about the EtherCAT configuration and the commissioning of Lenze devices in the EtherCAT network.

6.4.1 DC configuration in the master

By default, the application of the DC synchronisation is deactivated in the device description (📖 34).

Parameterise the DC synchronisation in the EtherCAT configuration software (»PLC Designer«, »TwinCAT«).

Set the synchronisation cycle time in the master. It is mainly defined by the processing time of the master and the slaves.

Note!

The synchronisation cycle time ...

- must be an integer multiple of 1 ms;
- may be maximally 15 ms.

6.4.2 DC configuration in the Inverter Drive 8400 (slave)

Note!

The settings of the parameter sync cycle time (C01121), sync phase position (C01122), sync tolerance (C01123) and sync PLL increment (C01124) common for the Lenze system bus (CAN) cannot be made for EtherCAT. These values are automatically calculated by the EtherCAT communication module and set internally in the inverter.

In order to use the DC synchronisation in the Inverter Drive 8400, select the sync source by means of the standard device code C01120:

Selection **4: MCI** (synchronisation via MCI (communication module))

6.4.3 Response of the Lenze EtherCAT nodes during start-up

Code C13883 indicates whether the DC synchronisation for the communication module has been activated.

If the DC synchronisation is used, the communication module only changes to the "Operational" state when the standard device has adapted its phase position to the DC signal. This process may take several seconds.

Note!

- If the communication module does not change to the "Operational" state, there might be an error in the configuration or in the EtherCAT wiring.
- The communication module compares the cycle time defined by the EtherCAT master to the internal processing time (1 ms) of the standard device. The synchronisation cycle time in the master must be identical with or an integer multiple of 1 ms.
- Furthermore it is checked whether the sync source selection in standard device code C01120 is correct.
- Further information can be found in the status information or emergency messages of the master.

The state of the standard device synchronicity is displayed under code C13884.

6.5 Establishing an online connection with the »Engineer«

With the »Engineer« you can establish an online connection to the individual field devices.

When an online connection has been established, you can for instance carry out parameter settings directly in the field device or diagnose the field device.

Stop!

If parameters in the »Engineer« are changed while the Engineer is connected online to the field device, the changes are directly accepted to the device!

Note!

To go online, the EtherCAT bus at least has to be in the "Pre-Operational" state.

The functions for establishing/cancelling an online connection in the »Engineer« can be executed via the **Online** menu:

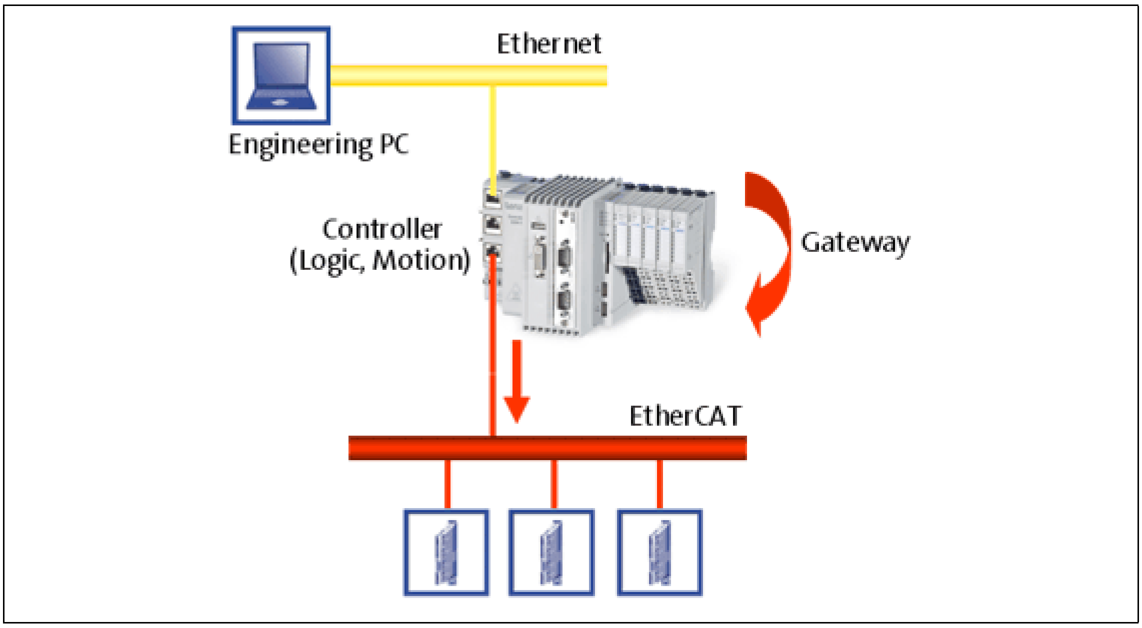
Menu command	Shortcut
Online → Go online	**<F4>**
Online → Set communication path and go online	
Configuring a bus connection: • Gateway Controller -> configure EtherCAT (📖 41) • Gateway Controller -> configure EtherCAT ADS (Beckhoff) (📖 43)	
Online → Go offline	**<Shift>+<F4>**

Documentation for the Lenze »Engineer«

Here you'll find further detailed information about how to establish an online connection.

6.5.1 Gateway Controller -> configure EtherCAT

The Lenze Controller provides a gateway function to establish an online connection to a field device via EtherCAT.

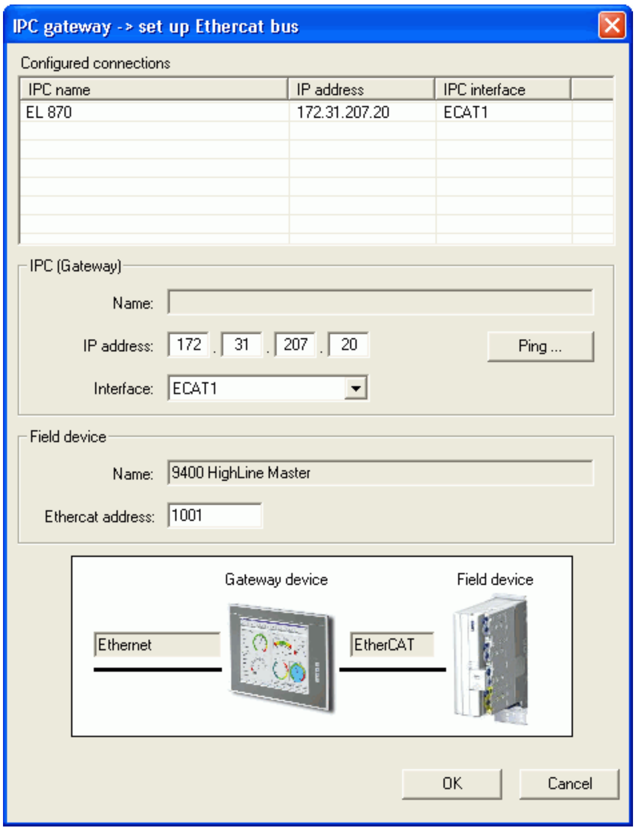


[6-2] Example: EtherCAT bus system with a Lenze Controller 3231 C as gateway

How to configure an online connection to a field device which is connected to the Lenze Controller via EtherCAT:

1. Go to the *Communication path* dialog box and the *Bus connection* list field, and select the entry "Gateway Controller -> EtherCAT" there.
2. Click **Search/Enter...**.

   The *Gateway Controller -> Set up EtherCAT bus* dialog box is shown:

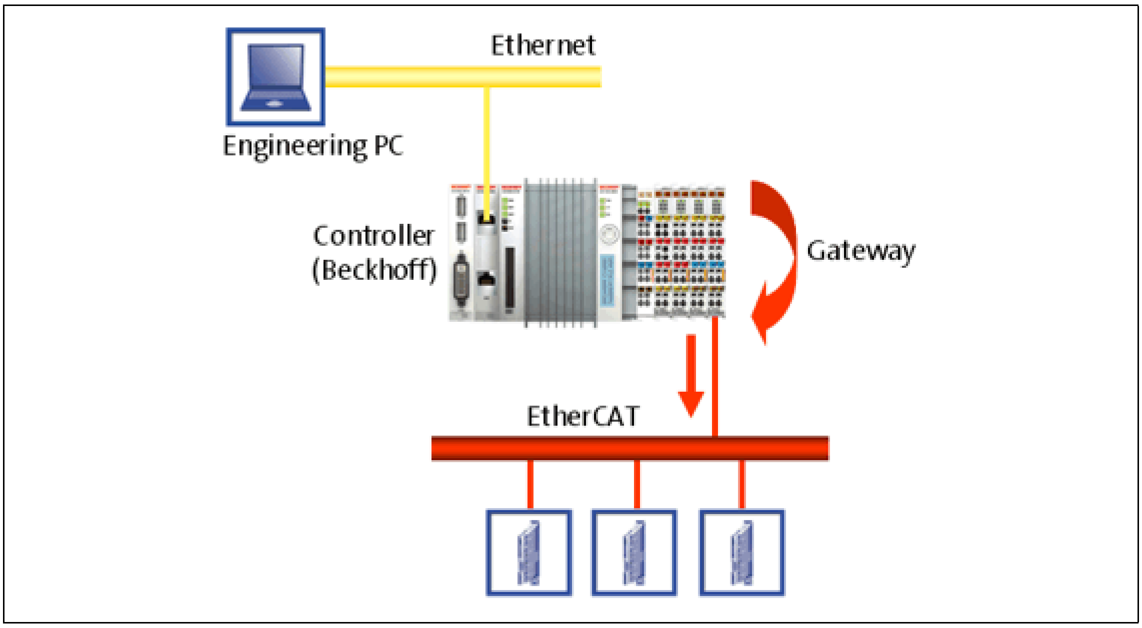


3. Enter the **IP address** of the Controller.

   By clicking the **Ping** button, you can carry out a simple test which verifies whether a device can actually be reached via the IP address set
4. Click **OK**.
   - The *Enter IP address* dialog box is closed.
   - In the *Communication path* dialog box in the **Device access path** column, the corresponding device access path is shown (e.g. "IPC:172_31_207_254.ECAT.ecat1.dev1001.").

6.5.2 Gateway Controller -> configure EtherCAT ADS (Beckhoff)

The **Gateway EtherCAT ADS** bus connection makes it possible to establish an online connection to a Lenze inverter that is connected to a Beckhoff Controller via EtherCAT (gateway function).



[6-3] Example: EtherCAT bus system with a Beckhoff Controller as Gateway

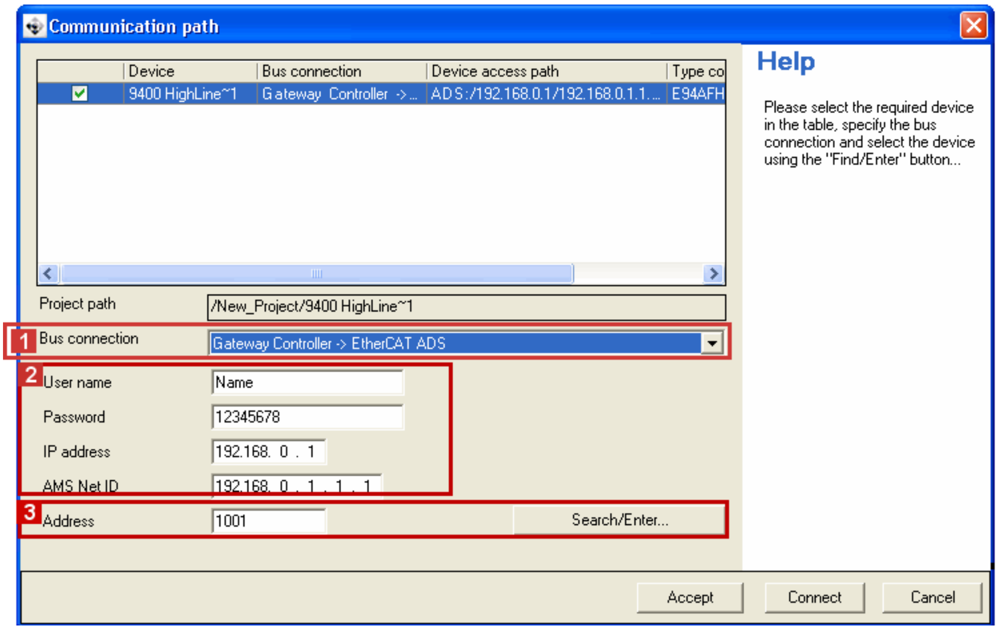
How to configure an online connection to a field device which is connected to a Beckhoff Controller via EtherCAT:

1. Highlight the project root in the project.

   Alternatively: Create a new project or carry out a fieldbus scan.
2. Execute the menu command **Insert → Insert device detected online**.
3. Select **Gateway Controller -> EtherCAT ADS** as bus connection.
4. Configure access data:
   - Configure the access data applicable to the Controller via the **Insert address** button.
   - The **Search** button initiates the Controller to display the fieldbus nodes connected to the EtherCAT segment.

How to use the EtherCAT ADS communication path:

1. Highlight the desired inverter, to which a gateway connection via EtherCAT ADS is to be established, in the project tree.
2. Call the menu command **Online → Set communication path and go online**.



3. Select 1 **Gateway Controller -> EtherCAT ADS** as bus connection.
4. Enter the access data applicable to the Controller in area 2.

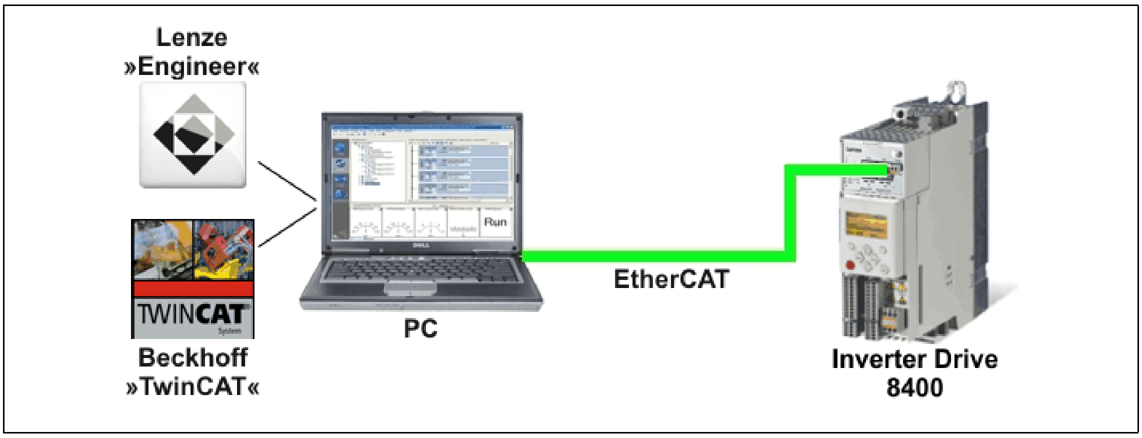
   Enter the user name, password, and the IP address and the AMS Net ID of the EtherCAT interface of the Controller.
5. In area 3, specify the EtherCAT address of the field device to which the online connection is to be established.

   Alternatively you can click the **Search/Enter** button which calls the *Select Device Access Path* dialog window. By this, the »Engineer« initiates the Controller to display the devices detected on the EtherCAT segment.

6.6 EtherCAT ADS communication parameters in »TwinCAT« and »Engineer«

In the following, two example structures are used to describe where you can find the EtherCAT ADS communication parameters in the Beckhoff »TwinCAT« and in the Lenze »Engineer«EtherCAT.

6.6.1 Example: Structure without a Beckhoff Controller

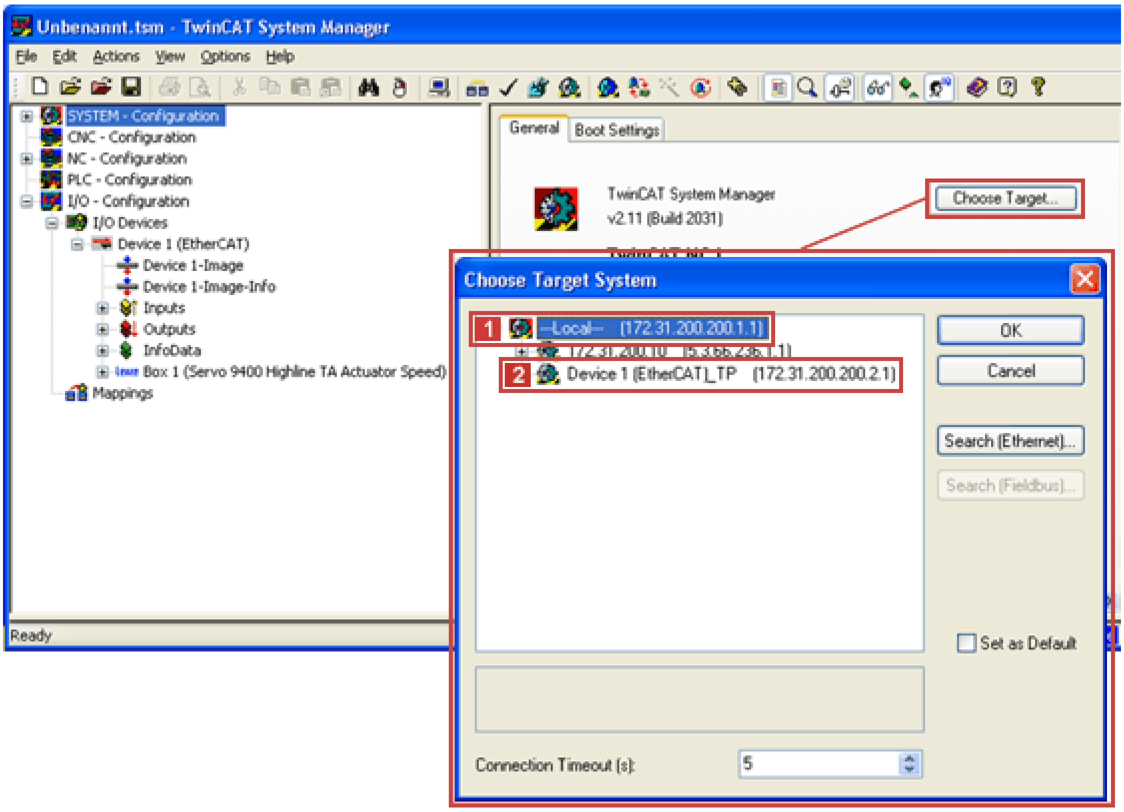


[6-4] Example: EtherCAT bus system without Beckhoff Controller

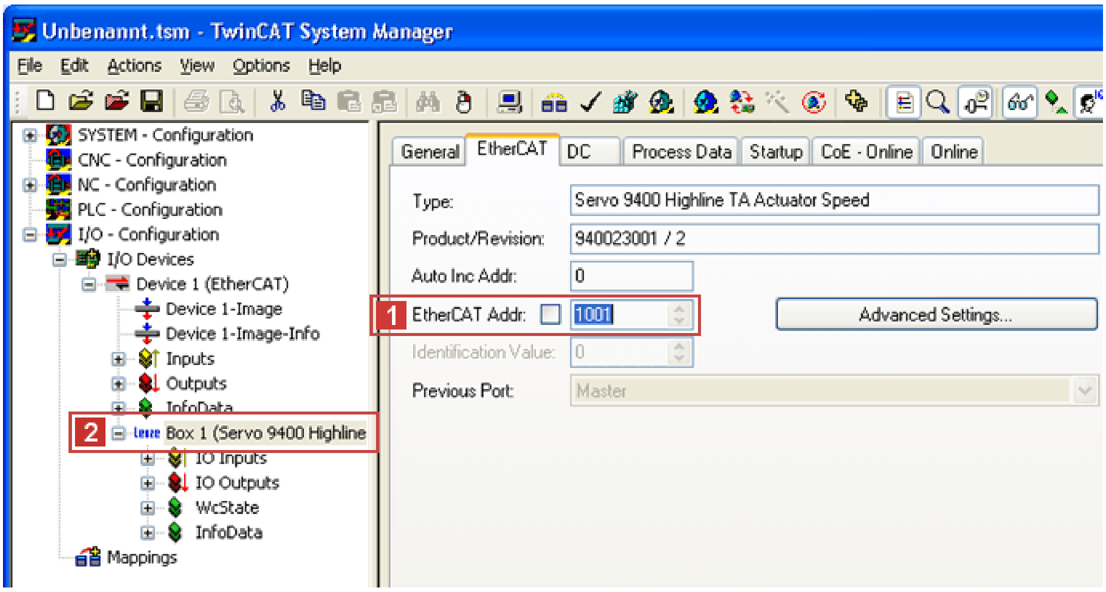
The Beckhoff Soft-PLC runs on the Microsoft Windows XP PC on which the Beckhoff »TwinCAT« and the Lenze »Engineer« are installed as well.

Display of the communication parameters in »TwinCAT«

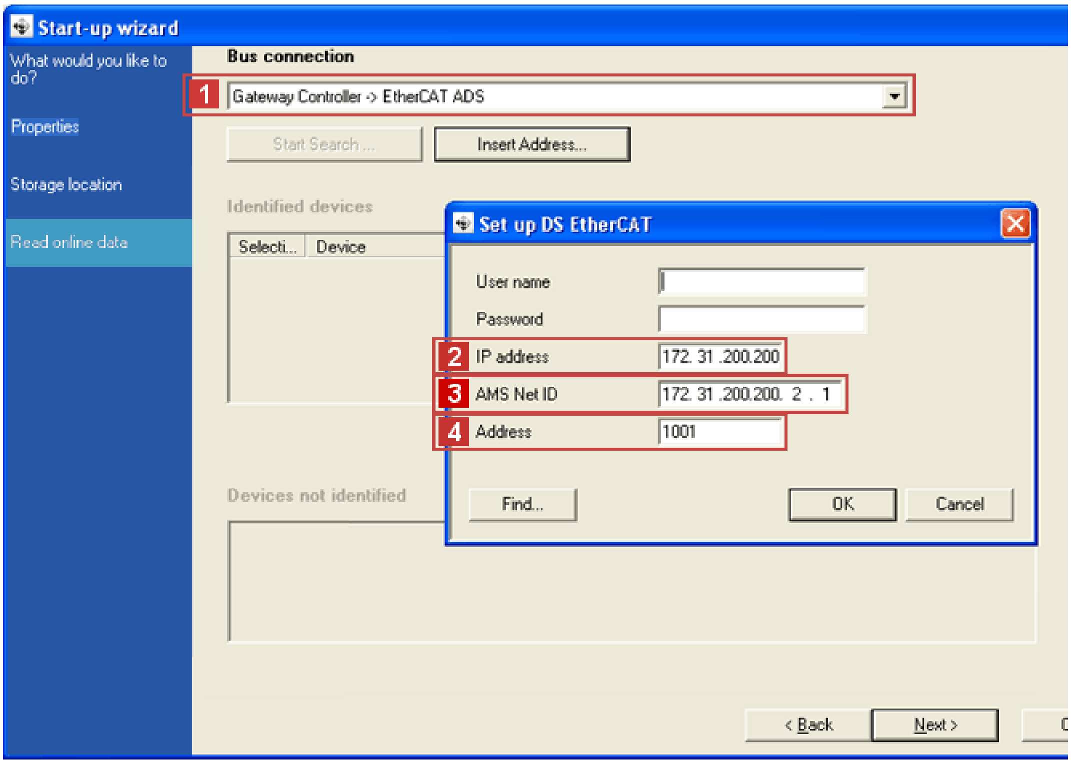
The communication parameters 1 **IP address** (here ’172.31.200.200’) and 2 **EtherCAT Master Net ID** (here ’172.31.200.200.2.1’) can be found under the target system selection:



The EtherCAT 1 **Slave address** (here ’1001’) can be found under the **EtherCAT** tab of the 2 EtherCAT slave:



Online device identification in the »Engineer« start-up wizard



In the »Engineer« start-up wizard under the 1 **Gateway Controller -> EtherCAT ADS** bus connection, a field device was detected online:

2 **IP address:** 172.31.200.200

3 **EtherCAT Net ID:** 172.31.200.200.2.1

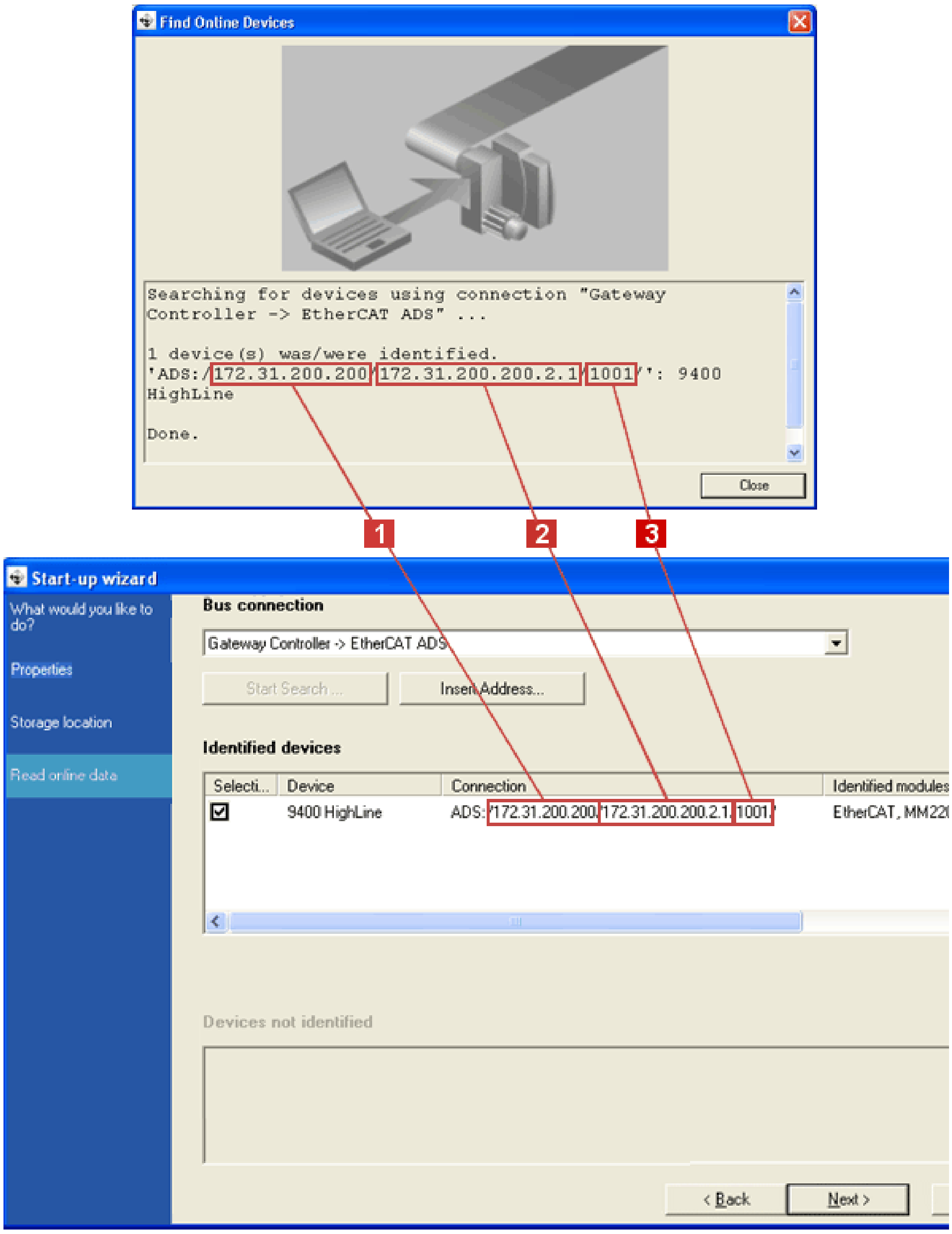
4 **EtherCAT slave address:** 1001

Display of the identification of the field device detected in the »Engineer« start-up wizard:

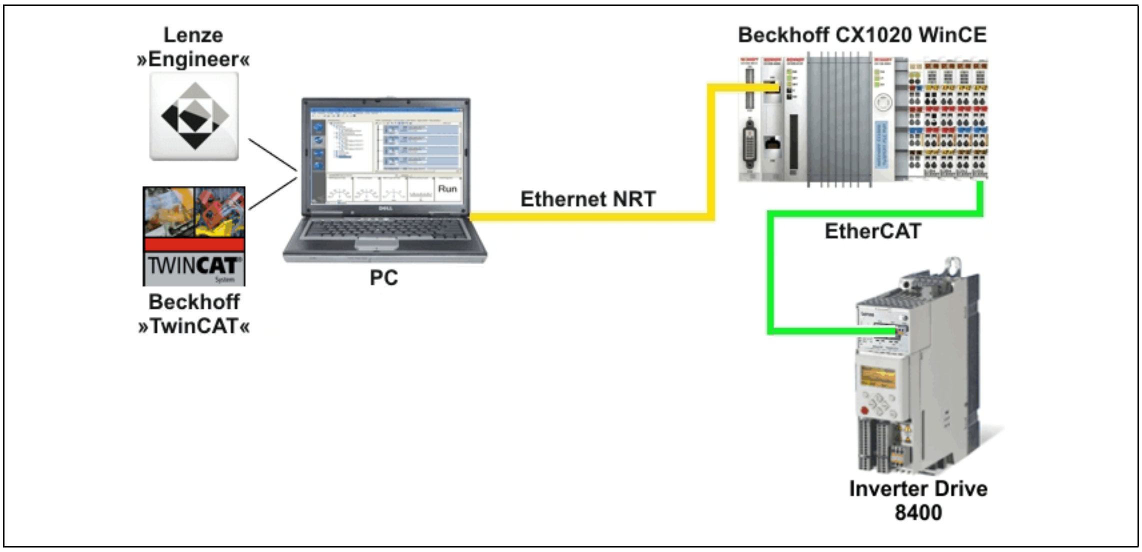
1 **IP address:** 172.31.200.200

2 **EtherCAT Net ID:** 172.31.200.200.2.1

3 **EtherCAT slave address:** 1001



6.6.2 Example: Structure with a Beckhoff DIN rail IPC CX1020

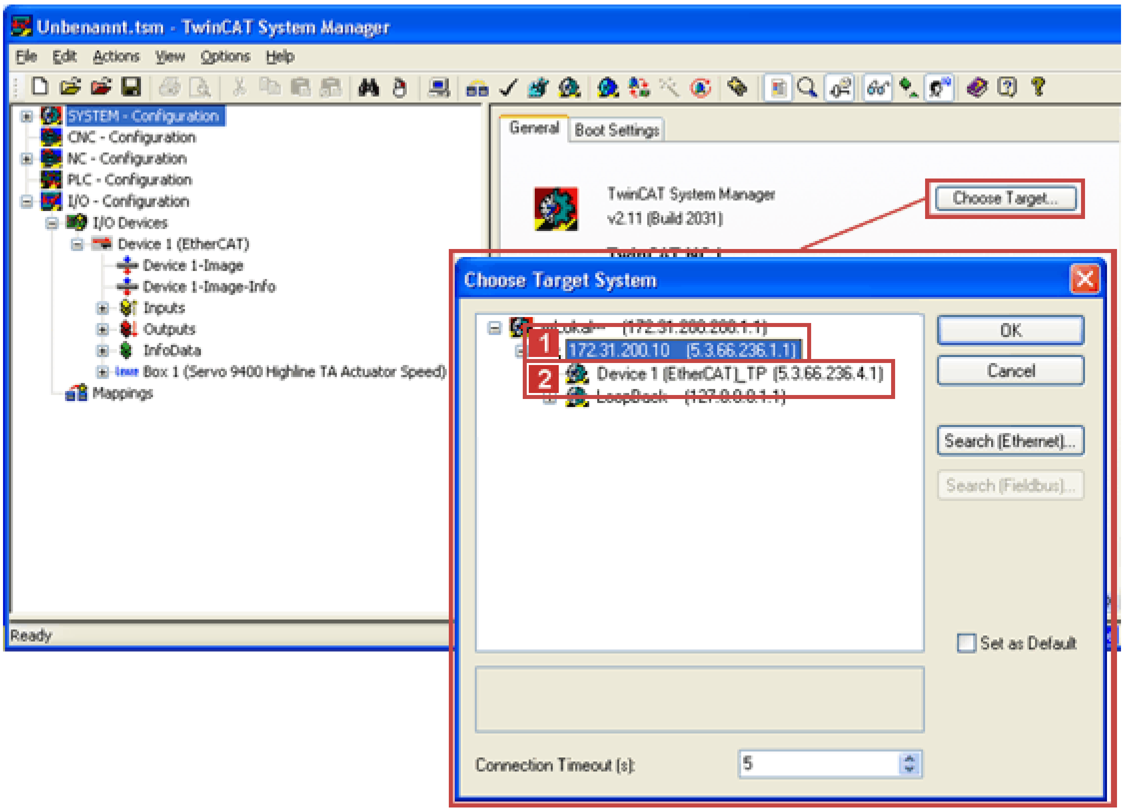


[6-5] Example: EtherCAT bus system with Beckhoff Controller

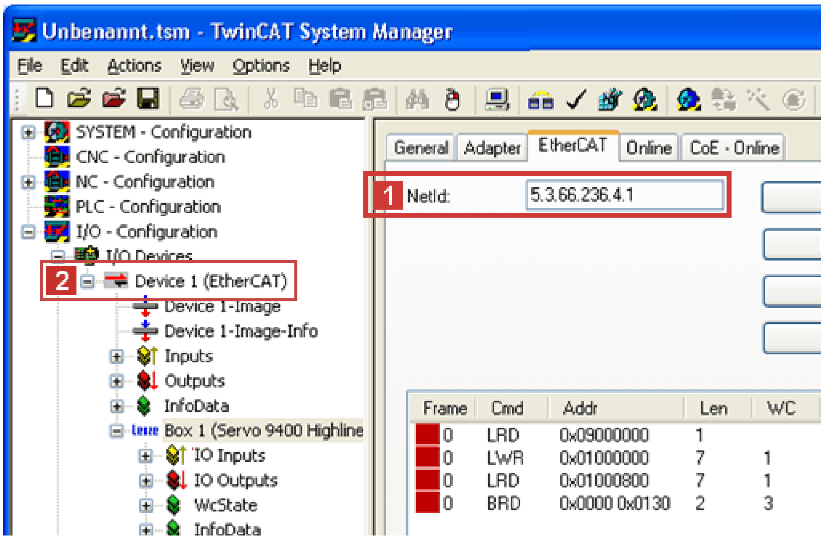
A Beckhoff DIN rail IPC with the Microsoft Windows CE operating system is used. The Beckhoff »TwinCAT« and the Lenze »Engineer« are installed on a Windows XP PC.

Display of the communication parameters in »TwinCAT«

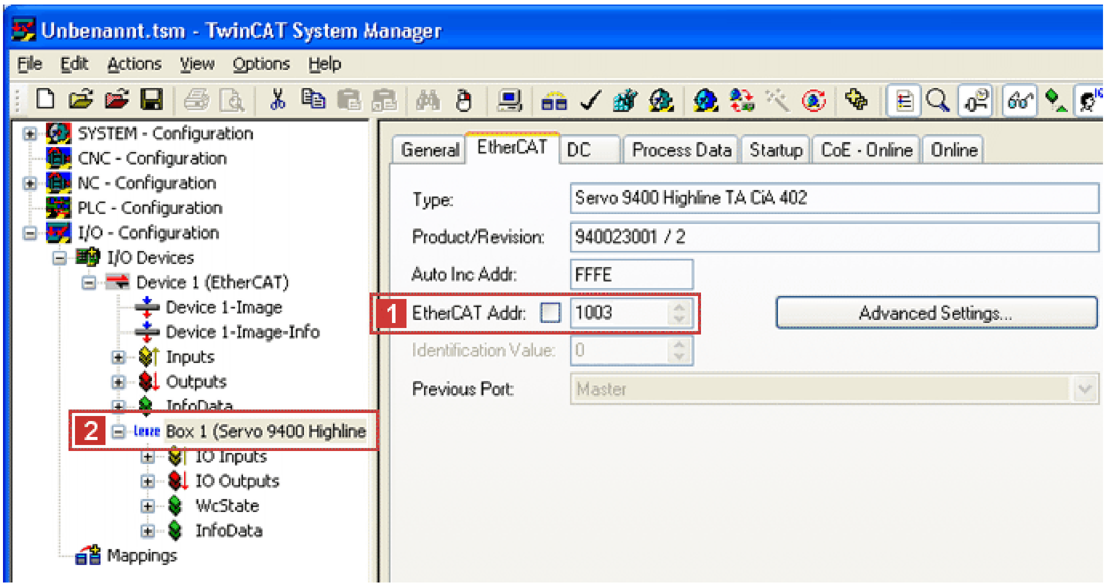
The communication parameters 1 **IP address** (here '172.31.200.10') and 2 **EtherCAT Master Net ID** (here '5.3.66.236.4.1') can be found under the target system selection:



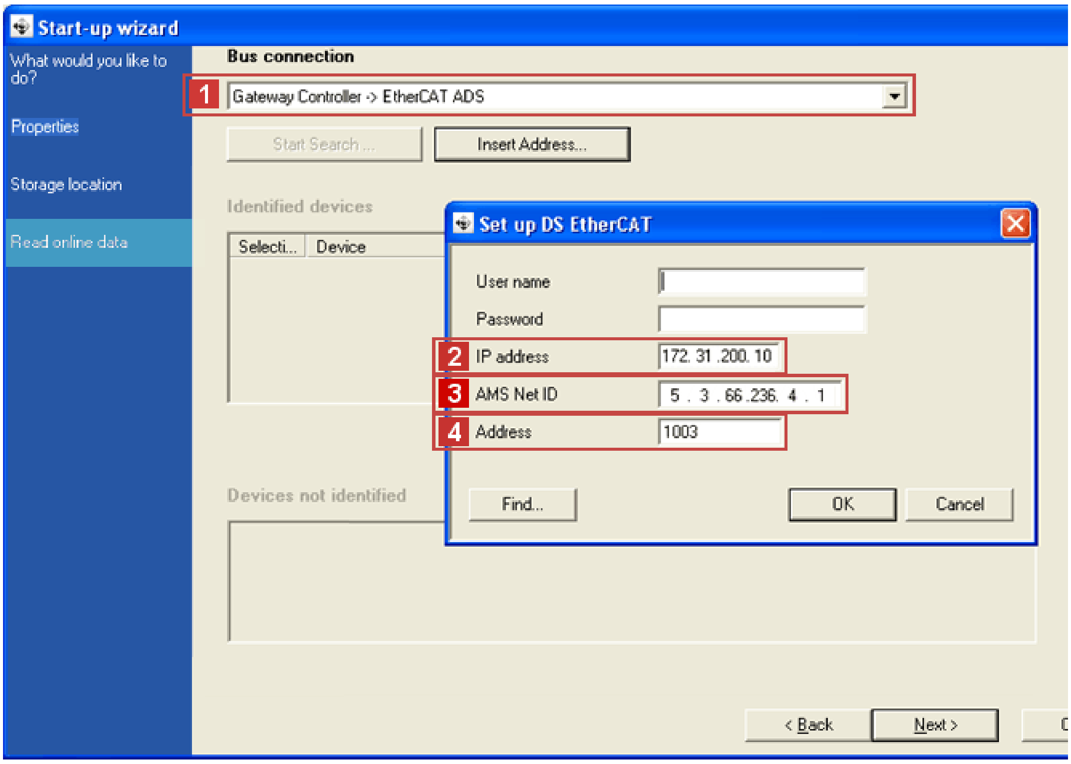
The 1 **EtherCAT Master Net ID** (here '5.3.66.236.4.1') can also be found under the **EtherCAT** tab of the 2 EtherCAT master:



The EtherCAT 1 **slave address** (here '1003') can be found under the **EtherCAT** tab of the 2 EtherCAT slave:



Online device identification in the »Engineer« start-up wizard



In the »Engineer« start-up wizard under the 1 **Gateway Controller -> EtherCAT ADS** bus connection, a field device was detected online:

2 **IP address:** 172.31.200.10

3 **EtherCAT Net ID:** 5.3.66.236.4.1

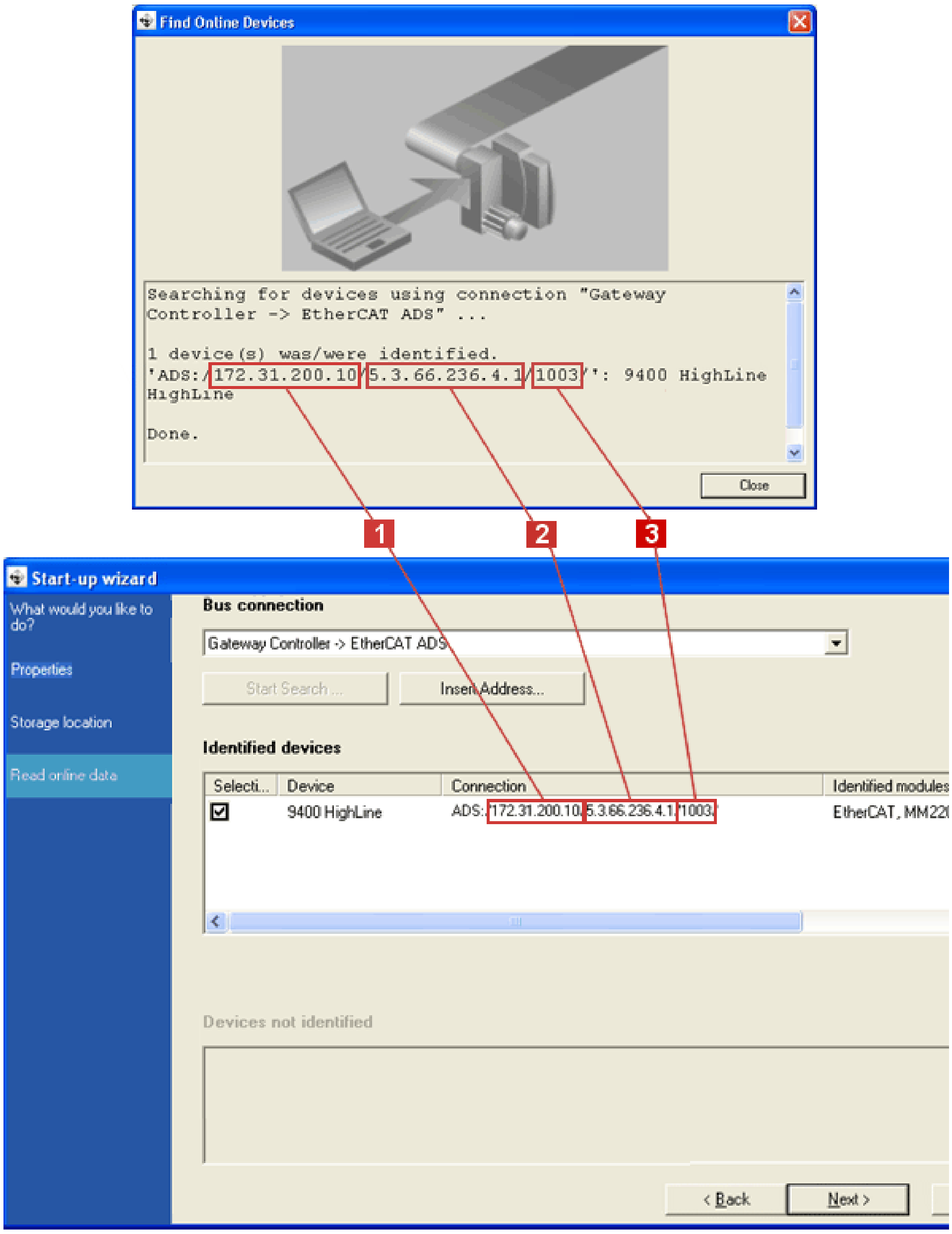
4 **EtherCAT slave address:** 1003

Display of the identification of the field device detected in the »Engineer« start-up wizard:

1 **IP address:** 172.31.200.10

2 **EtherCAT Net ID:** 5.3.66.236.4.1

3 **EtherCAT slave address:** 1003



6.7 Initial switch-on

Documentation for the standard device

Observe the safety instructions and residual hazards stated.

Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

After communication has been established, the externally supplied module operates independently of the power on/off state of the standard device.

Activating changed settings

In order to activate changed settings ...

- execute the device command "11: Save all parameter sets" via the standard device code **C00002** and ...
- then execute a "reset node" of the node or switch off the voltage supply of the communication module and switch it on again.

Protection against uncontrolled restart

After a fault (e.g. short-time mains failure), the restart of a drive is not always wanted and - in some cases - even not allowed.

In the Lenze setting for Inverter Drives 8400, the restart protection is active.

Via the standard device code **C00142** ("Autostart option") you can set the restart behaviour of the inverter:

C00142 = 9 (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- Bit 0 (inhibit if device is ON) and bit 3 (inhibit in case of undervoltage) are set.
- An explicit inverter enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X4/RFR.

C00142 = 8 (enabled)

- In order to directly enable the device at switch-on, bit 0 must be set to zero (FALSE).
- An uncontrolled restart of the drive is possible.

7 Data transfer

Compared with conventional Ethernet, the collision-free transfer of frames on the fieldbus makes EtherCAT a real-time capable bus system.

Communication is always initiated by the EtherCAT master, e.g. a Lenze Controller. A frame sent by the master passes through all EtherCAT slaves. The last slave of the communication chain sends the frame back to the EtherCAT master. On the way back, the frame is directly sent to the master, without being processed in the slaves.

EtherCAT transmits data in so-called "EtherCAT frames". The EtherCAT nodes only extract the data intended for them while the EtherCAT frame passes through the device. At the same time output data are inserted into the frame while it passes through the device. Read and write accesses are only executed on a small section of the entire EtherCAT frame – the datagrams. Therefore it is not necessary to receive the complete frame before it can be processed. Processing starts as soon as possible.

EtherCAT transmits process data, parameter data, configuration data, and diagnostic data between the EtherCAT master and the inverters (slaves) that are part of the fieldbus. The data are transmitted via corresponding communication channels depending on their time-critical behaviour (see Process data transfer (🕮 59) / Parameter data transfer (🕮 68)).

7.1 EtherCAT-Frames

EtherCAT frames have the following structure:

Ethernet header			Ethernet data				FCS
48 bits	48 bits	16 bits	11 bits	1 bit	4 bits	48 ... 1498 bytes	32 bits
Destination	Source	EtherType	Frame header			Datagrams	
			Length	Reserved	Type		

Ethernet header

The Ethernet header contains the following information:

- Target address of the EtherCAT frame (destination)
- Source address of the EtherCAT frame (source)
- Type of the EtherCAT frame (EtherType = 0x88A4)

Ethernet data

The Ethernet data contain the following information:

- Length of the datagrams within the EtherCAT frame (Length)
- One reserved bit (Reserved)
- Type of the datagrams within the EtherCAT frame (Type)
- EtherCAT datagrams (Datagrams)

FCS

Checksum of the EtherCAT frame

7.2 EtherCAT datagrams

Read and write accesses are always only executed in one small section of the complete EtherCAT frame – the datagrams.

EtherCAT datagrams have the following structure:

EtherCAT Command header	Data	WKC
10 bytes	Max. 1486 bytes	2 bytes

EtherCAT Command header

The EtherCAT command header contains the following information:

- Command to be executed
- Addressing information
- Length of the data area (Data)
- Interrupt field

Data

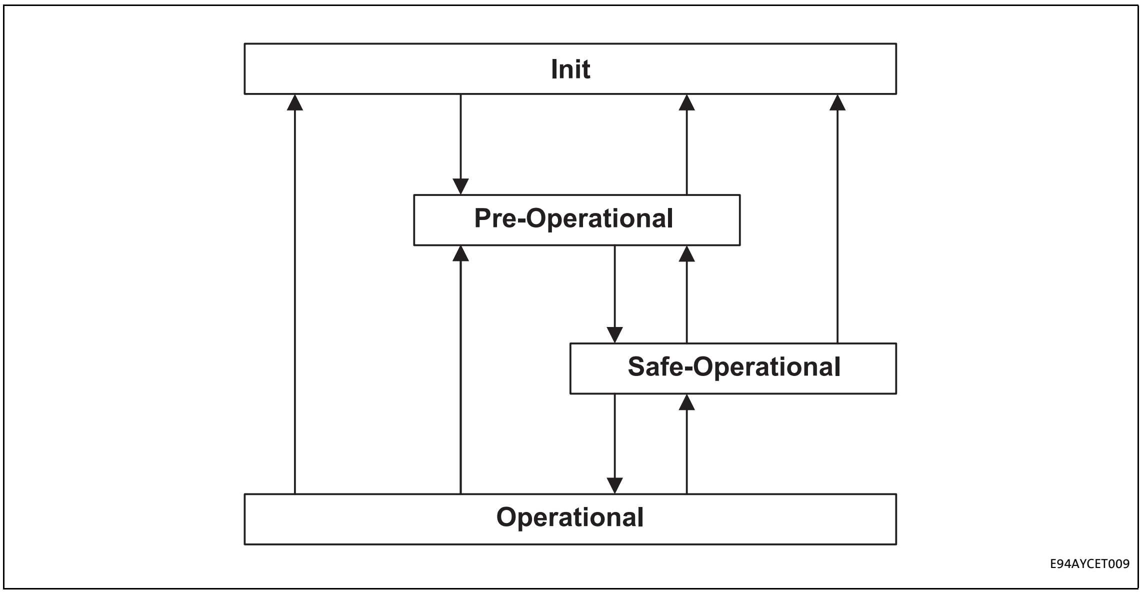
The data area contains the data of the command to be executed.

WKC (Working Counter)

The working counter is evaluated by the master for monitoring the execution of the command.

7.3 EtherCAT state machine

Before communication is possible via EtherCAT, the fieldbus passes through the EtherCAT state machine during start-up. The following illustration depicts the possible state changes from the point of view of an EtherCAT slave:



[7-1] EtherCAT state machine

State	Description
Init	• Initialisation phase • No SDO/PDO communication with the slaves • Device detection possible by means of a fieldbus scan
Pre-operational	• The fieldbus is active. • SDO communication (mailbox communication) is possible. • No PDO communication

Safe-operational	• SDO communication (mailbox communication) is possible.
	• PDO communication:
	  • The input data in the process image are updated.
	  • The output data from the process image are not transferred to the slaves.

Operational	Normal operation: • SDO communication • PDO communication • Fieldbus synchronisation successful (if used)

Note!

- A fieldbus scan can be carried out during any EtherCAT status.
- SDO communication via the EtherCAT bus is only possible if at least the "Pre-Operational" state has been reached.
- Only in the transitional phases between states can bus nodes be in different states.

The current status of the EtherCAT state machine is shown in C13861 and indicated via the **BS** LED.

Possible errors at the state transitions are shown in C13879. Additionally, an error message is entered in the "AL status code" EtherCAT register (📖 58).

▶ Diagnostics with the »Engineer« (📖 89)
▶ Fieldbus status displays (📖 86)

AL Status Code

Information about how the "AL Status Code" EtherCAT register (address 0x0134:0x0135) can be accessed can be found in the documentation of the EtherCAT master.

These error messages can be entered in the "AL Status Code" register:

AL Status Code [hex]	Description
0x0000	No fault
0x0011	Invalid status change requested
0x0012	Unknown status requested
0x0013	"Bootstrap" status is not supported
0x0016	Invalid mailbox configuration "Pre-operational"
0x001A	Synchronisation error
0x001B	Sync manager watchdog
0x001D	Invalid output data configuration
0x001E	Invalid input data configuration
0x002B	Invalid input and output data
0x0030	Invalid configuration of DC synchronisation
0x9001	Firmware watchdog error
0x9002	Mapping error

8 Process data transfer

Process data are transmitted by means of EtherCAT datagrams (📖 56) via the process data channel.

The Inverter Drive 8400 is controlled by means of the process data.

The transmission of process data is time-critical.

Process data are transferred cyclically between the Controller (EtherCAT master) and the inverters (slaves) (continuous exchange of current input and output data).

The master can directly access the process data. In the PLC for instance, the data are directly stored in the I/O area.

Up to 16 process data words (16 bits/word) per direction can be exchanged.

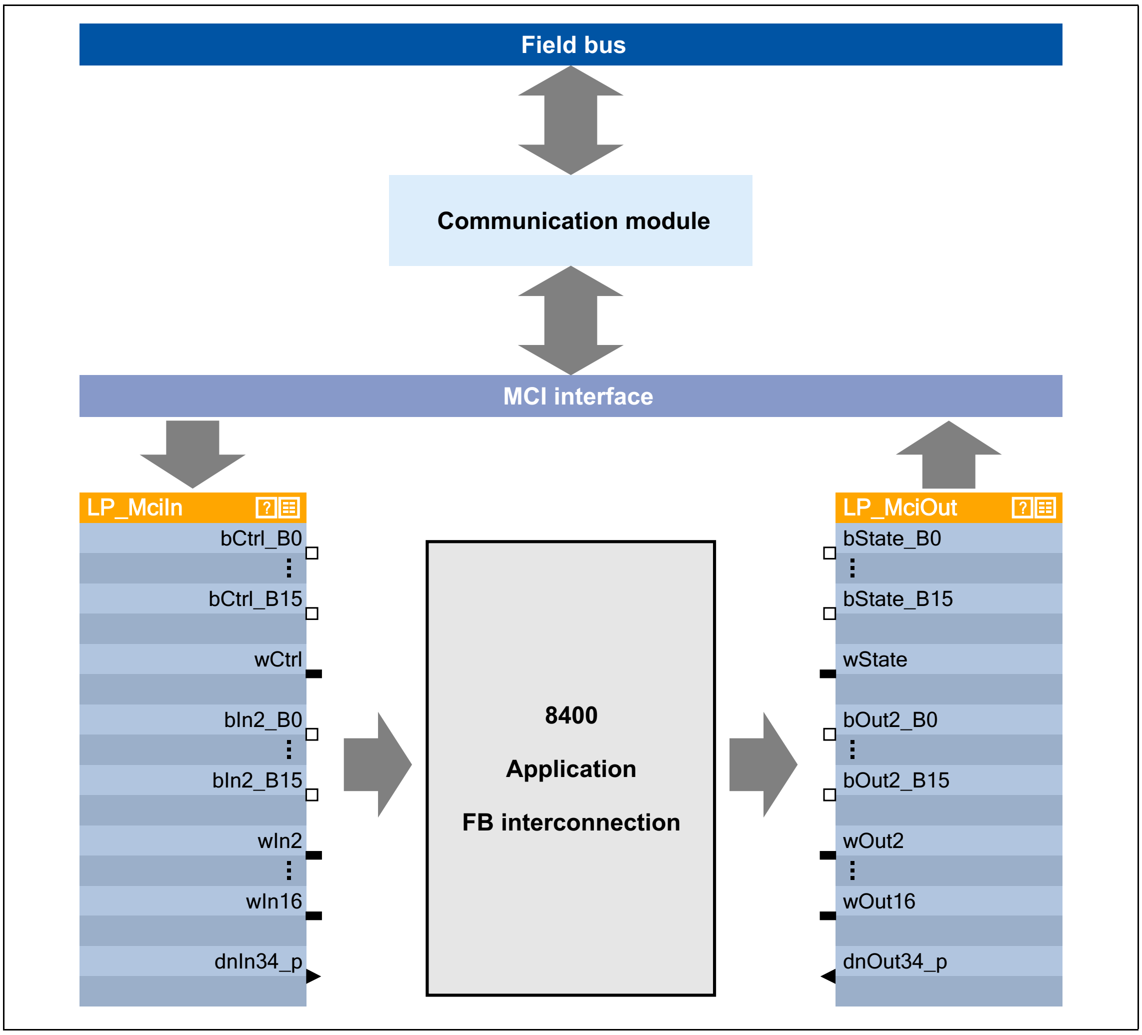
Process data are not saved in the Inverter Drive 8400.

Process data are for instance setpoints, actual values, control words, and status words.

8.1 Accessing process data / PDO mapping

Process data (MCI PDO) are transmitted via the MCI interface.

- Max. 16 words are exchanged per direction.
- The process data are accessed via the port blocks **LP_MciIn** and **LP_MciOut**. These port blocks are also referred to as process data channels.
- The port/function block interconnection of the process data objects (PDO) is made via the Lenze »Engineer«.



[8-1] Outer and inner data transfer between bus system, inverter, and application

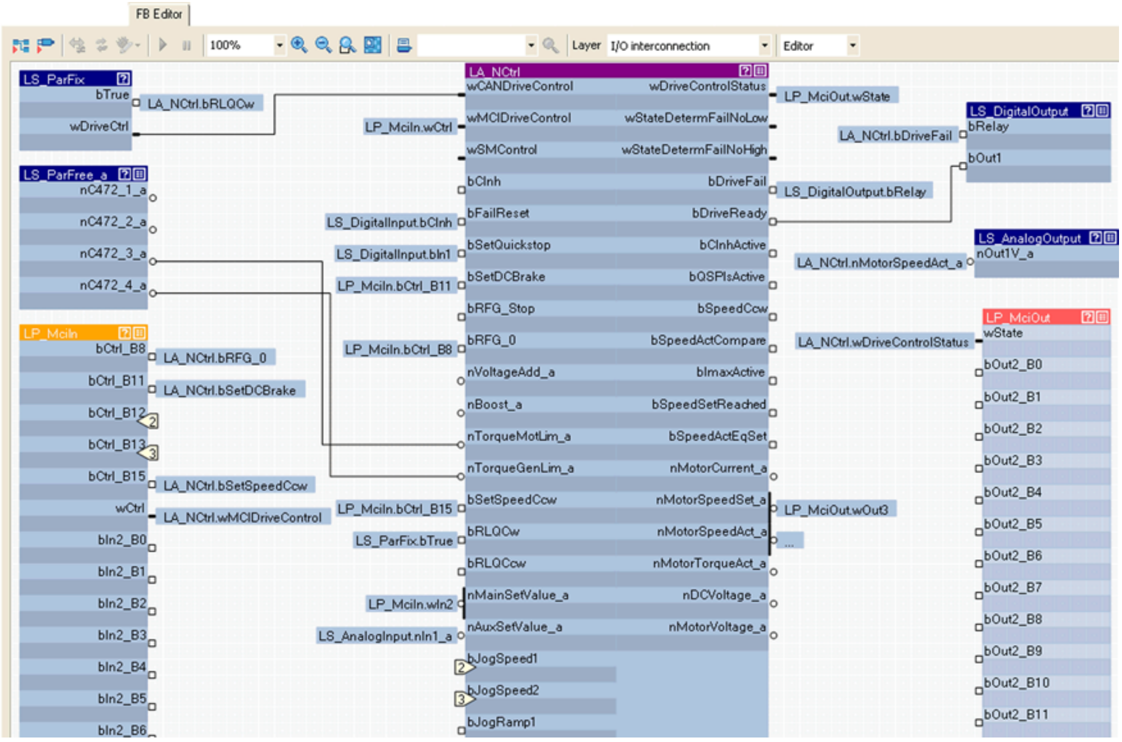
Software manual / online help for the Inverter Drive 8400

Here you can find detailed information on the port/function block interconnection in the »Engineer« and on port blocks.

8.2 Preconfigured port interconnection of the process data objects (PDO)

The preconfigured port interconnection of the process data objects can be activated by setting the standard device code **C00007 = 40: MCI**.

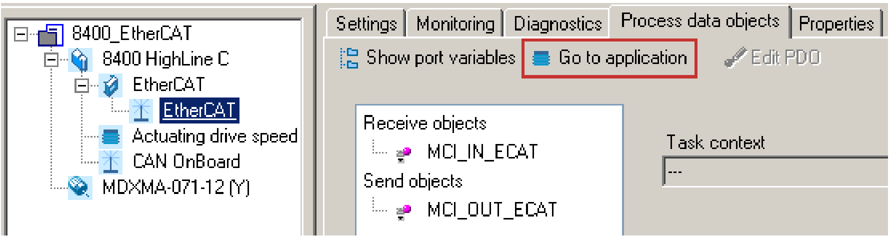
The »FB Editor« serves to display the port blocks "LP_MciIn" and "LP_MciOut" with the preconfigured interconnections:



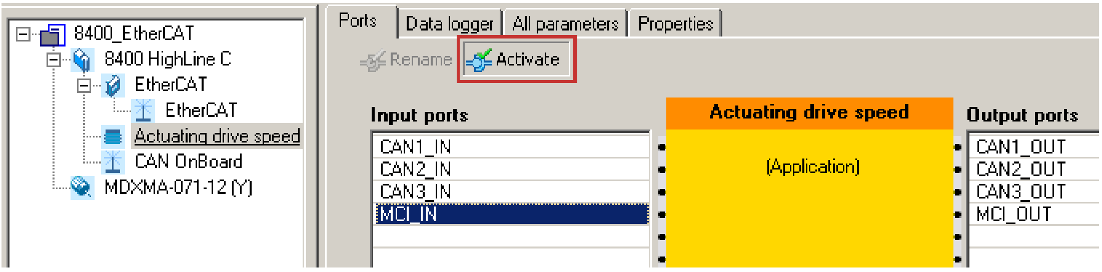
8.3 Free configuration of the port interconnection of the process data objects (PDO)

How to configure the port interconnection in the »Engineer«:

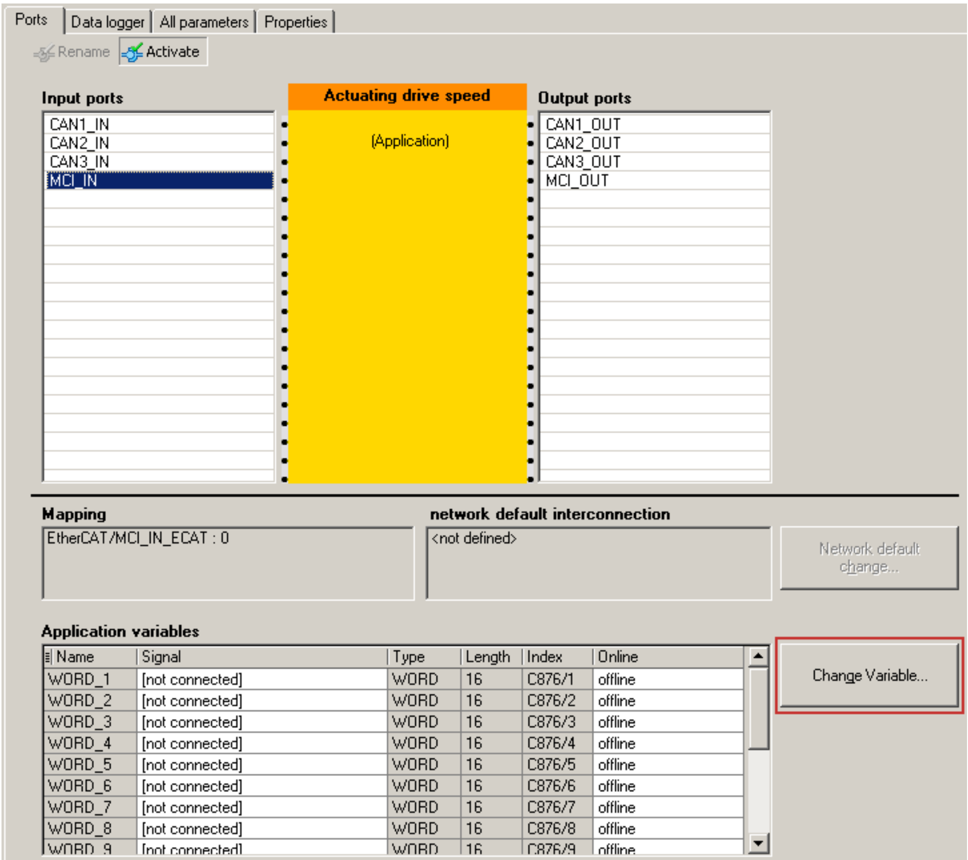
1. Under the **Process data objects** tab, click on the **Go to application** button.



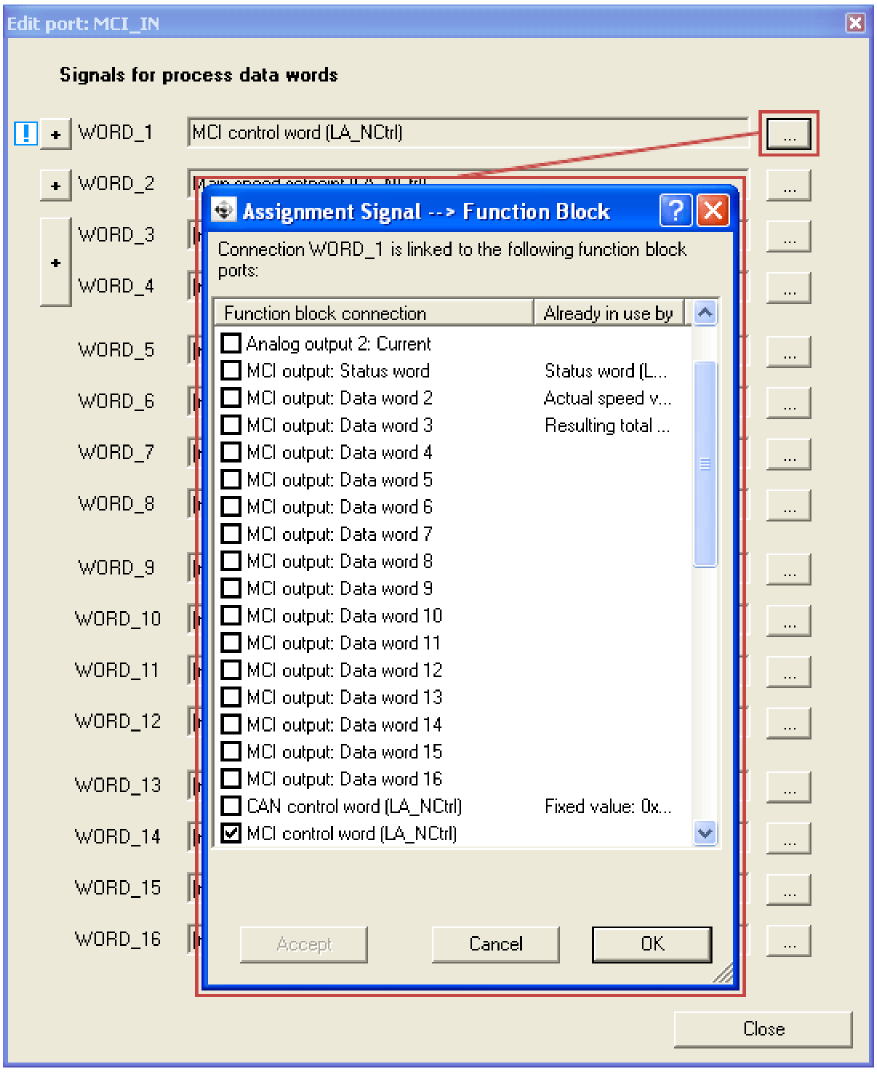
2. Under the **Ports** tab, select port block "MCI_IN" or "MCI_OUT" by mouse-click and activate it via the **Activate** button.



3. Click the **Change Variable...** button.

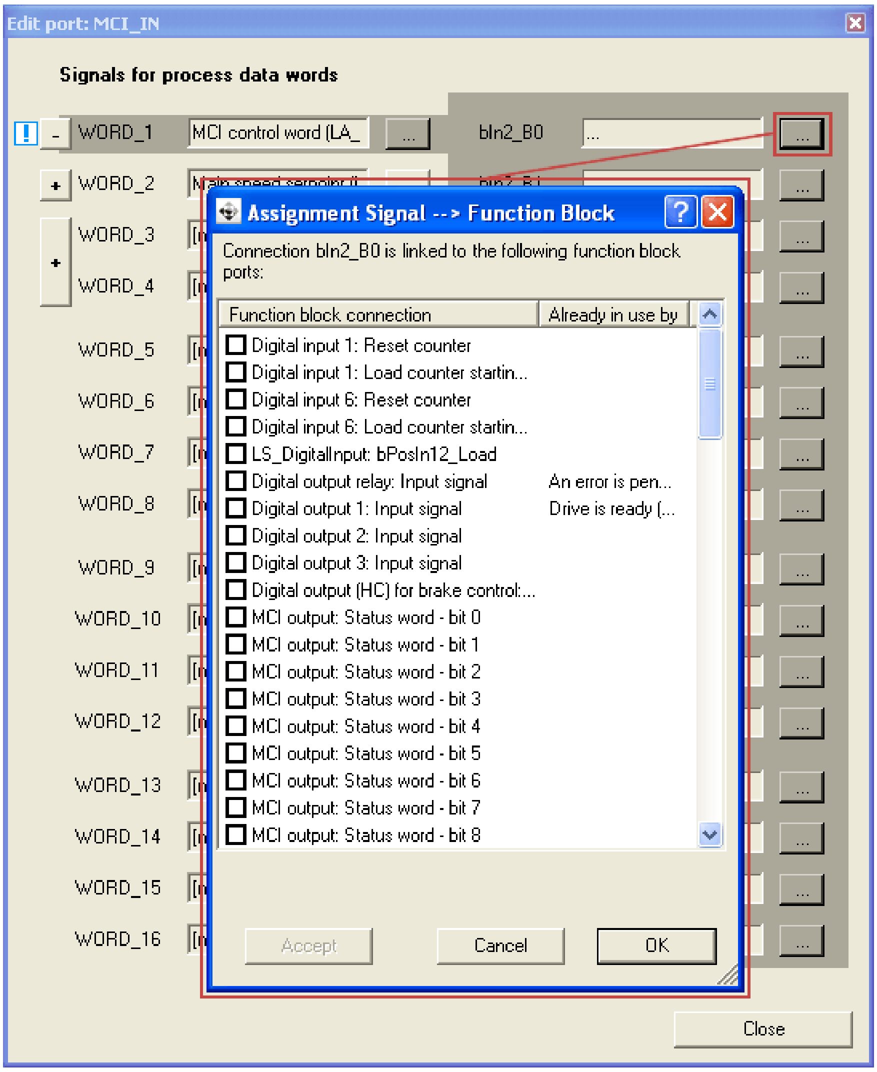


4. The [...] button serves to assign signals to the process data words in the *Assignment Signal --> Function Block* dialog box.
   ➜ Select signals and then click the **OK** button.



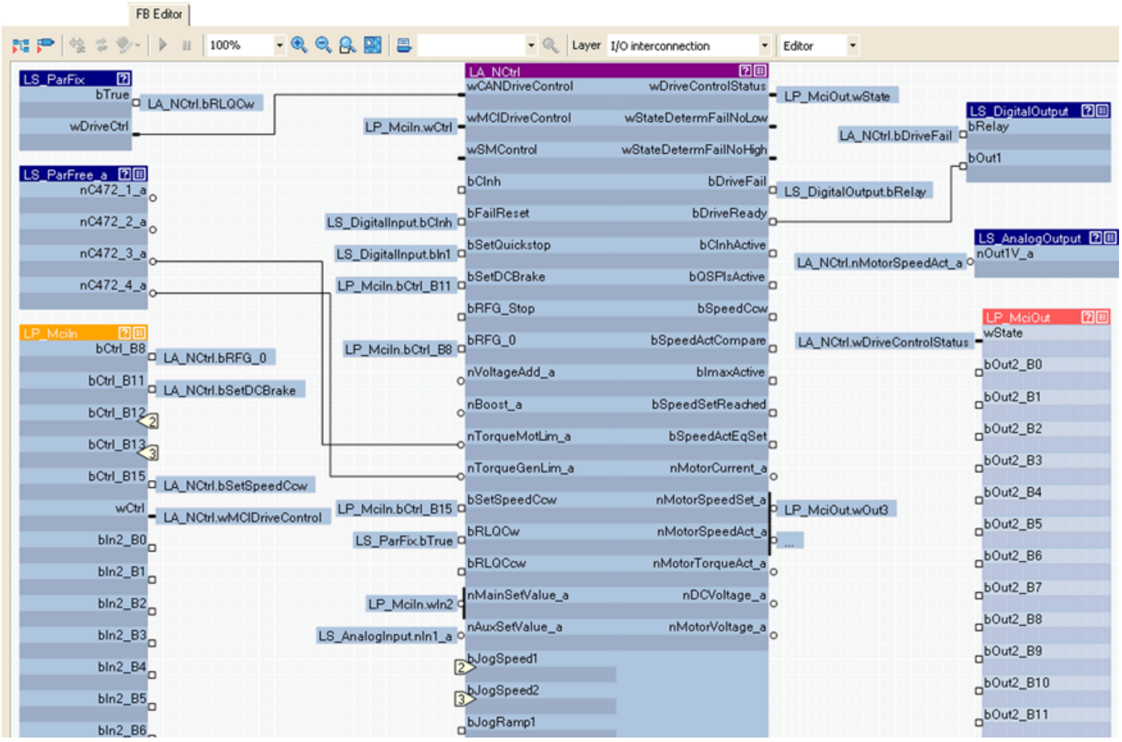
Moreover you can assign signals to the individual control and status bits at the WORD_1 and WORD_2 process data words via the + and ... buttons.

➜ Select the signals and then click **OK**.



Tip!

When the port blocks "LP_MciIn" and "LP_MciOut" are activated (see 1.), they will be visible in the »FB-Editor«. Here you can also assign signals to the process data words.



8.4 CiA402 PDO mapping in case of Inverter Drive 8400 TopLine P

The Inverter Drive 8400 TopLine P supports the CiA402 profile.

The following tables show an overview of the CiA402 objects used in the receipt PDO (Rx-PDO) and transmit PDO (Tx-PDO).

Software manual/online help for the Inverter Drive 8400 TopLine P
Here you will find detailed information on the functions of the CiA402 objects.

Receipt PDO (Rx-PDO)

Word	Index [hex]	Description
1	0x6040	Control word
2	0x6060	Operating mode
	0x60FE.1	Physical outputs
3	0x6042	Target velocity for "Velocity Mode" (is not supported yet)
4	0x607A	Target position
5		
6	0x60B1	Offset for velocity
7		
8	0x60B2	Offset for torque
9	0x60B8	Touch probe function
10	0x60E0	Positive torque limit
11	0x60E1	Negative torque limit
12	0xXXXX	Upper speed limit for "CST Mode" (is not supported yet)
13		
14	0xXXXX	Lower speed limit for "CST Mode" (is not supported yet)
15		
16	0xA580.1	Freely interconnectable

Transmit PDO (Tx-PDO)

Word	Index [hex]	Description
1	0x6041	Status word
2	0x6061	Display of the operating mode
	0x60FD	Physical inputs
3	0x6044	Current velocity for "Velocity Mode" (is not supported yet)
4	0x6064	Current position
5		
6	0x606C	Current velocity
7		
8	0x6077	Current torque
9	0x60B9	Touch probe status
10	0x60BA	Touch probe 1 positive edge
11		
12	0x60BB	Touch probe 1 negative edge
13		
14	0x603F	Error code
15	0x6074	Following error
16		

9 Parameter data transfer

Parameter data are transmitted via the fieldbus as so-called SDOs (Service Data Objects). The SDO services provide for the write and read access to the object directory.

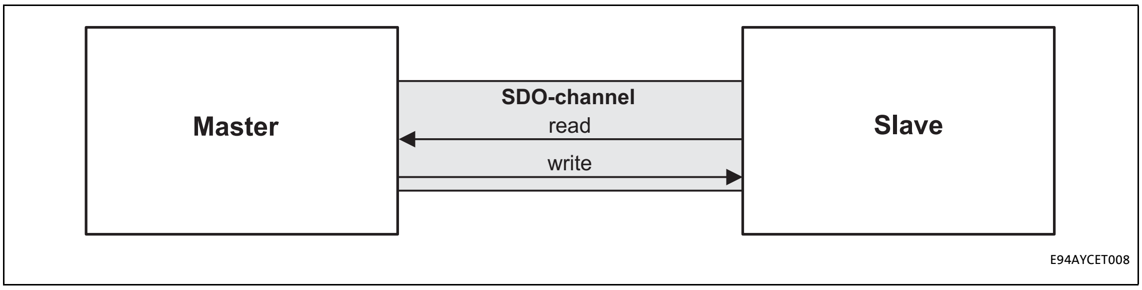
The SDO channel provides for access to Implemented CoE objects (📖 79) and Lenze codes by means of the CoE protocol.

In general, the parameter data transfer is not time-critical.

Parameter data are, for instance, operating parameters, motor data, diagnostic information.

9.1 Establishing a connection between master and slave

Basically a master can always request parameter jobs from a slave if the slave is at least in the "Pre-operational" state.



[9-1] Data communication via the SDO channel

9.2 Reading and writing parameters

Parameters ...

- for instance are set for one-time system settings or if materials are changed within a machine;
- are transmitted with a low priority.

In the case of Lenze inverters, the parameters to be changed are contained in codes.

Indexing of the Lenze codes

If they are accessed via a communication module, the codes of the inverter are addressed by the index.

The index of Lenze code numbers within the manufacturer-specific area of the object directory is between 8192 (0x2000) and 24575 (0x5FFF).

Conversion formula	
Index [dec]	**Index [hex]**
24575 - Lenze code	0x5FFF - Lenze $code_{hex}$

Example of C00002 (device commands)	
Index [dec]	**Index [hex]**
24575 - 2 = 24573	0x5FFF - 2 = 0x5FFD

Structure of a mailbox datagram

Mailbox data are transmitted in a datagramm within an EtherCAT frame. The data area of the mailbox datagram has the following structure:

Mailbox Header	CoE Header	SDO control byte	Index	Subindex	Data	Data
6 bytes	2 bytes	1 byte	2 bytes	1 byte	4 bytes	1 ... n bytes

9.2.1 Reading parameters (SDO upload)

1. The master sends "Initiate Domain Upload Request".
2. The slave acknowledges the request with a positive response ("Initiate Domain Upload Response").
   In the event of an error the slave responds with "Abort Domain Transfer".

Note!

In the case of jobs for the inverter, please make sure that you convert the code into an index.

▶ Indexing of the Lenze codes (📖 69)

SDO upload request

Detailed breakdown of the data for an "SDO upload request":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the instructing party. Station address of the target if an EtherCAT slave is the instructing party.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Reserved	BYTE	4 bits (0 ... 3)	0x00
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is read. 0x01: The complete object is read. (Is currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x02: Upload request
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Reserved	DWORD	4 bytes	0x00

SDO Upload Expedited Response

An "SDO Upload Expedited Response" is effected if the data length of the parameter data to be read is up to 4 bytes.

Detailed breakdown of the data for an "SDO Upload Expedited Response":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the instructing party. Station address of the target if an EtherCAT slave is the instructing party.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x01: Size of the data in the "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x00: 4 bytes of data 0x01: 3 bytes of data 0x02: 2 bytes of data 0x03: 1 byte of data
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is read. 0x01: The complete object is read. (Is currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x02: Upload response
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Data	DWORD	4 bytes	Data of the object

SDO Upload Normal Response

An "SDO Upload Normal Response" is effected if the data length of the parameter data to be read ≥ is 4 bytes.

Detailed breakdown of the data for an "SDO Upload Normal Response":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	n ≥ 0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the instructing party. Station address of the target if an EtherCAT slave is the instructing party.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x01
	Transfer type		1 bit (1)	0x00: Normal transfer
	Data set size		2 bits (2, 3)	0x00
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is read. 0x01: The complete object is read. (Is currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x02: Upload response
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Complete size	DWORD	4 bytes	Total data length of the object
	Data	BYTE	n - 10 bytes	Data of the object

Example

In case of an **Upload** to index 0x5FD8 (Standard setting of C00039/1 (fixed setpoint_1) = $0x0FA0_{hex}$ (4000_{dec})), the transmitted response structure contains the following data:

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	0x00
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x01: Length of the data in the "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x02: 2 bytes of data
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is read.
	Command specifier		3 bits (5 ... 7)	0x02: Upload response
	Index	WORD	2 bytes	0xD8: Index low byte of the object 0x5F: Index high byte of the object
	Subindex	BYTE	1 byte	0x01
	Data	DWORD	2 bytes	0x0FA0

9.2.2 Writing parameters (SDO download)

1. The master sends "Initiate Domain Download Request".
2. The slave acknowledges the request with a positive response ("Initiate Domain Download Response").
   In the event of an error the slave responds with "Abort Domain Transfer".

Note!

In the case of jobs for the inverter, please make sure that you convert the code into an index.

▸ Indexing of the Lenze codes (📖 69)

SDO Download Expedited Request

An "SDO Download Expedited Request" is effected if the data length of the parameter data to be written is up to 4 bytes.

Detailed breakdown of the data for an "SDO Download Expedited Request":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the instructing party. Station address of the target if an EtherCAT slave is the instructing party.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Size indicator	BYTE	1 bit (0)	0x01: Size of the data in the "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x00: 4 bytes of data 0x01: 3 bytes of data 0x02: 2 bytes of data 0x03: 1 byte of data
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is written. 0x01: The complete object is written. (Is currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x01: Download request
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Data	DWORD	4 bytes	Data of the object

SDO Download Normal Request

An "SDO Download Normal Request" is effected if the data length of the parameter data to be written ≥ is 4 bytes.

Detailed breakdown of the data for an "SDO Download Normal Request":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	n ≥ 0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the instructing party. Station address of the target if an EtherCAT slave is the instructing party.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Size indicator	BYTE	1 bit (0)	0x01
	Transfer type		1 bit (1)	0x00: Normal transfer
	Data set size		2 bits (2, 3)	0x00
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is written. 0x01: The complete object is written. (Is currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x01: Download request
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Complete size	DWORD	4 bytes	Total data length of the object
	Data	BYTE	n - 10 bytes	Data of the object

SDO Download Response

Detailed breakdown of the data for a "SDO download response":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the instructing party. Station address of the target if an EtherCAT slave is the instructing party.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x0
	Transfer type		1 bit (1)	0x0
	Data set size		2 bits (2, 3)	0x0
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is written. 0x01: The complete object is written. (Is currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x3: Download response
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Reserved	DWORD	4 bytes	0x00

Example

In the case of a **Download** to the index 0x1600, the transmitted request structure contains the following data:

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	0x00
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 Bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Size indicator	BYTE	1 bit (0)	0x01: Size of the data in the "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x00: 4 bytes of data
	Complete access		1 bit (4)	0x00: The entry addressed with an index and subindex is written.
	Command specifier		3 bits (5 ... 7)	0x01: Download request
	Index	WORD	2 bytes	0x00: Index low byte of the object 0x16: Index high byte of the object
	Subindex	BYTE	1 byte	0x01: Subindex of the object
	Data	DWORD	4 bytes	0x5C930110

9.3 Implemented CoE objects

Lenze devices can be parameterised with both Lenze codes and the manufacturer-independent "CoE objects". In order to fully comply with EtherCAT communication, you must only use the CoE objects for parameterisation. The CoE objects described in this manual are defined in the "EtherCAT Specification, Part 6 – Application Layer Protocol Specification".

Index	Designation	Subindex	Subindex name	Type	Bits	Access
0x1000	Device type	-	-	UDINT	32	R
0x1008	Device name	-	-	STRING(8)	64	R
0x1009	Hardware version	-	-	STRING(8)	64	R
0x100A	Software version	-	-	STRING(7)	56	R
0x1018	Identity	0	Subindex 000	USINT	8	R
		1	Vendor ID	UDINT	32	R
		2	Product code	UDINT	32	R
		3	Revision number	UDINT	32	R
		4	Serial number	UDINT	32	R
0x10F1	Error Settings Object	0	Subindex 000	USINT	8	R
		1	Local Error Reaction	USINT	8	R
		2	Sync Error Counter Limit	USINT	8	RW
0x1600	RxPDO 1	0	Subindex 000	USINT	8	RW
		1 ... 16	Output Object 1 ... 16	UDINT	32	RW
0x1A00	TxPDO 1	0	Subindex 000	USINT	8	RW
		1 ... 16	Input Object 1 ... 16	UDINT	32	RW
0x1C00	Sync Man Communication type	0	Subindex 000	USINT	8	R
		1	Elements	UDINT	32	R
0x1C12	Sync Man 2 Assignment	0	Subindex 000	USINT	8	R
		1	PDO Mapping object index of assigned RxPDO	UDINT	32	R
0x1C13	Sync Man 3 Assignment	0	Subindex 000	USINT	8	R
		1	PDO Mapping object index of assigned TxPDO	UDINT	32	R
0x1C32	Sync Man 2 Synchronization	0	Subindex 000	USINT	8	R
		1	Synchronization type	UINT	16	R
		2	Cycle time / ns	UDINT	32	R
		3	Shift time / ns	UDINT	32	R
		4	Sync types supported	UINT	16	R
		5	Minimum cycle time / ns	UDINT	32	R
		6	Minimum shift time / ns	UDINT	32	R
0x1C33	Sync Man 3 Synchronization	0	Subindex 000	USINT	8	R
		1	Synchronization type	UINT	16	R
		2	Cycle time / ns	UDINT	32	R
		3	Shift time / ns	UDINT	32	R
		4	Sync types supported	UINT	16	R
		5	Minimum cycle time / ns	UDINT	32	R
		6	Minimum shift time / ns	UDINT	32	R

R: Read access only
RW: Read and write access

9.4 EtherCAT objects of the communication module

The object directory displays the Parameters of the communication module (📖 97) as objects:

Index	Code	Index name	Subindex	Subindex name	Type	Bits	Access
0x29E5	C13850	All words to master	1 ... 16	All words to master	UNSIGNED	16	R
0x29E4	C13851	All words from master	1 ... 16	All words from master	UNSIGNED	16	R
0x29E3	C13852	All words to standard device	1 ... 16	All words to standard device	UNSIGNED	16	R
0x29E2	C13853	All words from standard device	1 ... 16	All words from standard device	UNSIGNED	16	R
0x29DC	C13859	Number of Tx PDOs	-	-	UNSIGNED	16	R
0x29DB	C13860	Number of Rx PDOs	-	-	UNSIGNED	16	R
0x29DA	C13861	Bus state	-	-	UNSIGNED	16	R
0x29D7	C13864	Active station address	-	-	UNSIGNED	16	R
0x29D4	C13867	Display of emergency data	-	-	STRING(8)	64	R
0x29C8	C13879	Bus error	-	-	UNSIGNED	16	R
0x29C7	C13880	Reaction on communication failure	2	1: Response to communication interruption 2: Response to sync telegram failure	UNSIGNED	8	RW
0x29C6	C13881	Response time when exiting "Operational"	-	-	UNSIGNED	16	RW
0x29C4	C13883	DC active	-	-	UNSIGNED	8	R
0x29C3	C13884	Synchronisation is active	-	-	UNSIGNED	8	R
0x29C2	C13885	Clear process data	-	-	UNSIGNED	8	RW
0x29C0	C13887	Suppress emergency message in case of	-	-	BITFIELD	8	RW
0x29B4	C13899	Station alias address	-	-	UNSIGNED	16	RW
0x29B3	C13900	Firmware product type	-	-	STRING(8)	64	R
0x29B2	C13901	Firmware compilation date	-	-	STRING(20)	160	R
0x29B1	C13902	Firmware version	-	-	STRING(11)	88	R

R: Read access only
RW: Read and write access

9.5 SDO abort codes (Abort codes)

If an SDO request is evaluated negatively, a corresponding error code is output.

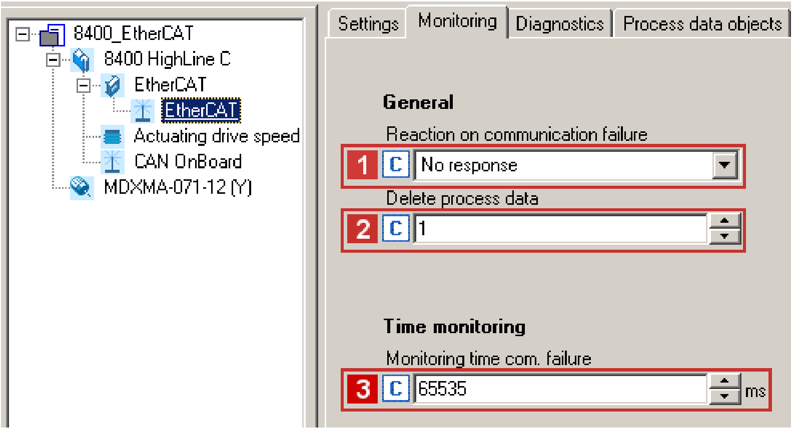
Index [hex]	Description
0x00000000	No fault
0x05030000	The status of the toggle bit has not changed
0x05040000	SDO protocol time-out
0x05040001	Invalid or unknown specification symbol for the client/server command
0x05040005	Not enough space in the main memory
0x06010000	Access to object not supported
0x06010001	Read access to a write-protected object
0x06010002	Write access to a write-protected object
0x06020000	An object does not exist in the object directory
0x06040041	An object cannot be mapped into the PDO
0x06040042	The number and/or length of the objects mapped would exceed the PDO length
0x06040043	General parameter incompatibility
0x06040047	General internal device incompatibility
0x06060000	Access has failed due to a fault in the hardware
0x06070010	The data type or the parameter length does not correspond
0x06070012	Incorrect data type (The parameter length is too large)
0x06070013	Incorrect data type (The parameter length is too small)
0x06090011	A subindex is not available
0x06090030	The value range for parameters is too great (only for write access)
0x06090031	The parameter value is too high
0x06090032	The parameter value is too low
0x06090036	The maximum value is lower than the minimum value
0x08000000	General error
0x08000020	Data cannot be transferred to the application or stored in the application
0x08000021	Due to local control, data cannot be transferred to the application or stored in the application
0x08000022	Due to the current device state, data cannot be transferred to the application or stored in the application
0x08000023	The dynamic object directory generation has failed, or no object directory is available

10 Monitoring

10.1 Interruption of EtherCAT communication

An interruption of the EtherCAT communication in the "Operational" state, e.g. due to cable break of failure of the EtherCAT master, is detected by the slave.

The response to an interrupted communication is triggered by settings in the **Monitoring** tab:



- During the initialisation of the EtherCAT communication, the sync manager watchdog monitoring time determined in the master is transferred to the slave.
  If the slave does not receive any valid process data in the "Operational" status, the setting in 2 C13885 is taken as a basis for the process data:
  - Value '0': The data sent last by the master are used.
  - Value '1': PDOs are set to the value '0'.

  After the watchdog monitoring time has elapsed, the slave changes to the "Safe operational" state (see C13861) and the green LED **BS** is activated (see Fieldbus status displays (📖 86)).
  There is no response in the slave.
- In order that a response is triggered in the slave, you have to set a 1 **Reaction on communication failure** (C13880/1).
- The response is delayed if you set an internal 3 **monitoring time** (C13881) in addition.
  In the Lenze setting (C13881 = 0), no delay is set.
  The monitoring time elapses as soon as the "Operational" state is exited. (See 1.).
  After the monitoring time has elapsed, the response set is executed with the error message "Operational status quit [0x01bc8131]" (📖 95).

10.2 Sync frame failure detection

During the Synchronisation with "Distributed Clocks" (DC) (📖 37), this monitoring checks whether an EtherCAT PDO telegram (Sync Manager 2 Event) has arrived between two signals.

For this purpose, the communication module comes with an internal EtherCAT telegram failure error counter. The telegram failure error counter is incremented by the value '3' in case of a telegram failure. For each PDO received correctly, the counter is decremented by ’1’. If the internal telegram failure error counter reaches a threshold adjustable in CoE object 0x10F1.2, a change to "Safe-Operational" takes place and the "Sync telegram failure [0x01bc8700]" error message is output.

In the standard setting, monitoring is deactivated (0x10F1.2 = 0). A sensible setting is a value > ’5’ in order that an error will only be triggered in the inverter when a second telegram fails.

An error response can be set via code C13880/2 ("0: No response" is preset here).

10.3 Interruption of internal communication

The response to a communication error between the communication module and the standard device can be set in the standard device code C01501 (module in the MCI slot).

An externally supplied communication module reports a connection abort to the standard device via an emergency telegram to the master and changes to the "Safe-operational" state.

The error message "Lost connection to 8400 [0x01bc3100]" (📖 93) is output.

11 Diagnostics

For fault diagnostics, the communication module is provided with LEDs on the front. Furthermore, you can carry out the Diagnostics with the »Engineer« (📖 89).

11.1 LED status displays

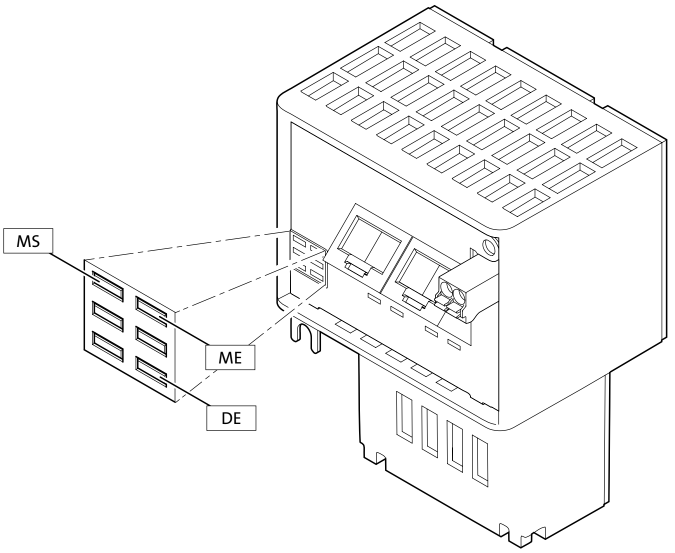
Note!

During normal operation ...
- only the LEDs **MS** (📖 85) and **BS** (📖 86) should be lit constantly.
- the green LED at the RJ45 sockets X246/X247 must be lit (📖 88).

The following status displays can be differentiated:
- Module status displays (📖 85)
- Fieldbus status displays (📖 86)
- Status displays at X246 and X247 (📖 88)

11.1.1 Module status displays

The LEDs **MS**, **ME** and **DE** show the module status.

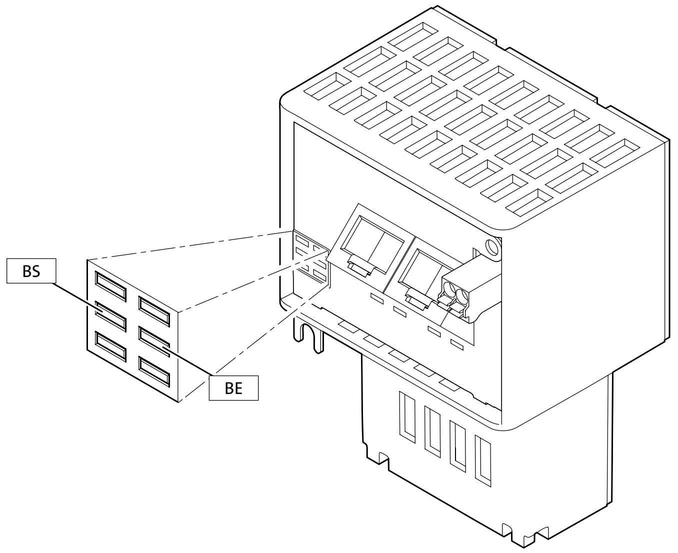


[11-1] LEDs MS, ME, DE

LED	Colour	State	Description
MS	green	on	The communication module is supplied with voltage and is connected to the standard device.
		blinking	 The communication module is supplied with voltage, but has no connection to the standard device (the standard device is switched off, in the initialisation phase, or not available).
ME	red	on	An error has occurred in the communication module.
DE	red	on	The communication module is not accepted by the standard device, or the standard device is not active. (See notes in the documentation for the standard device).

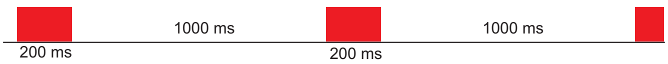
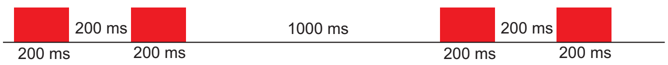
11.1.2 Fieldbus status displays

The LEDs **BS** and **BE** show the fieldbus status.



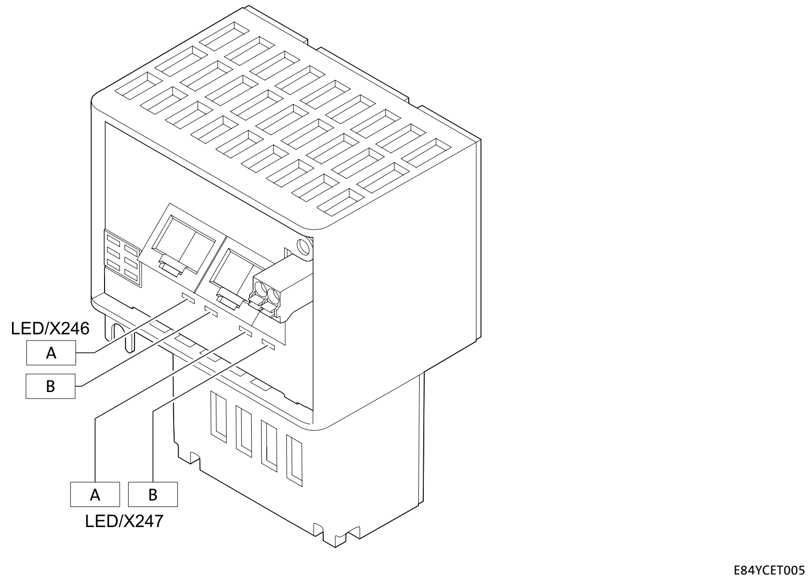
[11-2] LEDs BS, BE

LED	Colour	State	Description
BS	green	off	The communication module is not active on the fieldbus or is in the "Init" state.
		blinking	200 ms 200 ms "Pre-operational" status is active: • Access to parameters and objects is possible. • No process data exchange.
		blinking once (single flash)	1000 ms 1000 ms 200 ms 200 ms "Safe-operational" status is active: • The data are not yet active in the standard device.
		on	The communication module is in the "Operational" state.

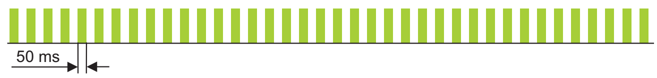
<table>
<tr><th>LED</th><th>Colour</th><th>State</th><th>Description</th></tr>
<tr><td rowspan="4">BE</td><td rowspan="4">red</td><td>off</td><td>No fault</td></tr>
<tr><td>blinking</td><td>
The configuration is invalid/faulty.</td></tr>
<tr><td>blinking once
(single flash)</td><td>
• A non requested state change has occurred. (The slave application has autonomously changed the EtherCAT status.)
• Synchronisation error (The EtherCAT node automatically changes to the "Safe-operational" state.)</td></tr>
<tr><td>blinking twice
(double flash)</td><td>
An "Application Watchdog Timeout" or a "Sync Manager Watchdog Timeout" has occurred.</td></tr>
</table>

11.1.3 Status displays at X246 and X247

The LEDs beneath the RJ45 sockets X246 and X247 show the EtherCAT connection status.

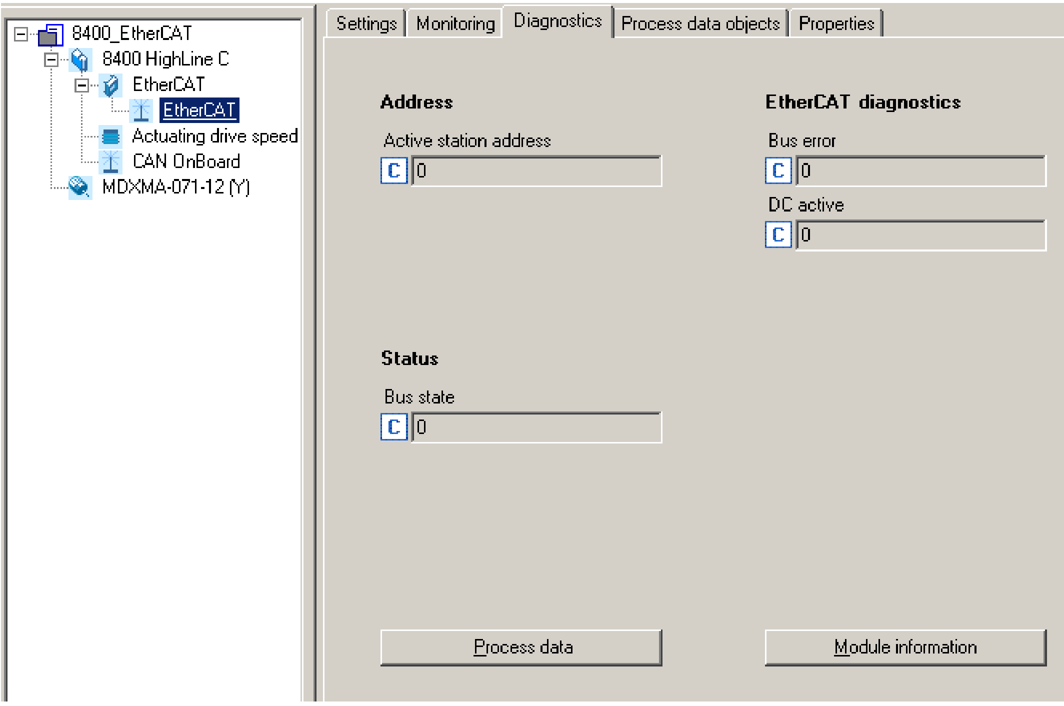


[11-3] LEDs at the RJ45 sockets X246 and X247

LED	Colour	State	Description
A	green	on	 A physical EtherCAT connection is available.
		flickering	 Data are being exchanged via EtherCAT.
B	red	off	This LED is not used.

11.2 Diagnostics with the »Engineer«

In the »Engineer«, the **Diagnostics** tab displays various pieces of EtherCAT diagnostic information.



11.3 Emergency requests / Emergency messages

Emergency messages are sent to the EtherCAT master once when the error status of the inverter changes, i.e. ...

- when an error in the inverter or in the communication module occurs;
- when an internal error of the communication module is eliminated.

An "Emergency Request" on the fieldbus consists of the components "Mailbox Header", "CANopen Header" and the actual "Emergency Message":

Mailbox header	CANopen header	Emergency Message
6 bytes	2 bytes	8 bytes

The emergency message last sent by the Inverter Drive 8400 is displayed in code C13867.

Code C13887 serves to select the error responses where no emergency messages are to be sent to the EtherCAT master.

11.3.1 Structure of the Emergency message

Example: Emergency message of the error "Operational status quit [0x01bc8131]"

Byte 1	Byte2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Emergency Error code		**Error Register (I-1001)**	**Reserved**	**Error code Inverter Drive 8400 / E84AYCET**			
Low byte	High byte	Low byte	High byte	Low word		High word	
				Low byte	High byte	Low byte	High byte
0x00	**0x10**	**0x01**	**0x00**	**0x31**	**0x81**	**0xbc**	**0x01**

- Bytes 1 and 2 display that an error is pending.
- Byte 3 display the contents of the error register (I-1001).
- Bytes 5 ... 8 contain the error code.

Software manual/online help for the Inverter Drive 8400
Here you will find detailed information on the error codes.

11.3.2 Emergency messages (overview)

The following emergency messages can occur:

Emergency no. [hex]	Designation	Extended error code [hex]	Description
0x1000	General error	0x01bc3100	Lost connection to 8400 [0x01bc3100] (🕮 93)
0x1000	General error	0x01bc6430	Invalid module configuration [0x01bc6430] (🕮 94)
0x1000	General error	0x01bc8131	Operational status quit [0x01bc8131] (🕮 95)
0xA000	Synchronisation error	0x01bc8265	Sync telegram failure [0x01bc8700] (🕮 95)
0x0000	Error reset	-	-

12 Error messages

This chapter provides the error messages of the communication module E84AYCET EtherCAT as a supplement to the error list in the software manual and the »Engineer« online help for the Inverter Drives 8400.

12.1 Short overview of the EtherCAT error messages

Software manual/online help for the Inverter Drive 8400

Here you can find general information on diagnostics & fault analysis and on error messages.

The following table lists all EtherCAT error messages in numerical order of the error number. Furthermore the preset error response and – if available – the parameters for setting the error response are specified.

Tip!

When you click the cross-reference in the first column, you will see a detailed description (causes and remedies) of this error message.

Error no. [hex]	Subject area no. [dec]	Error no. [dec]	Error text	Error type (Error response)	Adjustable in
0x01bc3100	444	12544	Lost connection to 8400 base device	1: Fault	-
0x01bc5531	444	21809	Memory: No access	1: Fault	-
0x01bc5532	444	21810	Memory: Read error	1: Fault	-
0x01bc5533	444	21811	Memory: Write error	1: Fault	-
0x01bc6010	444	24592	Restart by watchdog reset	1: Fault	-
0x01bc6011	444	24593	Internal error	1: Fault	-
0x01bc6100	444	24832	Internal error	1: Fault	-
0x01bc6101	444	24833	Internal error	1: Fault	-
0x01bc641f	444	25631	Invalid parameter record	1: Fault	-
0x01bc6420	444	25632	Error: Lenze setting loaded	1: Fault	-
0x01bc6430	444	25648	Invalid module configuration	1: Fault	-
0x01bc8131	444	33073	"Operational" status quit	0: No response	C13880/1
0x01bc8700	444	34560	Sync telegram failure	0: No response	C13880/2

12.2 Possible causes and remedies

This chapter lists all EtherCAT error messages in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

Lost connection to 8400 [0x01bc3100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
MCI communication for the Inverter Drives 8400 is interrupted. • Inverter Drive 8400 is switched off. • The communication module has not been inserted properly into the slot of the Inverter Drives 8400.	• Switch on Inverter Drive 8400. • Check screwed connection of the communication module to the MC slot of the Inverter Drive 8400. • Send communication module and Inverter Drive 8400 with error description to Lenze.

Memory: No access [0x01bc5531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access via standard device to parameter set within the memory module was not successful.	Repeat download of the application (including module).

Memory: Read error [0x01bc5532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameters of the memory module could not be read.	Repeat download of the application (including module).

Memory: Write error [0x01bc5533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameters of the memory module could not be written.	Repeat download of the application (including module).

Restart by watchdog reset [0x01bc6010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Internal error [0x01bc6011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Internal error [0x01bc6100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Internal error [0x01bc6101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Invalid parameter set [0x01bc641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Repeat download of the application (including module).

Error: Lenze setting loaded [0x01bc6420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access via standard device to parameter set within the memory module was not successful.	Repeat download of the application (including module).

Invalid module configuration [0x01bc6430]

Response (Lenze setting printed in bold)	**Setting:** not possible
☒ **None** ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Faulty module configuration. • Incorrect sync settings.	Check and correct module configuration/sync settings.

Operational status quit [0x01bc8131]

Response (Lenze setting printed in bold)	Setting: C13880/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
The EtherCAT data exchange was stopped in the "Operational" state. See also chapter "Interruption of EtherCAT communication" (📖 82).	• Check cables and terminals. • The master has to reset the node to the "Operational" status. (If required, check a pending emergency message first).

Sync telegram failure [0x01bc8700]

Response (Lenze setting printed in bold)	Setting: C13880/2 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
• The cycle time settings are faulty. • The task time setting of a linked PLC program is faulty. • EtherCAT master / DC master settings are incorrect. • The sync source setting is faulty.	• Check the cycle times. • Check the task time of a linked PLC program. • Check the EtherCAT master / DC master settings. • Check the settings in C01120 of the standard device (selection of the sync source).

13 Parameter reference

This chapter supplements the parameter list and the table of attributes in the software manual and in the »Engineer« online help for the Inverter Drive 8400 by the parameters of the E84AYCET communication module (EtherCAT).

Software manual/online help for the Inverter Drive 8400

Here you will find general information about parameters.

13.1 Communication-relevant parameters of the standard device

This chapter lists the communication-relevant parameters of the Inverter Drive 8400 in numerically ascending order.

C01120

Parameter | Name: **C01120 | Sync signal source**			Data type: UNSIGNED_8 Index: 23455_d = $5B9F_h$
Selection of the source for DC synchronisation signals. • Select "4: MCI" for the communication module. • In general, only one source can synchronise the drive. ▸ Synchronisation with "Distributed Clocks" (DC) (📖 37)			
Selection list (Lenze setting printed in bold)		**Info**	
0	**Off**	Synchronisation off	
1	CAN on board	Synchronisation via the system bus "CAN on board"	
4	MCI	Synchronisation via MCI (communication module)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

C01501

Parameter | Name: **C01501 | Reaction on communication error with MCI**			Data type: UNSIGNED_8 Index: 23074_d = $5A22_h$
Configuration of monitoring functions for the communication module			
Selection list			
0	No response		
1	Error		
2	Trouble		
3	Quick stop by trouble		
4	Warning locked		
5	Warning		
6	Information		
Subcodes	**Lenze setting**	**Info**	
C01501/1	0: No response	Reaction on MCI communication failure • Response to a communication error (module in slot)	
C01501/2	0: No response	Reaction on MCI module invalid • Response to an incompatible communication module or module not in slot	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ COM ☐ MOT Scaling factor: 1			

13.2 Parameters of the communication module

This chapter lists the parameters of the E84AYCET communication module (EtherCAT) in ascending numerical order.

C13850

Parameter | Name: **C13850 | All words to master**			Data type: UNSIGNED_16 Index: 10725_d = $29E5_h$
Display of the process data words (subcodes 1 ... 16) which are transferred from the communication module to the master. Only those which are configured are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master**			Data type: UNSIGNED_16 Index: 10724_d = $29E4_h$
Display of the process data words (subcodes 1 ... 16) which are transferred from the master to the communication module. Only those which are configured are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to standard device**			Data type: UNSIGNED_16 Index: 10723_d = $29E3_h$
Display of the process data words (subcodes 1 ... 16) which are transferred from the communication module to the standard device.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words from standard device**			Data type: UNSIGNED_16 Index: 10722_d = $29E2_h$
Display of the process data words (subcodes 1 ... 16) which are transferred from the standard device to the communication module.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13853/1			
...			
C13853/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13859

Parameter | Name: **C13859 | Number of Tx PDOs**			Data type: UNSIGNED_16 Index: 10716_d = $29DC_h$
Number of process data words sent			
Display area (min. value | unit | max. value)			
0		16	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13860

Parameter | Name: **C13860 | Number of Rx PDOs**			Data type: UNSIGNED_16 Index: 10715_d = $29DB_h$
Number of process data words received			
Display area (min. value | unit | max. value)			
0		16	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Bus state**			Data type: UNSIGNED_16 Index: 10714_d = $29DA_h$
Display of the current bus status ▸ EtherCAT state machine (📖 57)			
Selection list (read only)		**Info**	
0	Nonexistent	No online value available (communication module is offline).	
1	Init		
2	Pre-operational		
3	Bootstrap	Is not supported.	
4	Safe-operational		
8	Operational		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13864

Parameter | Name: C13864 | Active station address			Data type: UNSIGNED_16 Index: 10711_d = $29D7_h$
Display of the station address allocated by the master			
Display area (min. value | unit | max. value)			
0		32767	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13867

Parameter | Name: C13867 | Display of emergency data	Data type: OCTET_STRING Index: 10708_d = $29D4_h$
Display of the emergency data last sent by the inverter (string with a length of 8 bytes). ▸ Emergency requests / Emergency messages (📖 90)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13879

Parameter | Name: C13879 | Bus error			Data type: UNSIGNED_16 Index: 10696_d = $29C8_h$
Bit coded display of the bus error Additionally, an error message is entered in the EtherCAT register "AL Status Code" (📖 58).			
Value is bit-coded:		**Info**	
Bit 0	General bus error		
Bit 1	Reserved		
...	...		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13880

Parameter | Name: C13880 | Reaction on communication failure			Data type: UNSIGNED_8 Index: 10695_d = $29C7_h$
The set response will be executed if ... • the node detects that is no longer in the "Operational" state and the monitoring time (C13881) has elapsed or • a sync telegram failure has been detected.			
Selection list			
0	No response		
1	Error		
3	Quick stop by trouble		
4	Warning locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	
C13880/1	0: No response	Response to communication interruption ▸ Interruption of EtherCAT communication (📖 82)	
C13880/2	0: No response	Response to sync telegram failure ▸ Sync frame failure detection (📖 83)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13881

Parameter | Name: **C13881 | Response time when exiting "Operational"**			Data type: UNSIGNED_16 Index: 10694_d = $29C6_h$
If the "Operational" state is exited, the response parameterised with C13880 occurs after the time set here has elapsed. A change in monitoring is effective immediately.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13883

Parameter | Name: **C13883 | DC active**		Data type: UNSIGNED_8 Index: 10692_d = $29C4_h$
The code displays if the DC synchronisation has been activated for the communication module. ▸ Synchronisation with "Distributed Clocks" (DC) (📖 37)		
Selection list (read only)		**Info**
0	DC unused	
1	DC for synchronisation	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13884

Parameter | Name: **C13884 | Base device synchronisation status**		Data type: UNSIGNED_8 Index: 10691_d = $29C3_h$
The code displays whether the inverter is DC-synchronised. ▸ Synchronisation with "Distributed Clocks" (DC) (📖 37)		
Selection list (read only)		**Info**
0	Not synchronised	
1	Synchronised	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13885

Parameter | Name: **C13885 | Clear process data**		Data type: UNSIGNED_8 Index: 10690_d = $29C2_h$
This code serves to set which process data the slave is to process for maintaining internal communication when the EtherCAT has exited the "Operational" state.		
Selection list (Lenze setting printed in bold)		**Info**
0	Use of most recent master PDOs	The data last sent by the master are used.
1	**PDOs are set to the value '0'**	The process data contents is set to the value '0'.
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13887

Parameter | Name: **C13887 | Suppress emergency message in case of**		Data type: BITFIELD_8 Index: 10688_d = $29C0_h$
This code serves to prevent the emergency messages from being transmitted to the EtherCAT master. Here, errors of a certain type can be suppressed deliberately. Furthermore, all errors are entered in the logbook. A change will only be effective if no error number with the error type selected here is available in the standard device code **C00165**.		
Value is bit-coded:		**Info**
Bit 0	Error	
Bit 1	Trouble	
Bit 2	Quick stop by trouble	
Bit 3	Warning locked	
Bit 4	Warning	
Bit 5	Information	
Bit 6	Reserved	
Bit 7	Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13899

Parameter | Name: **C13899 | Station alias**			Data type: UNSIGNED_16 Index: 10676_d = $29B4_h$
This code serves to set a station alias address. In order to use a station alias address, you must select a value > '0'. • The station alias address must only be set if the node is part of a "hot connect" group. • The station alias address must be unambiguous and may only be assigned once within the EtherCAT network. • Use the same station alias address in the EtherCAT master and in the slave. ▸ Address allocation (🕮 36)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		32767	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13900

Parameter | Name: **C13900 | Firmware: Product type**	Data type: VISIBLE_STRING Index: 10675_d = $29B3_h$
The code contains a string with a length of 8 bytes. The following identification code is output: • "E84AFCET" (up to V01.05) • "E84AFYET" (from V02.00)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 10674_d = $29B2_h$
The code contains a string with a length of 20 bytes. The creation date ("MM DD YYYY") and the time ("hh:mm:ss") of the software are output here. Example: "Mar 21 2005 12:31:21"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware version**	Data type: VISIBLE_STRING Index: 10673_d = $29B1_h$
The code contains a string with a length of 5 bytes. The firmware version is output here. Example: "01.00"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

13.3 Table of attributes

The table of attributes contains information required for communicating with the Inverter Drive 8400 via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter designation	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index by which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze- code	Only required for access via a bus system
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of the array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes, with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte, without sign
			UNSIGNED_16	2 bytes, without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ no decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing only possible if the controller is inhibited (CINH)	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13850	All words to master	10725	29E5	A	16	UNSIGNED_16	1	☑		
C13851	All words from master	10724	29E4	A	16	UNSIGNED_16	1	☑		
C13852	All words to standard device	10723	29E3	A	16	UNSIGNED_16	1	☑		
C13853	All words from standard device	10722	29E2	A	16	UNSIGNED_16	1	☑		
C13859	Number of Tx PDOs	10716	29DC	E	1	UNSIGNED_16	1	☑		
C13860	Number of Rx PDOs	10715	29DB	E	1	UNSIGNED_16	1	☑		
C13861	Bus state	10714	29DA	E	1	UNSIGNED_16	1	☑		
C13864	Active station address	10711	29D7	E	1	UNSIGNED_16	1	☑		
C13867	Display of emergency data	10708	29D4	E	1	OCTET_STRING		☑		
C13879	Bus error	10696	29C8	E	1	UNSIGNED_16	1	☑		
C13880	Reaction on communication failure	10695	29C7	A	2	UNSIGNED_8	1	☑	☑	
C13881	Response time when exiting "Operational"	10694	29C6	E	1	UNSIGNED_16	1	☑	☑	
C13883	DC active	10692	29C4	E	1	UNSIGNED_8	1	☑		
C13884	Synchronisation is active	10691	29C3	E	1	UNSIGNED_8	1	☑		
C13885	Clear process data	10690	29C2	E	1	UNSIGNED_8	1	☑	☑	
C13887	Suppress emergency message in case of	10688	29C0	E	1	BITFIELD_8		☑	☑	
C13899	Station alias address	10676	29B4	E	1	UNSIGNED_16	1	☑	☑	
C13900	Firmware product type	10675	29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware compilation date	10674	29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware version	10673	29B1	E	1	VISIBLE_STRING		☑		

Index

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you notice this, please send your suggestions and points of criticism in a short e-mail to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

E84AYCET communication module (EtherCAT) · Communication Manual · EDS84AYCET · 13429553 · DMS 5.0 EN · 05/2013 · TD17

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany
+49 (0)51 54 / 82-0
+49 (0)51 54 / 82-28 00
Lenze@Lenze.de
www.Lenze.com

Service

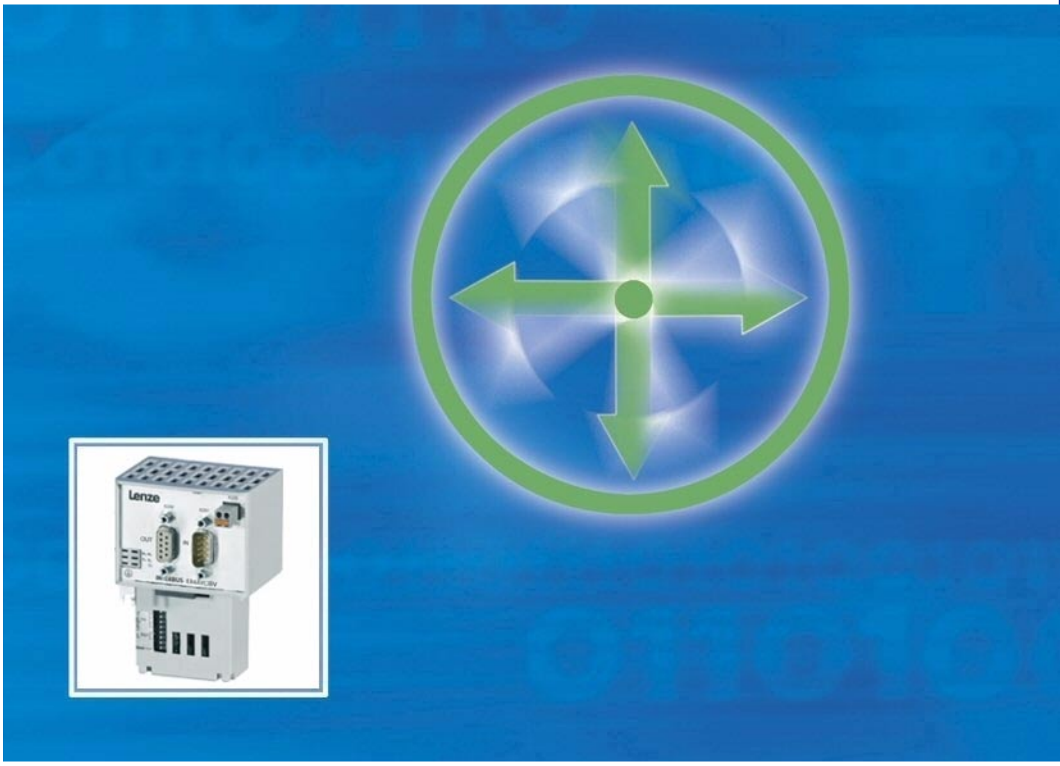
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
00 80 00 / 24 4 68 77 (24 h helpline)
+49 (0)51 54 / 82-11 12
Service@Lenze.de

EDS84AYCIB
13362260

L-force *Communication*

Communication Manual

8400



E84AYCIB

INTERBUS communication module

Lenze

Contents

1 About this documentation

Contents

The descriptions contained in this documentation refer only to the E84AYCIB communication module (INTERBUS).

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **"Inverter Drives 8400" hardware manual**.

The features and functions of the communication module are described in detail.

Typical applications are illustrated by means of examples.

This documentation also contains:

- Safety instructions that must be observed
- Key technical data relating to the communication module
- Information about the versions of the Lenze standard devices to be used
- Notes on troubleshooting and fault elimination

The theoretical concepts are only explained to the level of detail required to understand the function of the communication module.

Depending on the software version of the controller used and the version of the »Engineer« software installed, the screenshots provided in this documentation may not match up with those displayed by your »Engineer«.

This documentation does not describe any software provided by other manufacturers. No liability can be accepted for corresponding data provided in this documentation. For information on how to use the software, please refer to the host system (master) documents.

All product names mentioned in this documentation are trademarks of their respective owners.

Tip!

Detailed information on the INTERBUS can be found on the internet page of the INTERBUS user organisation:

www.interbusclub.com

Target group

This documentation addresses to people involved in configuring, installing, commissioning and maintaining a machine's networking and remote maintenance features.

Tip!

You will find documentation and software updates relating to Lenze products on the Internet, in the "Services & Downloads" area at:

www.Lenze.com

Validity information

The information in this documentation applies to the following devices:

Extension module	Type designation	From hardware version	From software version
INTERBUS communication module	E84AYCIB	VA	01.00

1.1 Document history

Material number	Version			Description
13295951	1.0	06/2009	TD17	First edition
13320780	2.0	11/2009	TD17	Parameter descriptions updated.
13329723	3.0	02/2010	TD17	Chapters "Diagnostics" and "Installation" updated.
13362260	4.0	11/2010	TD17	General revision

Your opinion is important to us!

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you notice this, please send your suggestions and points of criticism in a short e-mail to:

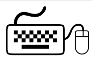
feedback-docu@Lenze.de

Thank you very much for your support.

Your Lenze documentation team

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Identification	Examples/notes
Spelling of numbers		
Decimal	Standard notation	Example:1234
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In quotation marks Point	Example: ’100’ Example: ’0110.0100’
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Text		
Program name	» «	PC software Example: Lenze »Engineer«
Control element	**Bold**	The **OK** button ... / The **Copy** command ... / The **Properties** tab ... / The **Name** input field ...
Hyperlink	Underlined	Optically highlighted reference to another subject which is activated with a mouse-click.
Symbols		
Page reference	(🕮 9)	Optically highlighted reference to another page which is activated with a mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Standard device	Lenze frequency inverter from the "Inverter Drives 8400" product series, with which the communication module can be used.
Controller	Application as directed (14)
»Engineer«	Lenze software, supporting you during the entire life cycle of a machine - from the planning phase to maintenance.
Code	"Container" for one or more parameters which you can use to parameterise or monitor the communication module.
Subcode	If a code contains several parameters, the individual parameters are stored under "subcodes". This manual uses a slash "/" as a separator between the code and subcode (e.g. "C00118/3").
PCP	Peripherals Communication Protocol (parameter data transmission)
PD	Process data / process data words
PDO	Process data object
PDU	Process Data Unit
PMS	Peripheral Message Specification
HW	Hardware
SW	Software

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

> **Pictograph and signal word!**
>
> (characterise the type and severity of danger)
>
> **Note**
>
> (describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for simple handling
		Reference to another documentation

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety instructions and application notes

- Lenze drive components ...
  - may only be used as directed.
    Application as directed (📖 14)
  - must never be commissioned if they display any signs of damage.
  - must never be modified technically.
  - must never be commissioned if they are not fully mounted.
  - must never be operated without the required covers.
  - can have live, moving and rotating parts during operation, depending on their degree of protection. Surfaces can be hot.
- The following applies to Lenze drive components ...
  - Only use accessories that have been approved for the product.
  - Only use genuine spare parts supplied by the manufacturer of the product.
- Observe all regulations for the prevention of accidents, directives and laws that apply to the location of use.
- Observe all the specifications contained in the enclosed documentation.
  - This is a precondition for ensuring safe, trouble-free operation and for making use of the stated product features.
    Product features (📖 15)
  - The specifications, processes, and circuitry described in this documentation are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.
- All works on and with Lenze drive components may only be carried out by qualified personnel. According to IEC 364 and CENELEC HD 384 these are persons who ...
  - are familiar with installing, mounting, commissioning and operating the product.
  - have the qualifications necessary for their occupation.
  - know and are able to apply all regulations for the prevention of accidents, directives and laws that apply to the location of use.

2.2 Device- and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- Use a safely separated power supply unit in accordance with EN 61800-5-1 ("SELV"/ "PELV").
- Only use cables that meet the listed specifications.
  Specification of the bus cable (📖 30)

Documentation for the standard device, control system, plant/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

- If Inverter Drives 8400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to provide reliable protection against accidental contact.
  Protective insulation (📖 18)

Device protection

- The communication module contains electronic components that can be damaged or destroyed by electrostatic discharge.
  Installation (📖 23)

3 Product description

3.1 Application as directed

The communication module ...

- Is an accessory module that can be used in conjunction with the following standard devices:

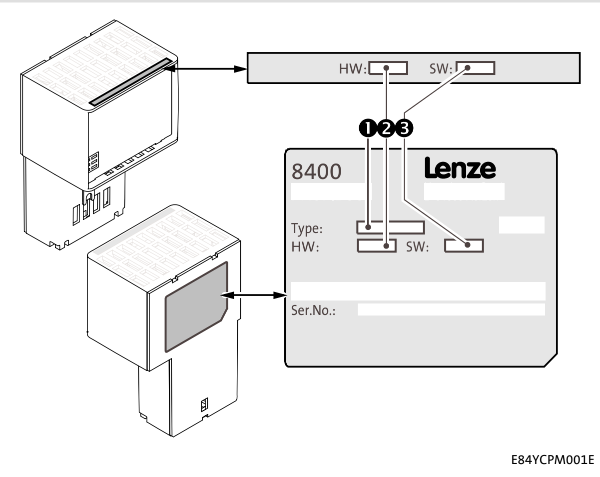
Product series	Type designation	From software version
Inverter Drives 8400 StateLine	E84AVSCxxxxx	04.00
Inverter Drives 8400 HighLine	E84AVHCxxxxx	04.00
Inverter Drives 8400 TopLine	E84AVTCxxxxx	01.00

- Is an item of equipment intended for use in industrial power systems
- Should only be used under the operating conditions prescribed in this documentation
- Should only be used in INTERBUS networks

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation, hardware version, and software version of a communication module are indicated on its nameplate:



1 Type designation (type)

E84	Product series
A	Version
Y	Module identification: extension module
C	Module type: communication module
IB	INTERBUS
V/S	V: Coated version
	S: Standard version

2 Hardware version (HW)

3 Software version (SW)

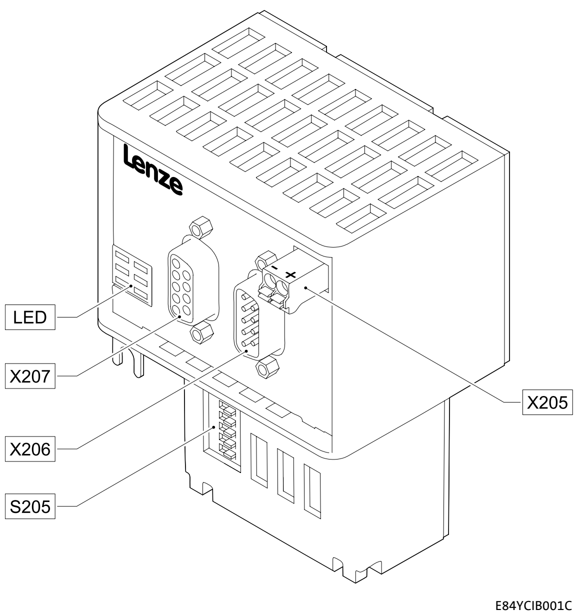
[3-1] Identification data

3.3 Product features

- Interface module for the INTERBUS communication system for attachment to the expansion slots of Inverter Drives 8400
- The communication module can either be supplied internally by the 8400 standard device or externally by a separate voltage source.
- Slave functionality
- Access to all Lenze parameters
- DIP switch settings:
  - Number of process data words and parameter data words
  - Baud rate (500 kbps or 2 Mbps)
- Bus coupling via remote bus in accordance with the RS485 standard
- Up to 10 process data words are possible
- Support of the PMS services:
  - Initiate
  - Abort
  - Reject
  - Read
  - Write
  - Get-OD
  - Identify
  - Status

3.4 Terminals and interfaces

- 2 connections for the INTERBUS
  - 1 input (9-pole Sub-D plug)
  - 1 output (9-pole Sub-D socket)
- 2-pole plug connector with spring connection for the external voltage supply of the communication module
- DIP switch for setting the ...
  - Number of process data words and parameter data words
  - Baud rate
- Front LEDs for diagnostics ...
  - of the communication module voltage supply;
  - of the module status;
  - of the INTERBUS status.



S205 DIP switch for setting the ...
- Number of process data words and parameter data words
- Baud rate

▸ Possible settings through DIP switch (📖 34)

X205 External voltage supply of the communication module
- 2-pole plug connector with spring connection

▸ External voltage supply (📖 31)

X206 INTERBUS input (IN)
- 9-pole Sub-D plug

X207 INTERBUS output (OUT)
- 9-pole Sub-D socket

▸ Network topology (📖 27)
▸ INTERBUS connection (📖 28)

MS ME BS BE DE 5 LED status displays for diagnostics

▸ Module status displays (📖 55)
▸ Fieldbus status displays (📖 56)

[3-2] E84AYCIB communication module (INTERBUS)

4 Technical data

"Inverter Drives 8400" hardware manual

This manual contains data on the **ambient conditions** and the **electromagnetic compatibility (EMC)** that also apply to the communication module.

4.1 General data and operating conditions

Field	Values
Order designation	• E84AYCIBV (coated version) • E84AYCIBS: (standard version)
Communication profile	INTERBUS
Interfaces	• Input (IN): 9-pole Sub-D plug • Output (OUT): 9-pole Sub-D socket
Communication medium	RS485
Network topology	Ring
Type of node	INTERBUS slave
Number of nodes	• 1 master • 512 slaves
Baud rate	500 kbps or 2 Mbps (can be set via DIP switch or code)
Max. cable length	• 400 m at 500 kbps • 150 m at 2 Mbps (between the individual INTERBUS nodes)
Process data words (PD) to be used	0 ... 10: 16 bits/word (can be set via DIP switch or code)
Parameter data words (PCP)	0, 1, 2, 4: 16 bits/word (can be set via DIP switch or code)
Max. number of data words	10 (PD + PCP): 16 bits/word
Max. PDU length	64 bytes
INTERBUS identification (module ID)	• 3 = 0x3 (PCP 0 words) • 227 = 0xE3 (PCP 1 word) • 224 = 0xE0 (PCP 2 words) • 225 = 0xE1 (PCP 4 words)
Voltage supply	External supply via the 2-pole plug connector • "+": U = 24 V DC (21.6 V - 0 % ... 26.4 V + 0 %), I_{max} = 180 mA • "-": Reference potential for external voltage supply
Conformities, approvals	CE

4.2 Protective insulation

Danger!

Dangerous electrical voltage

If Inverter Drives 8400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to provide reliable protection against accidental contact.

Possible consequences:

- Death or serious injury

Protective measures:

- If protection against accidental contact needs to be provided for the control terminals of the controller and for the connections of the plugged-in device modules, ...
  – a double isolating distance must exist.
  – the components to be connected must be provided with the second isolating distance.

Note!

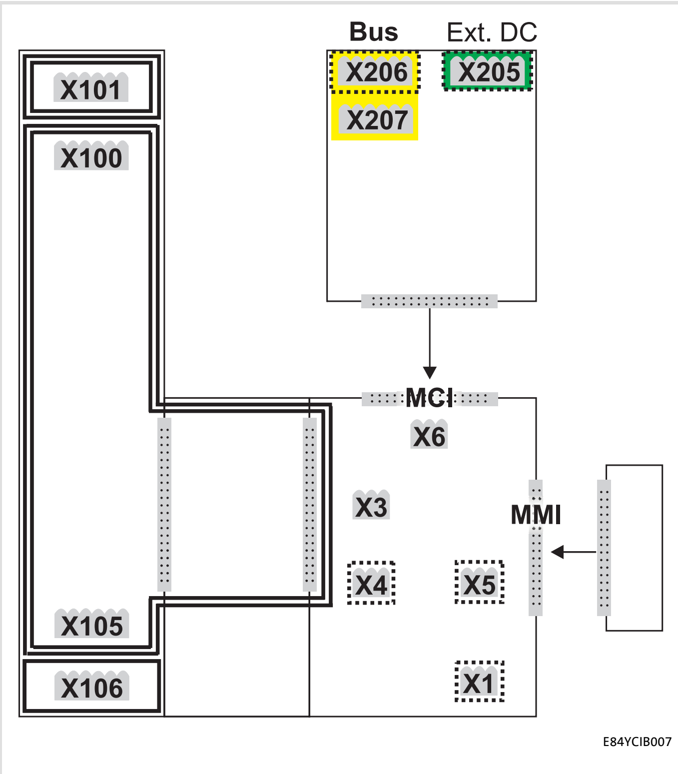
The protective insulation provided in Inverter Drives 8400 is realised in accordance with EN 61800-5-1.

The illustration below ...

- shows the arrangement of the terminal strips and the separate potential areas of the controller.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.

Note!

The INTERBUS input (X206) is isolated from the voltage supply (X205) and the INTERBUS output (X207).



|| Reinforced insulation

| Basic insulation

¦ Functional insulation

[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection
X100	Mains/DC-bus connection
X101	Relay contact
X105	Motor/brake resistor
X106	Motor PTC
X1	System bus (CANopen)
X3	Analog inputs/outputs
X4	Digital outputs
X5	Digital inputs
X6	Diagnostics
MCI	Slot for communication module
MMI	Slot for memory module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MCI and the mains terminal X100?

- The separate potential area with the better protective insulation is decisive.
  - The separate potential area of the device module bus terminal is "functionally insulated".
  - The separate potential area of the mains terminal has a "reinforced insulation".
- Result: The insulation between the mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.3 Protocol data

Field	Values
Process data words	0 ... 10 words (16 bits/word)
Supported PMS services	• Initiate • Abort • Reject • Read • Write • Get-OD • Identify • Status

4.4 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in the INTERBUS network depend on the...

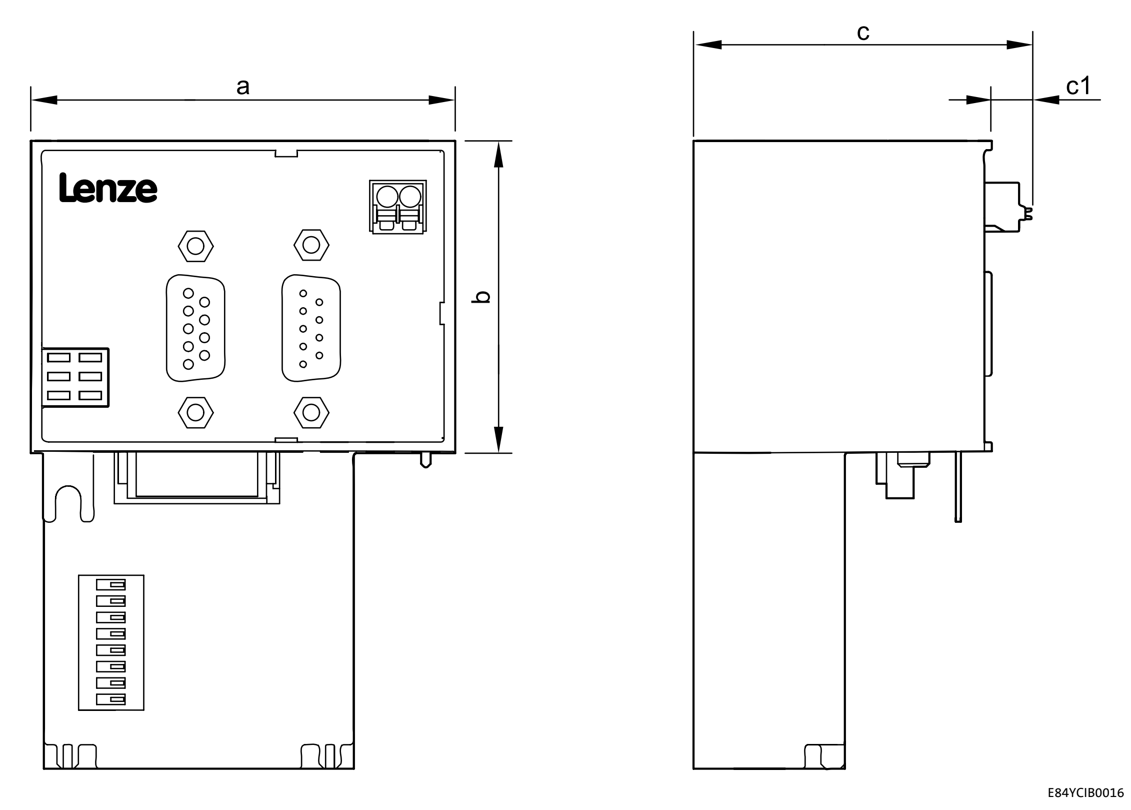
- Processing time in the controller;
- Telegram runtime (baud rate / telegram length).

Processing time in the controller

The parameter data and process data are independent of each other.

Data	Processing time
Process data	Approx. 2 ms + 1 ms tolerance + runtime of the technology application used
Parameter data	Approx. 30 ms + 20 ms tolerance (typical) • Some codes may require a longer processing time (see software manual/ »Engineer« online help for Inverter Drives 8400).

4.5 Dimensions



[4-2] Dimensions

Type	Dimensions [mm]			
	a	b	c	c1
E84AYCIB	67	50	57	8

5 Installation

Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is damaged.
- Fieldbus communication is not possible or is faulty.

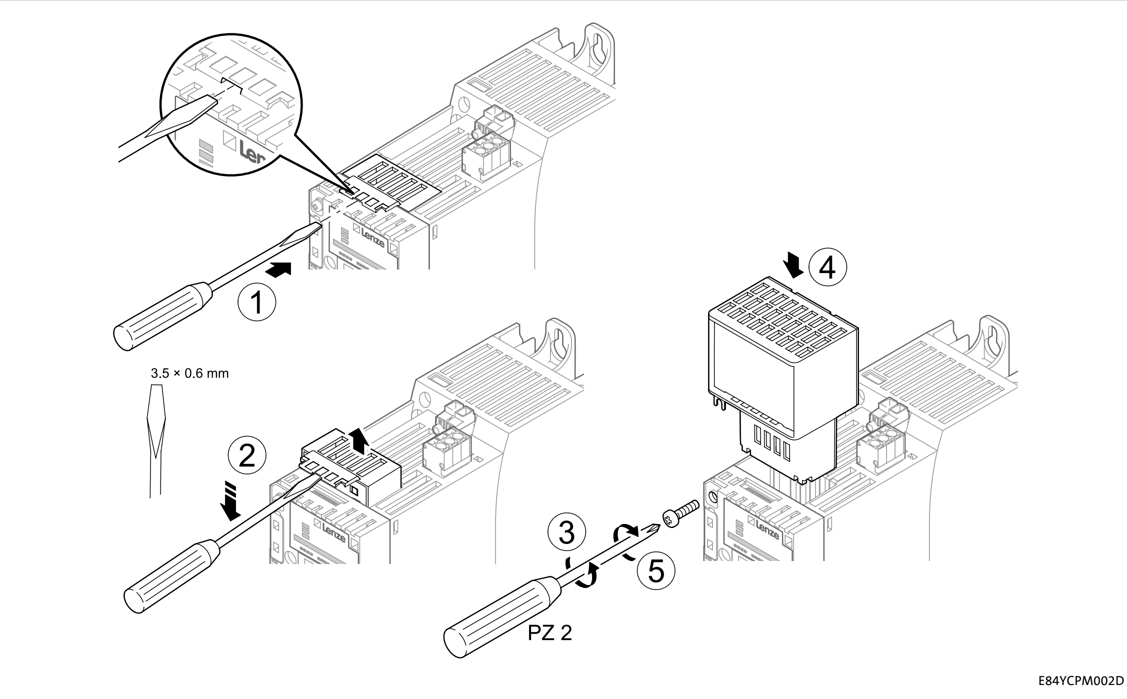
Protective measures:

- Ensure that you are free of electrostatic charge before you touch the module.

5.1 Mechanical installation

The communication module can be plugged into the MCI slot or removed from there while the controller is switched on. When the module is plugged in, it is recognised automatically and checked for plausibility regarding its function and version.

5.1.1 Mounting for standard devices 0.25 kW and 0.37 kW

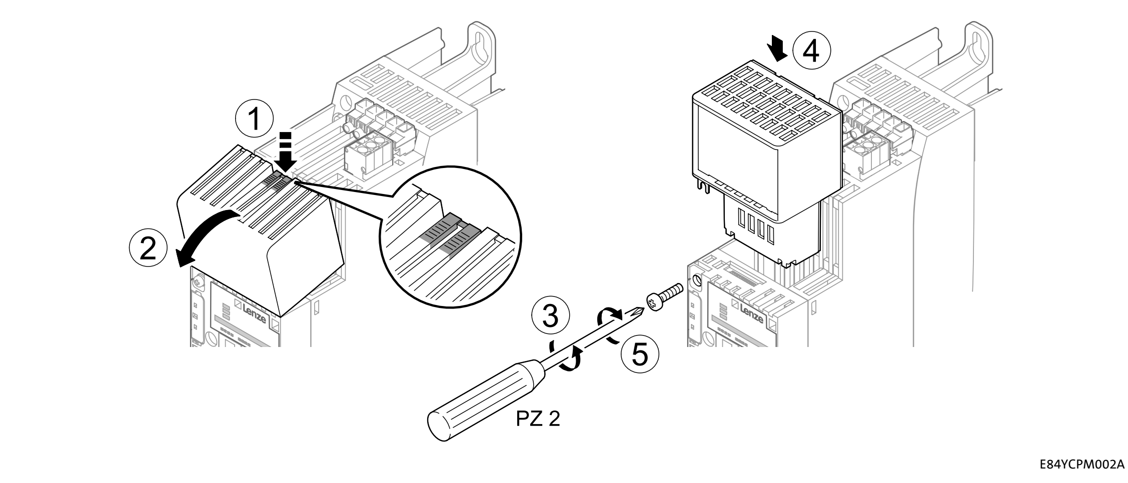


[5-1] Mounting for standard devices 0.25 kW and 0.37 kW

Mounting steps

1. Use a screwdriver to lever out the cover of the MCI slot of the standard device and remove it (1, 2).
2. Loosen the securing screw for the communication module at the standard device (3).
3. Insert the communication module into the MCI slot of the standard device (4).
4. Tighten the securing screw again (5).

5.1.2 Mounting for standard devices from 0.55 kW onwards

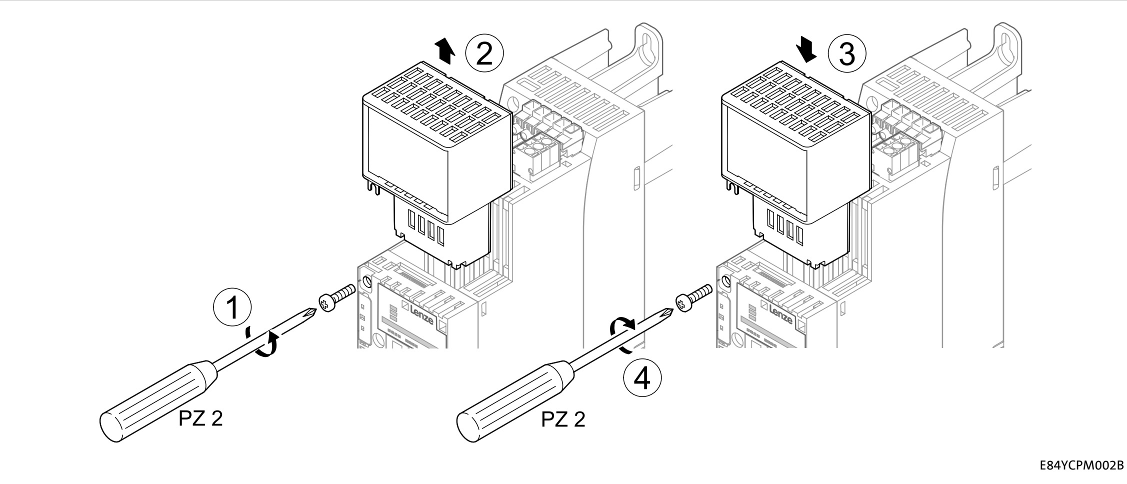


[5-2] Mounting for standard devices from 0.55 kW onwards

Mounting steps

1. Slightly impress the pressure surface on the top side of the standard device MCI slot cover (1).
2. Bend the cover forward and remove it from the standard device (2).
3. Loosen the securing screw for the communication module at the standard device (3).
4. Insert the communication module into the MCI slot of the standard device (4).
5. Tighten the securing screw again (5).

5.1.3 Replacing the communication module



[5-3] Replacing the communication module

Mounting steps

1. Loosen the securing screw for the communication module at the standard device (1).
2. Remove the communication module from the MCI slot of the standard device (2).
3. Insert the new communication module into the MCI slot of the standard device (3).
4. Tighten the securing screw again (4).

5.2 Electrical installation

Documentation for the standard device, control system, plant/machine

Observe the notes and wiring instructions contained in this documentation.

5.2.1 EMC-compliant wiring

In typical systems, standard shielding of the Ethernet cables is sufficient.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

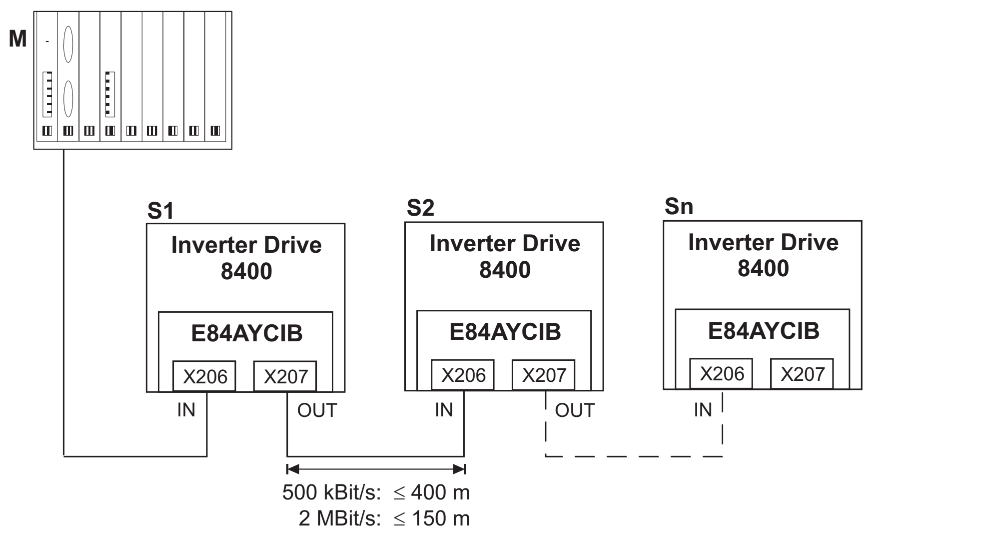
For this observe the following notes:

1. Remove the plastic sheath of the cable on a length of 2 cm.
2. Fasten the cable shield onto the shield contact of the standard device.

5.2.2 Network topology

The bus system must be designed as a ring. Feed and return lines must be integrated in the same bus cable. The ring goes from the INTERBUS master via all other nodes and back again to the master.

An INTERBUS ring can consist of maximally 513 nodes (1 master + standard devices connected).

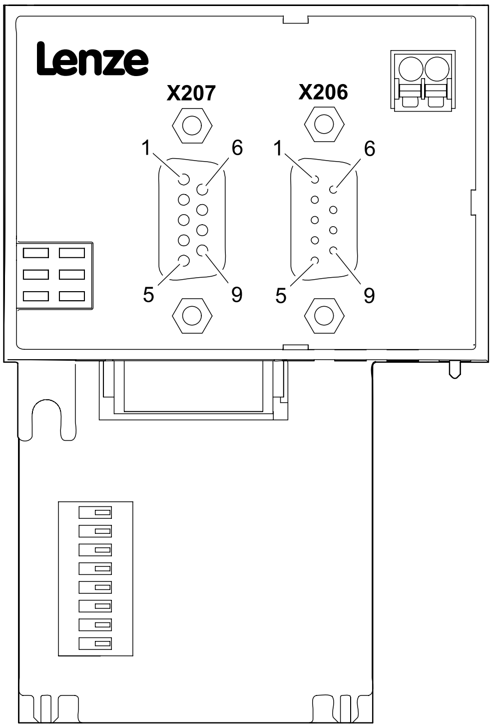


M: Master (master control, e.g. PLC, Industrial PC)
S1 ... Sn: Slaves

[5-4] INTERBUS ring

5.2.3 INTERBUS connection

The INTERBUS connection of the communication module is effected via **X206** (input, 9-pole Sub-D plug) and **X207** (output, 9-pole Sub-D socket).



[5-5] INTERBUS terminals X206 (input) and X207 (output)

The nodes on the bus system have to be connected to each other by means of a fieldbus cable in accordance with the INTERBUS specification. INTERBUS cables are for instance produced by PHOENIX CONTACT (Germany).

▸ Specification of the bus cable (📖 30)

Assignment of the 9-pole Sub-D plug X206 (IN)

Pin	Designation	Input/output	Description
1	DO1	Input	RS485: DO1 not inverted
2	DI1	Output	RS485: DI1 not inverted
3	GND		Reference potential
4	Free		Not assigned
5	Vcc5	Output	5 V DC
6	/DO1	Input	RS485: DO1 inverted
7	/DI1	Output	RS485: DI1 inverted
8	Vcc5	Output	5 V DC
9	Free		Not assigned

Assignment of the 9-pole Sub-D socket X207 (OUT)

Pin	Designation	Input/output	Description
1	DO2	Output	RS485: DO2 not inverted
2	DI2	Input	RS485: DI2 not inverted
3	GND		Reference potential
4	GND		
5	Vcc5	Output	5 V DC
6	/DO2	Output	RS485: DO2 inverted
7	/DI2	Input	RS485: DI2 inverted
8	Vcc5	Output	5 V DC
9	RBST	Signalling input	Connection to the outgoing INTERBUS is plugged.

5.2.4 Specification of the bus cable

Note!

Only use cables that meet the listed specifications.

Specification of the INTERBUS cable	
Cable type	Sold by the meter (e.g. PHOENIX CONTACT: IBS RBC meter-T, order no. 28 06 28 6)
No. of conductors	3 x 2, twisted in pairs, with joint shielding
Conductor cross-section	> 0.2 mm^2
DC-cable resistance	< 96 Ω/km
Impedance (characteristic)	• 120 Ω ± 20 % (f = 64 kHz) • 100 Ω ± 15 Ω (f > 1 MHz)
Capacitance per unit length	< 60 nF/km (f = 800 Hz)

5.2.5 Bus cable length

Adapt the baud rate according to the bus cable length:

Baud rate	Cable length between the individual INTERBUS nodes
500 kbps	Max. 400 m
2 Mbps	Max. 150 m

Note!

Select the baud rate which depends on the data volume, cycle time, and number of nodes only as high as required for the application.

5.2.6 External voltage supply

The communication module can be supplied externally with voltage via separate supply cables at the 2-pole plug connector (X205).

Note!

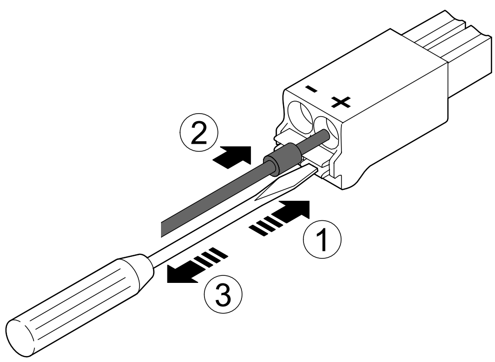
With external voltage supply, always use a separate power supply unit, safely separated in accordance with EN 61800-5-1 in every control cabinet ("SELV" / "PELV").

- External voltage supply of the communication module is necessary if the bus communication is to continue when the supply of the standard device fails.
- It is not possible to access the parameters of a standard device that is disconnected from the mains.

Wiring the X205 plug connector

Stop!

Only wire the plug connector if the standard device is disconnected from the mains.



[5-6] Wiring of the 2-pole plug connector with spring connection

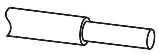
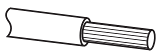
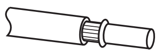
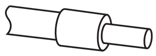
How to wire the plug connector with spring connection:

1. Place a screwdriver into the notch below the terminal and keep it pressed.
2. Place the supply cable in the terminal.
3. Remove the screwdriver from the notch.

Assignment of the X205 plug connector

Designation	Explanation
+	U = 24 V DC (21.6 V - 0 % ... 26.4 V + 0 %) I = 180 mA
-	Reference potential for external voltage supply

Terminal data

Field	Values	
Electrical connection	2-pole plug connector with spring connection	
Possible connections	Rigid:	
		0.2 ... 1.5 mm^2 (AWG 24 ... 16)
	Flexible:	
		Without wire end ferrule 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, without plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, with plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
Stripping length	10 mm	

6 Commissioning

During commissioning, system-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the controller. For Lenze devices this is effected via the so-called codes.

The codes of the controller and for communication are saved non-volatilely as a data set in the memory module.

Additionally there are codes for diagnostics and monitoring of the nodes.

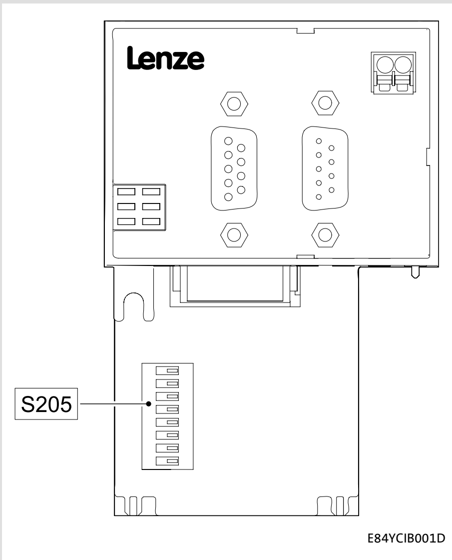
▸ Parameters of the communication module (📖 63)

6.1 Before initial switch-on

Stop!

Before switching on the standard device with the communication module for the first time, check the entire wiring for completeness, short circuit, and earth fault.

6.2 Possible settings through DIP switch



[6-1] DIP switch

The following can be set through the DIP switch (S205):

- Number of process data words (PD)
  Switches: 1 ... 4 (35)
- Number of parameter data words (PCP)
  Switches: 5 and 6 (36)
- Baud rate
  Switch: 8 (37)

Lenze setting: All switches "OFF"

Switch 7 has no function.

Note!

To make any modified settings take effect, switch the voltage supply of the communication module off and then on again.

Initial switch-on (39)

The settings can also be made via codes:

- All DIP switches = "OFF" (Lenze setting):
  At switch-on, the configuration from codes C13892, C13893, and C13894 becomes active.
- At least one DIP switch = "ON":
  At switch-on, the values are accepted from the switch positions.

The data word sum (PD + PCP) may maximally amount to 10 words.

6.2.1 Setting the number of process data words (PD)

- The number of process data words (PD) can be set via switches 1 ... 4 or code C13893 (for this, see Settings in the »Engineer« (📖 38)).
- 0 ... 10 process data words can be used.
- The current setting is displayed in C13860/2.

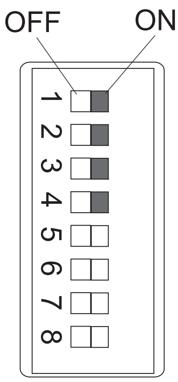
Note!

The data word sum (PD + PCP) must amount to 1 ... 10 words. Impermissible settings are reported by the LED "BE" (red blinking).

▸ LED status displays (📖 54)

The communication module then continues to operate internally with the following values:

- PD = 2 (words)
- PCP = 1 (word)

DIP switch	Number of PD	Switch				Max. number of PCP
		1	2	3	4	
	0	OFF	OFF	OFF	OFF	4
	1	OFF	OFF	OFF	**ON**	
	2	OFF	OFF	**ON**	OFF	
	3	OFF	OFF	**ON**	**ON**	
	4	OFF	**ON**	OFF	OFF	
	5	OFF	**ON**	OFF	**ON**	
	6	OFF	**ON**	**ON**	OFF	
	7	OFF	**ON**	**ON**	**ON**	2
	8	**ON**	OFF	OFF	OFF	
	9	**ON**	OFF	OFF	**ON**	1
	10	**ON**	OFF	**ON**	OFF	0

6.2.2 Setting the number of parameter data words (PCP)

- The number of parameter data words (PCP) can be set via switches 5 and 6 or code C13892 (for this, see Settings in the »Engineer« (📖 38)).
- 0, 1, 2 or 4 parameter data words can be used.
- The current setting is displayed in C13860/1.

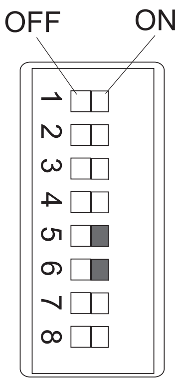
Note!

The data word sum (PD + PCP) must amount to 1 ... 10 words. Impermissible settings are reported by the LED "BE" (red blinking).

▸ LED status displays (📖 54)

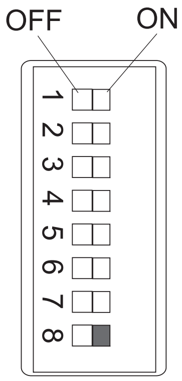
The communication module then continues to operate internally with the following values:

- PD = 2 (words)
- PCP = 1 (word)

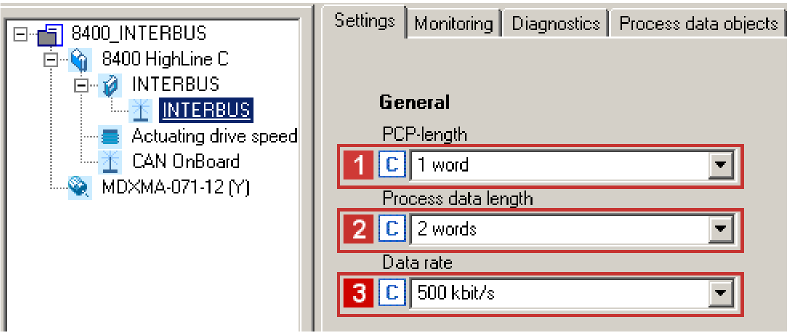
DIP switch	Number of PCP	Switch		Max. number of PD	ID code [hex]
		5	6		
	0	OFF	OFF	10	0x03
	1	OFF	**ON**	9	0xE3
	2	**ON**	OFF	8	0xE0
	4	**ON**	**ON**	6	0xE1

6.2.3 Setting the baud rate

- The baud rate can be set via switch 8 or code C13894 (for this, see Settings in the »Engineer« (📖 38)).
- The current setting of the baud rate is displayed in C13863/.

DIP switch	Switch 8	Baud rate	Max. cable length between adjacent nodes
	OFF	500 kbps	400 m
	ON	2 Mbps	150 m

6.3 Settings in the »Engineer«



Under the **Settings** tab in the »Engineer« you can set the following parameters:

- 1 **Number of parameter data words** (PCP, C13892)
- 2 **Number of process data words** (PD, C13893)
- 3 **Baud rate** (C13894)

6.4 Initial switch-on

Documentation for the standard device

Observe the safety instructions and information on residual hazards contained in this documentation.

Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

After communication has been established, the power on/off state of the standard device is irrelevant.

Activating altered settings

To activate altered settings, ...

- use standard device code **C00002** to execute the device command "11: save all parameter sets", and ...
- then switch the voltage supply of the communication module off and on again.

Protection against uncontrolled restart

Following a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting for Inverter Drives 8400, restart protection is active.

The restart behaviour of the controller can be set using **C00142** ("autostart option"):

- **C00142 = 9** (Lenze setting)
  – The controller remains inhibited (even when the fault is no longer active).
  – Bit 0 (inhibit at power-on) and bit 3 (inhibit in the case of undervoltage) are set.
  – An explicit controller enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X4/RFR.
- **C00142 = 8** (enabled)
  – In order to directly enable the device at switch-on, bit 0 must be set to zero (FALSE).
  – An uncontrolled restart of the drive is possible.

7 Data transfer

The INTERBUS master and controller communicate with each other by exchanging data telegrams via INTERBUS. The user data area of the data telegram contains parameter data or process data. In the controller, the parameter data and process data are assigned to different communication channels.

Communication channels

- The process data channel transmits process data.
  - The process data are used to actuate the controller.
  - The host (master) can directly access the process data. In the PLC, for instance, the data are directly assigned to the I/O area.
  - Process data are not saved in the controller.
  - Process data are transmitted cyclically between the host and the controllers (permanent exchange of current input and output data).
  - Process data for instance are setpoints, actual values, control words, and status words.
  - In the case of Inverter Drives 8400 a maximum of 16 process data words (16 bits/ word) can be exchanged for each direction.

Note!

Observe the direction of the flow of information!
- Process input data (Rx data):
  - Process data from the controller (slave) to the master
- Process output data (Tx data):
  - Process data from the master to the controller (slave)

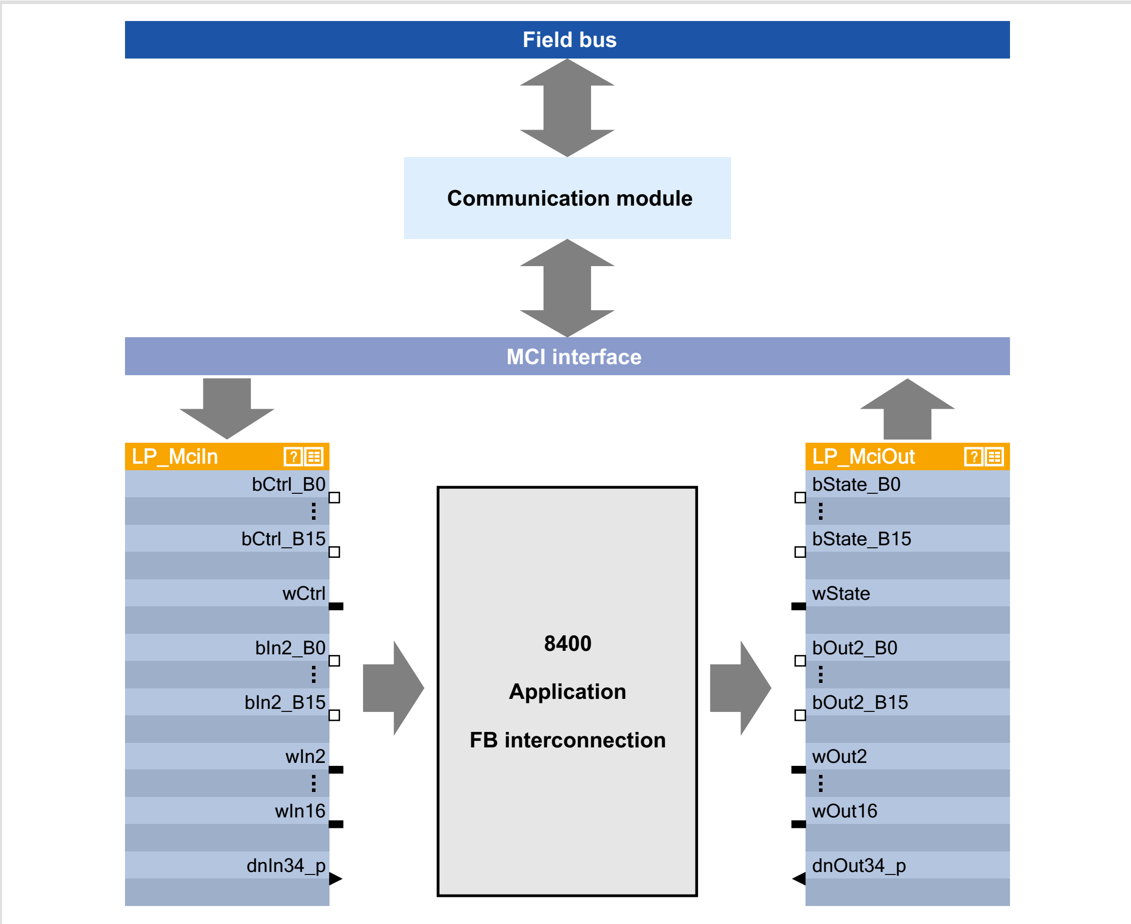
- The parameter data channel transmits parameter data.
  - The parameter data channel provides access to all Lenze codes.
  - The transmission of parameter data usually is not time-critical.
  - Parameter data for instance are operating parameters, motor data, and diagnostics information.
  - The parameter changes must be stored via code **C00002** of the Inverter Drive 8400.
  - The parameter data channel assigns up to 4 words of the input and output data words in the master and is structured identically for both transmission directions.

8 Process data transfer

8.1 Access to process data / PDO mapping

The process data (MCI-PDOs) are transferred via the MCI interface.

- A maximum of 16 words is exchanged for each direction.
- The process data are accessed via the port blocks **LP_MciIn** and **LP_MciOut**. These port blocks are also referred to as process data channels.
- The port block **LP_MciIn** maps the MCI-PDOs received.
- The port block **LP_MciOut** maps the MCI-PDOs to be transmitted.
- The port/function block interconnection of the process data objects (PDO) is made via the Lenze »Engineer«.



[8-1] External and internal data transfer between the bus system, controller, and application

Software manual / »Engineer« online help for the Inverter Drive 8400

Here you'll find detailed information on the port /function block interconnection in the »Engineer« and on port blocks.

Note!

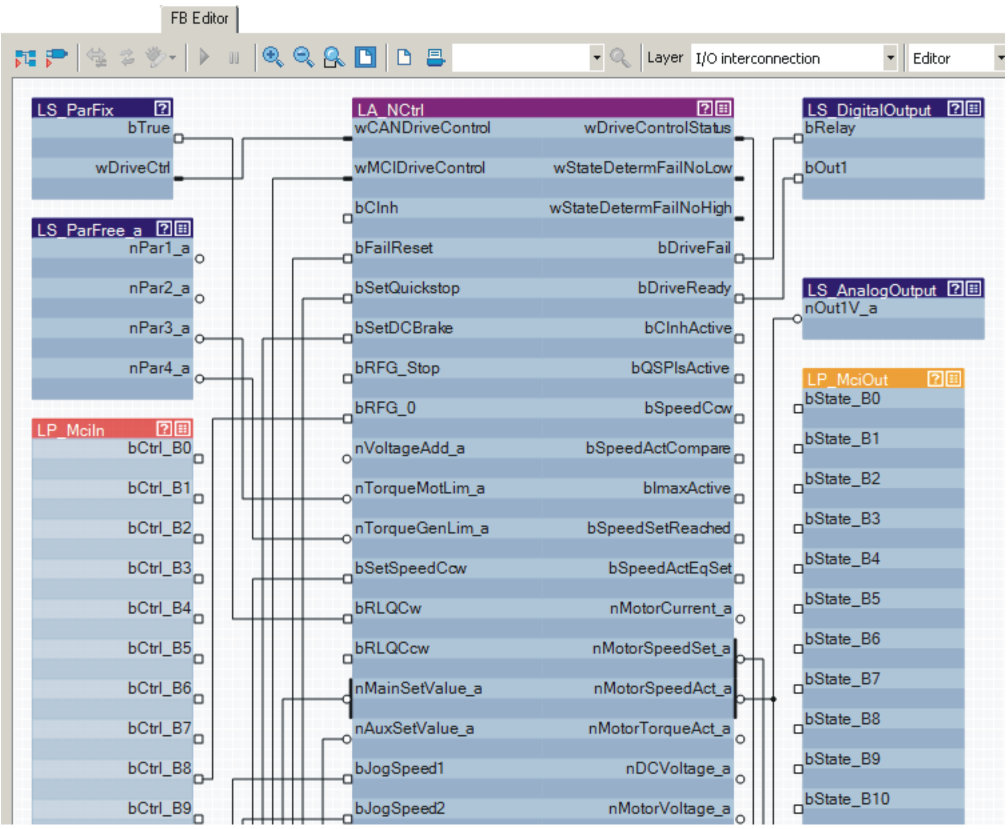
The »Engineer« screenshots shown in the following are only examples for the setting sequence and the resulting displays.

The data in the display fields highlighted in white may differ from the ones of your project.

8.2 Preconfigured port interconnection of the process data objects (PDO)

The preconfigured port interconnection of the process data objects can be activated by setting standard device code **C00007 = "40: MCI"**.

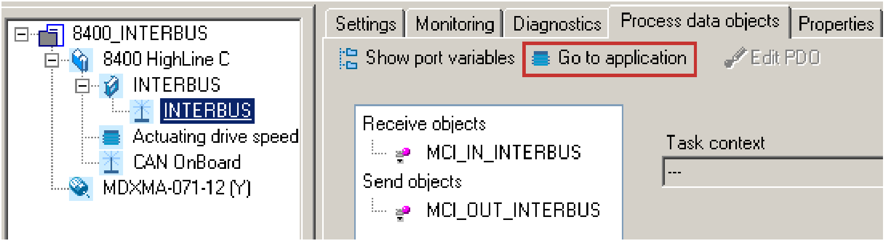
The »FB Editor« serves to display the port blocks "LP_MciIn" and "LP_MciOut" with the preconfigured interconnections:



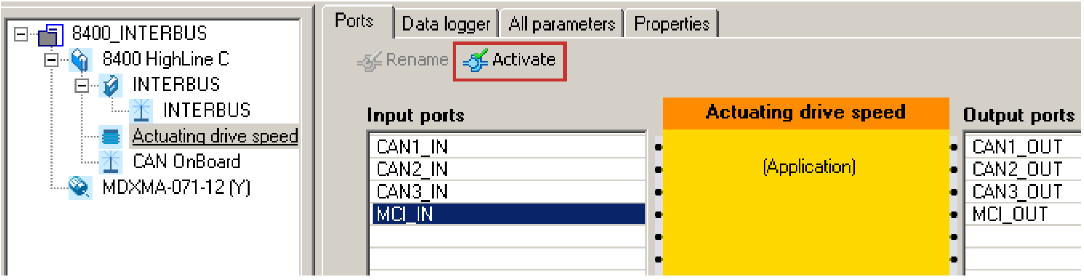
8.3 Freely configurating the port interconnection of the process data objects (PDO)

How to freely configurate the port interconnection in the »Engineer«:

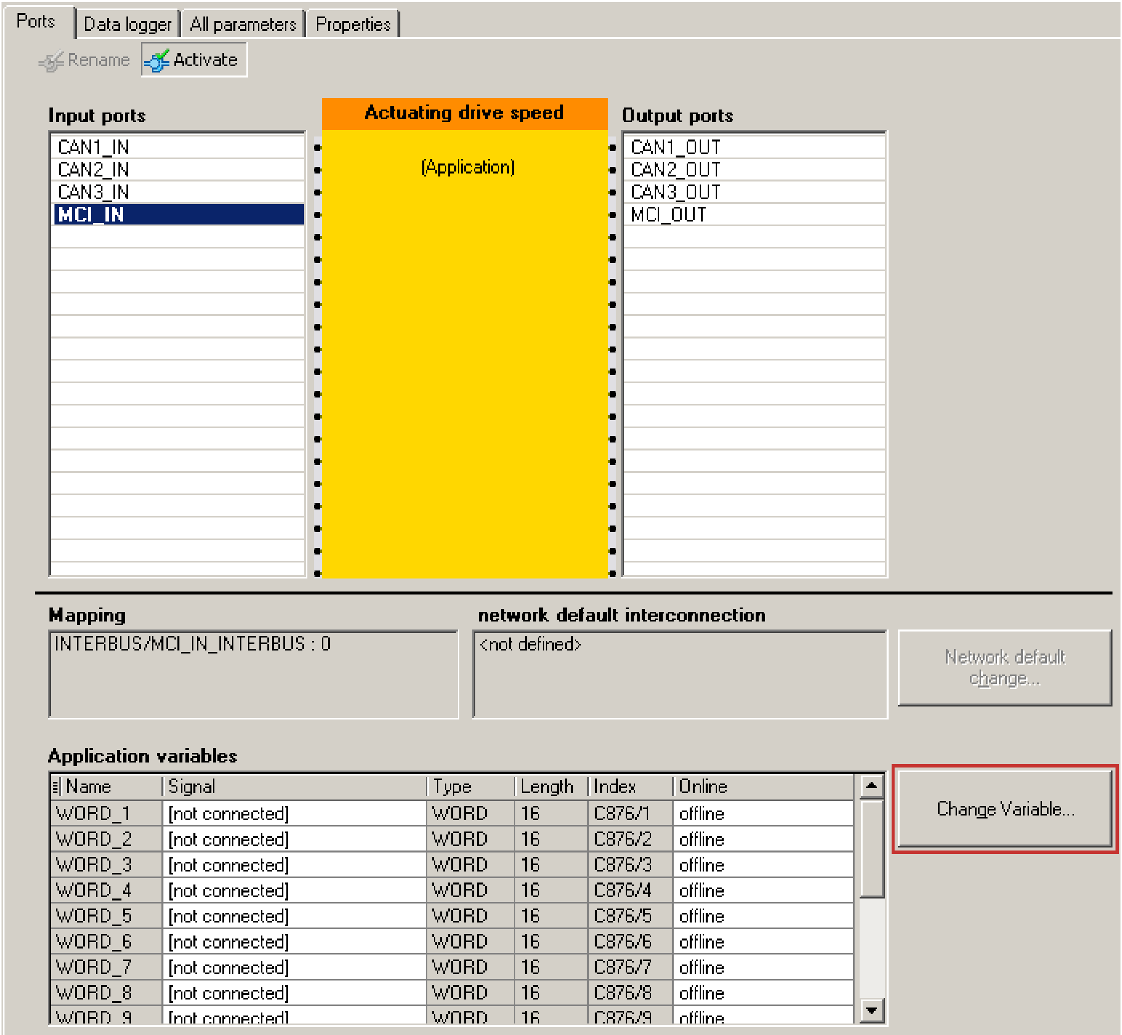
1. Go to the **Process data objects** tab to click the **Go to application** button.



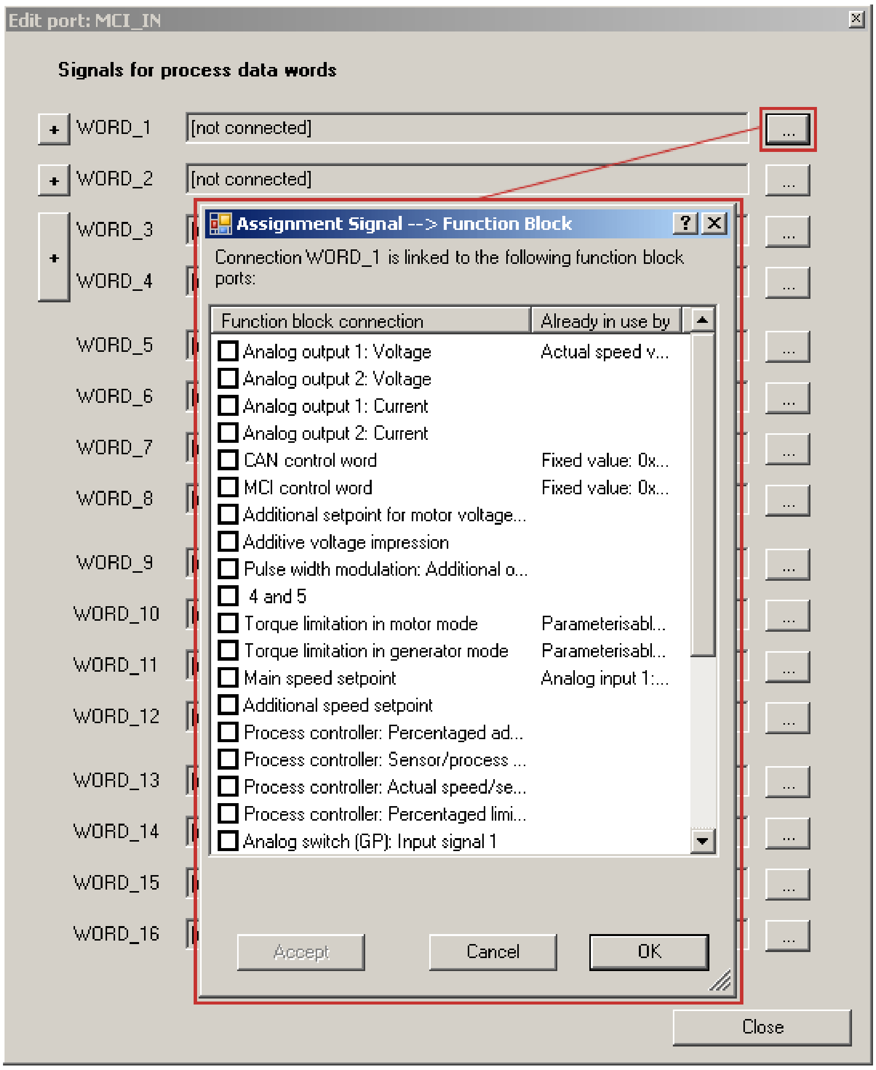
2. Select the "MCI_IN" or "MCI_OUT" port blocks in the **Ports** tab via mouse-click and activate them with the **Activate** button.



3. Click the **Change variable...** button.

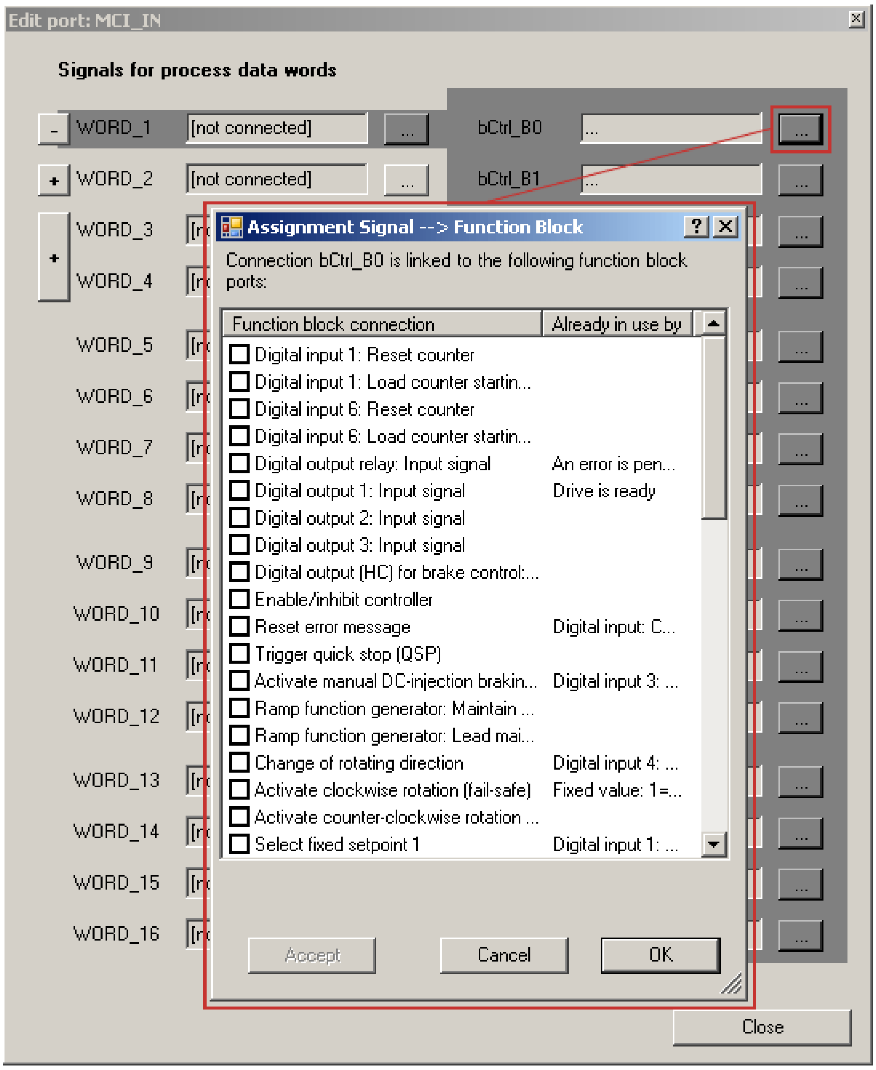


4. The [...] button serves to assign signals to the process data words in the *Assignment signal --> function block* dialog box.
   → Select signals and then click the **OK** button.



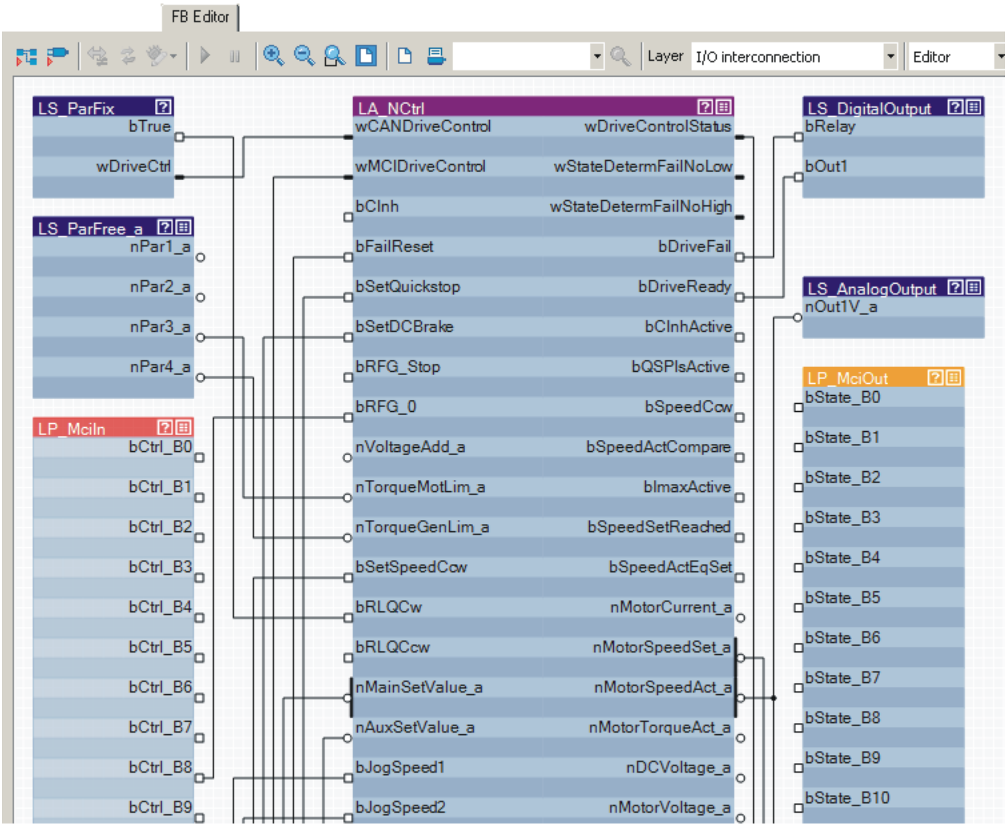
Moreover you can assign signals to the individual control and status bits at the WORD_1... WORD_4 process data words via the + and ... buttons.

→ Select the signals and then click **OK**.



Tip!

When the port blocks "LP_MciIn" and "LP_MciOut" are activated (see 1.), they will be visible in the »FB Editor«. Here you can also assign signals to the process data words.



9 Parameter data transfer

9.1 Addressing of the parameter data

The parameter data are addressed via codes which are listed in the form of a code table in this communication manual and in the corresponding documentation for your controller.

Parameter reference (📖 63)

Addressing of the Lenze parameters

With the INTERBUS parameter data channel, the parameters of a device are not directly addressed via the Lenze codes but via an index and subindex.

The conversion is made via an offset (24575 or 0x5FFF):

- INTERBUS index (dec) = 24575 - Lenze code number (dec)
- INTERBUS index (hex) = 0x5FFF - Lenze code number (hex)

Example of C00105 (deceleration time quick stop (QSP))

- INTERBUS index (dec) = 24575 - 105 = 24470
- INTERBUS index (hex) = 0x5FFF - 0x69 = 0x5F96

9.2 Initialising PCP communication

Make entries into the CRL (communication relation list) so that communication between the INTERBUS master and the communication module can take place.

Make the following entries in the CRL of the INTERBUS master:

Field name	Entry
Communication reference	2
Connection type	Master slave acyclic
Connection attribute	Defined
Max-PDU Sending-High-Prio	0
Max-PDU Sending-Low-Prio	64
Max-PDU Receiving-High-Prio	0
Max-PDU Receiving-Low-Prio	64
Supported Services Request	0x803000
Supported Services Response	0x000000
Maximum SCC	1
Maximum RCC	1
Maximum SAC	1
Maximum RAC	1

9.3 Supported PMS services

Parameters are transferred via the PCP channel (PCP = Peripherals Communication Protocol). This is executed via PMS services.

In the following, only parameters and their contents are given that are returned by the Lenze controllers. All other transfer parameters of the given PMS services can be obtained from the corresponding descriptions of the INTERBUS master.

The following PMS services are supported by Lenze controllers:

- Initiate: Establish a connection from the INTERBUS master to the controller
- Abort: Abort connection
- Reject: Reject impermissible PMS service
- Read: Read parameters
- Write: Write parameters
- Get-OD: Read out the object directory
- Identify: Identification of the controller
- Status: Read the status of the controller

9.3.1 Initiate

The "Initiate" PMS service establishes a logic connection between the INTERBUS master and the communication module.

The controller provides the following parameters:

Designation	Value	Description
Profile number	0	No profile is supported.
Password	0	The password function of INTERBUS is not supported.
Access groups	0	There are no access groups.
Access-Protection Supported	TRUE	Access protection is supported.
Version OD	0	Version of the object directory

9.3.2 Abort

The "Abort" PMS service aborts a logic connection between the INTERBUS master and the communication module.

9.3.3 Reject

The "Reject" PMS service rejects a non-supported PMS service.

9.3.4 Read / Write

The "Read" PMS service reads parameters from the controller. The controller outputs the requested parameter or an error message.

The "Write" PMS service writes to parameters of the controller. The controller outputs a positive feedback or an error message.

The following error messages can occur:

Error Class	Error Code	Additional Code [hex]	Description
6	3	0x00	No access authorisation
6	5	0x10	Impermissible job parameter
6	5	0x11	Invalid subindex
6	5	0x12	Data length too big
6	5	0x13	Data length too small
6	6	0x00	Object is not a parameter
6	7	0x00	Object does not exist
6	8	0x00	Data types do not comply with each other
8	0	0x00	Job cannot be executed
8	0	0x20	Job currently cannot be executed
8	0	0x21	Cannot be executed because of local control
8	0	0x22	Cannot be executed because of device state
8	0	0x30	Quit value range / parameter can only be changed when the controller is inhibited.
8	0	0x31	Value of the parameter is too high
8	0	0x32	Value of the parameter is too low
8	0	0x33	Subparameter outside the value range
8	0	0x34	Value of the subparameter is too high
8	0	0x35	Value of the subparameter is too low
8	0	0x36	Maximum value is lower than minimum value
8	0	0x41	Communication object cannot be displayed on process data
8	0	0x42	Length of the process data exceeded
8	0	0x43	General collision with other values

9.3.5 Get-OD

The "Get-OD" PMS service reads out the object description for every parameter and data type.

9.3.6 Identify

The "Identify" PMS service provides information on how to identify the controller.

The controller with the plugged-on communication module provides the following parameters for this:

Parameter	Type	Description
Device manufacturer	Visible string	Company name "Lenze"
Device type	Visible-string (15 characters)	Device name for the controller and the communication module
Device version	Visible-string (15 characters)	Firmware version of the controller and the communication module

Example: Visible string "device type" (15 characters)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
E	8	4		A	F	G	H	C		A	F	Y	I	B

Character	Description
1 ... 3	Product range of the controller and the communication module (E84 = 8400 series)
4	Blank
5 ... 6	A: Version of the controller F: Firmware of the controller
7 ... 9	G: Device / controller SC: StateLine C HC: HighLine C
10	Blank
11 ... 12	A: Version of the communication module F: Firmware of the communication module
13 ... 15	Y: Communication module IB: INTERBUS

Structure of the visible string "device version" (15 characters)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4	.	0	0		0	1	.	0	0				

Character	Description
1 ... 5	Firmware version of the controller (from C00099/0, e.g. "04.00.xx.yy", without internal revision status ["xx"] and build status ["yy"])
6	Blank
7 ... 11	Firmware version of the communication module (from C13902, e.g. "01.00.xx.yy", without internal revision status ["xx"] and build status ["yy"])
12 ... 15	Blank

9.3.7 Status

The "Status" PMS service provides status information about the controller.

The controller provides the following values:

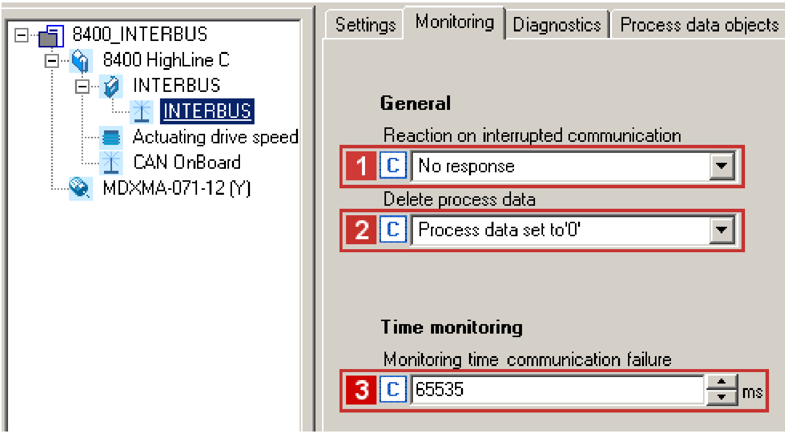
Status	Value	Description
Logical status	0 = ready for communication	Information about the current operating mode of the controller with regard to communication
Physical status	• 0 = ready for operation (device state "OPERATION ENABLED") • 1 = ready for operation to a limited extent (all other device states)	Information about the current operating status of the controller.
Local Detail	0	Is not supported.

10 Monitoring

Communication fault

- If INTERBUS communication is interrupted, e.g. due to cable break or INTERBUS master failure, no process data are transmitted to the slave in the "IBS-ACTIVE" (57) status.
- After the time to be parameterised by the user in C13881 has elapsed, the response parameterised in C13880/1 is executed in the controller.

Settings in the »Engineer«



Under the **Settings** tab in the »Engineer« you can set the following parameters:

- 1 **Reaction on interrupted INTERBUS communication** (C13880/1)
- 2 **Delete process data** (C13885)
  – Setting of the process data which the controller is to process further for maintaining internal communication when the INTERBUS has failed.
- 3 **Monitoring time for INTERBUS communication failure** (C13881)

11 Diagnostics

For fault diagnostics, the INTERBUS module is provided with the LEDs on the front. Furthermore code C13861 can be used to query the current bus status (for this, see Diagnostics with the »Engineer« (📖 57)).

11.1 LED status displays

Note!

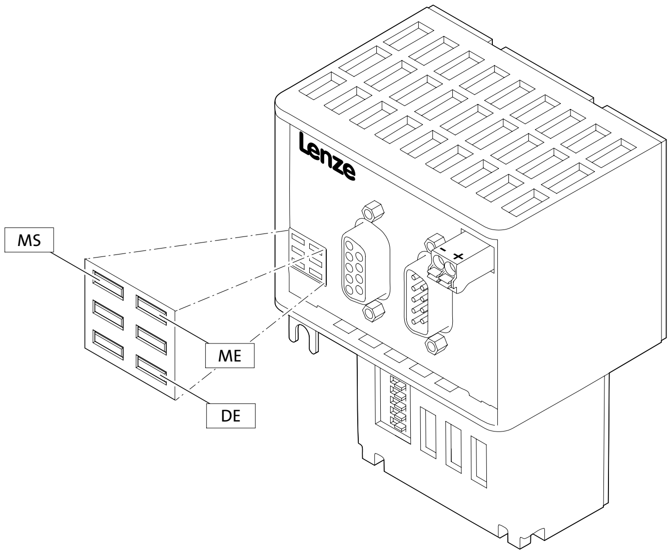
In normal operation only the LEDs **MS** (📖 55) and **BS** (📖 56) should be lit constantly.

The following status displays are distinguished:

- Module status displays (📖 55)
- Fieldbus status displays (📖 56)

11.1.1 Module status displays

The LEDs **MS**, **ME** and **DE** display the module status.

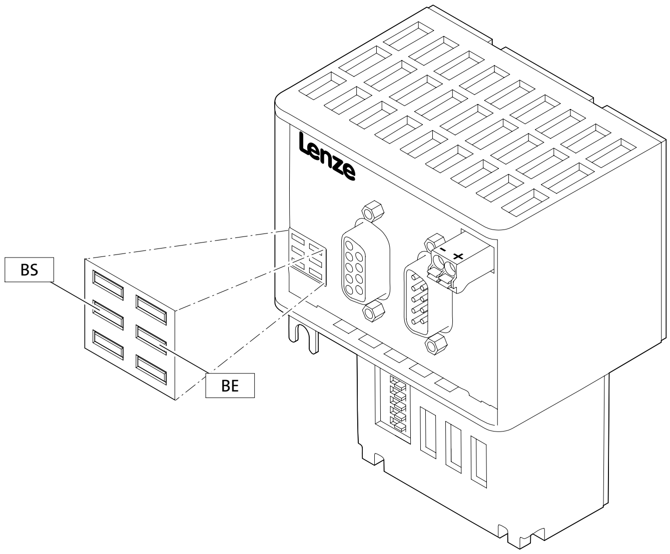


E84YCIB006

LED	Colour	Status	Description
MS	Green	On	 The communication module is supplied with voltage and has established a connection to the standard device.
		Blinking	 The communication module is supplied with voltage, but has not established a connection to the standard device. (Standard device is switched off, initialising or not present.)
ME	Red	Blinking	 An error in the communication module has occurred.
DE	Red	On	 The communication module is not accepted by the standard device (see notes in the documentation for the standard device), or the standard device is not active.

11.1.2 Fieldbus status displays

The LEDs **BS** and **BE** display the fieldbus status.



E84YCIB006

LED	Colour	Status	Description
BS	Green	Off	The communication module is not active on the fieldbus. Data cycles are not executed.
		Blinking	 Communication has been established via the communication module. The INTERBUS is active. Data cycles are executed.
BE	Red	On	 INTERBUS communication is interrupted. Data cycles are not executed.
		Blinking	 Impermissible setting: • Data word sum (PD + PCP) > 10 • Data word sum (PD + PCP) = 0 The communication module has been initialised and continues to operate internally with the following values: • PD = 2 (words) • PCP = 1 (word)

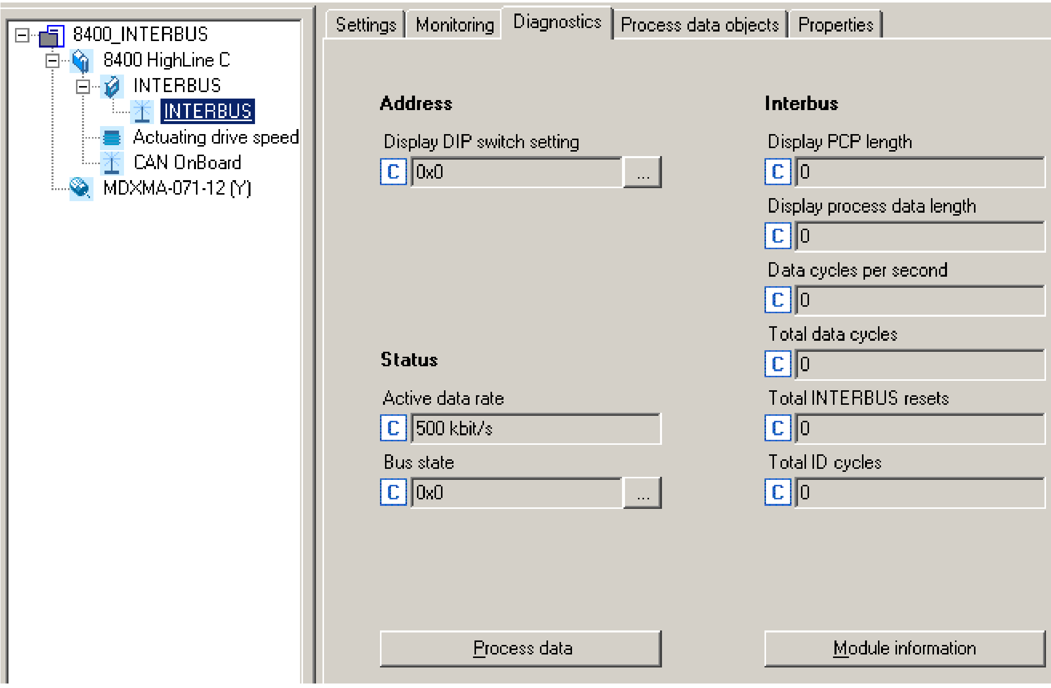
11.2 Querying the current bus status

Code C13861 displays the current bus status of the INTERBUS node:

Value of C13861 [hex]	Bus status	Description
0xyyy0	IBS-INIT	Initialisation
0xyyy1	IBS-ACTIVE	The bus is active. Data cycles are executed.
0xyyy2	IBS-READY	The bus is ready for operation. No data cycles are executed.
yyy = device-internal use		

Diagnostics with the »Engineer«

In the »Engineer« the **Diagnostics** tab serves to display various pieces of diagnostic INTERBUS information.

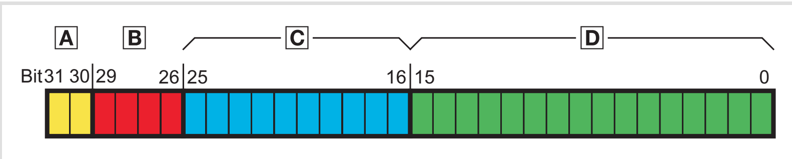


11.3 Diagnostic data

- Pending diagnostic data are signalised by the master via an alarm message to the slave.
- Errors and warnings in the standard device and the plugged-in module are transmitted to the master as extended diagnostic messages.
- The diagnostic data can be viewed via the hexadecimal representation of the Engineering tool.

Bytes	Meaning	Value [hex]
1 ... 6	Diagnostics block header	0x0010 001C 0100
7 ... 8	Alarm type	0x0001 (Diagnosis)
9 ... 12	API	0x0000 0000
13, 14	Slot number	0x0001 / 0x0002
15, 16	Subslot number	0x0001
17 ... 20	Module ID number	ID according to module
21 ... 24	Submodule number	ID according to module
25, 26	Alarm specifier	0xB000
27, 28	User structure identifier	0x0001
29 ... 32	Error code of the Inverter Drive 8400	

Error code of the Inverter Drive 8400



A Reserved
B Error type
C Error subject area
D Error ID

[11-1] Structure of the error number

- Bytes 29 ... 32 of the diagnostic message contain the error code of the Inverter Drive 8400.
- The logbook and standard device code **C00165** show the error number in the following syntax to improve the legibility:
  [error type].[error subject area no.].[error ID]

Software manual/»Engineer« online help for the Inverter Drive 8400

Here you'll find detailed information on the structure and the contents of the error codes.

12 Error messages

This chapter provides the error messages of the communication module as a supplement to the error list in the software manual and the »Engineer« online help for the Inverter Drive 8400.

Software manual/»Engineer« online help for the Inverter Drive 8400

Here you'll find general information on diagnostics & fault analysis and error messages.

12.1 Short overview (A-Z) of the INTERBUS error messages

The following table contains all error messages of the communication module in alphabetical order with the preset error response and, if available, the parameter for setting the error response.

Tip!

When you click the cross-reference in the last column, you will get to the detailed description (causes and remedies) of the corresponding error message.

Error text	Error type	Subject area no.	Error no.	Adjustable in	Detailed information
Communication stack reset	Error	444	33062	-	0x01bc8126
Data exchange stopped	No response	444	33073	C13880/1	0x01bc8131
Internal error	Error	444	24592	-	0x01bc6010
Internal error	Error	444	24593	-	0x01bc6011
Internal error	Error	444	24832	-	0x01bc6100
Internal error	Error	444	24833	-	0x01bc6101
Invalid configuration	Warning	444	33061	-	0x01bc8125
Invalid initialisation	Warning	444	33063	-	0x01bc8127
Invalid module configuration	Error	444	25648	-	0x01bc6430
Invalid parameter set	Error	444	25631	-	0x01bc641f
Lenze setting loaded	Error	444	25632	-	0x01bc6420
Lost connection to 8400 base device	Error	444	12544	-	0x01bc3100
Memory not accessible	Error	444	21809	-	0x01bc5531
Memory read error	Error	444	21810	-	0x01bc5532
Memory write error	Error	444	21811	-	0x01bc5533

12.2 Possible causes and remedies

This chapter includes a list of all error messages of the communication module in numerically ascending order of the error number. Possible causes and remedies as well as responses to error messages are described in detail.

Tip!

A list of all error messages of the communication module in alphabetical order is provided in the previous chapter "Short overview (A-Z) of the INTERBUS error messages" (📖 59).

Lost connection to 8400 base device [0x01bc3100]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Network cable (plug) is defective. Network cable at INTERBUS terminal X206 (IN) or X207 (OUT) is disconnected.	Check network cable (plug) and replace it, if required. Plug in network cable at INTERBUS terminal X206 (IN) or X207 (OUT).

Memory not accessible [0x01bc5531]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Memory could not be accessed.	Send module with error description to Lenze.

Memory read error [0x01bc5532]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameter in the memory module could not be read.	Repeat download of the application (including module).

Memory write error [0x01bc5533]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameter in the memory module could not be written.	Repeat download of the application (including module).

Internal error [0x01bc6010]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Internal error [0x01bc6011]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Internal error [0x01bc6100]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Internal error [0x01bc6101]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

Invalid parameter set [0x01bc641f]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Loading of an active parameter set was not possible.	Repeat download of the application (including module).

Lenze setting loaded [0x01bc6420]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access via standard device to the parameter set in the memory module failed.	Repeat download of the application (including module).

Invalid module configuration [0x01bc6430]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module configuration is faulty.	Check and correct module configuration.

Invalid configuration [0x01bc8125]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The active configuration is invalid: • Data word sum (PD + PCP) > 10 words • Data word sum (PD + PCP) = 0	Adapt the number of PD and PCP: • Data word sum (PD + PCP) = 1 ... 10 words

Communication stack reset [0x01bc8126]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Error during processing the INTERBUS services	Execute reinitialisation by the master.

Invalid initialisation [0x01bc8127]

Response (Lenze setting in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The INTERBUS initialisation has failed.	Execute reinitialisation by the master.

Data exchange stopped [0x01bc8131]

Response (Lenze setting in bold)	**Setting:** C13880/1
☒ **None** ☐ System fault ☒ Fault ☐ Trouble ☒ Quick stop by trouble ☒ Warning locked ☐ Warning ☒ Information	

Cause	Remedy
The data exchange at the INTERBUS has been terminated.	• Check cables and connections. • Execute reinitialisation by the master.

13 Parameter reference

This chapter supplements the parameter list and the table of attributes in the software manual and the »Engineer« online help for the Inverter Drive 8400 by parameters of the E84AYCIB communication module (INTERBUS).

Software manual/»Engineer« online help for the Inverter Drive 8400
Here you'll find general information about parameters.

13.1 Parameters of the communication module

This chapter lists the parameters of the E84AYCIB communication module (INTERBUS) in numerically ascending order.

C13850

Parameter | Name: **C13850 | All words to master**			Data type: UNSIGNED_16 Index: 10725_d = $29E5_h$
Display of the process data words which are transmitted from the communication module to the master. • Subcodes 1 ... 16 display all process data words to the master. Only those which are configured are valid. Maximally the first 10 words are relevant to the E84AYCIB INTERBUS module.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Information**
C13850/1			
...			
C13850/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master**			Data type: UNSIGNED_16 Index: 10724_d = $29E4_h$
Display of the process data words which are transmitted from the master to the communication module. • Subcodes 1 ... 16 display all process data words from the master. Only those which are configured are valid. Maximally the first 10 words are relevant to the E84AYCIB INTERBUS module.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Information**
C13851/1			
...			
C13851/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to standard device**			Data type: UNSIGNED_16 Index: 10723_d = $29E3_h$
Display of the process data words which are transmitted from the communication module to the standard device. • Subcodes 1 ... 16 display all process data words to the standard device. Maximally the first 10 words are relevant to the E84AYCIB INTERBUS module.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Information**
C13852/1			
...			
C13852/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words from standard device**			Data type: UNSIGNED_16 Index: 10722_d = $29E2_h$
Display of the process data words which are transmitted from the standard device to the communication module. • Subcodes 1 ... 16 display all process data words from the standard device. Maximally the first 10 words are relevant to the E84AYCIB INTERBUS module.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Information**
C13853/1			
...			
C13853/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13860

Parameter | Name: **C13860 | Active settings**			Data type: UNSIGNED_8 Index: 10715_d = $29DB_h$
Display of the set number for • Parameter data words (PCP) • Process data words (PD) to be used (16 bits/word)			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Information**
C13860/1			Display of PCP data length
C13860/2			Display of process data length
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Bus status**		Data type: BITFIELD_16 Index: 10714_d = $29DA_h$

Display of the current bus status

▸ Querying the current bus status (📖 57)

Meaning of the resulting hexadecimal values:
- 0xyyy0 = IBS-INIT
- 0xyyy1 = IBS-ACTIVE
- 0xyyy2 = IBS-READY

Value is bit coded:		Information
Bit 0	Bit 0	
...	...	
Bit 15	Bit 15	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13862

Parameter | Name: **C13862 | Counter**			Data type: UNSIGNED_16 Index: 10713_d = $29D9_h$

Counter for cycles and INTERBUS resets

Display area (min. value | unit | max. value)			
0		65535	

Subcodes	Information
C13862/1	Data cycles per second
C13862/2	Total data cycles
C13862/3	Number of INTERBUS resets
C13862/4	Number of ID cycles

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13863

Parameter | Name: **C13863 | Active baud rate**		Data type: UNSIGNED_8 Index: 10712_d = $29D8_h$

Display of the set baud rate

Selection list (read only)		
0	500 kbps	
1	2.00 Mbps	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13880

Parameter | Name: **C13880 | Reaction on communication failure**		Data type: UNSIGNED_8 Index: 10695_d = $29C7_h$
Monitoring response in case of a communication fault A change in the monitoring response becomes effective immediately. ▸ Communication fault (📖 53)		
Selection list		
0	No response	
1	Error	
3	Quick stop by trouble	
4	Warning locked	
6	Information	
Subcodes	**Lenze setting**	**Information**
C13880/1	0: No response	Response to interrupted INTERBUS communication
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13881

Parameter | Name: **C13881 | Monitoring time for Interbus communication failure**			Data type: UNSIGNED_16 Index: 10694_d = $29C6_h$
If the "IBS-ACTIVE" status (INTERBUS is active and data are cyclically exchanged) is quit, the response parameterised in C13880/1 is executed after the monitoring time set for the data exchange has elapsed. • The value "65535" in this code deactivates monitoring. • A change in monitoring is effective immediately. ▸ Communication fault (📖 53)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**65535 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Delete process data**		Data type: UNSIGNED_8 Index: 10690_d = $29C2_h$
Setting of the process data which the controller is to process further for maintaining internal communication when the INTERBUS has failed.		
Selection list (Lenze setting printed in bold)		
0	Use of the last master process data	
1	**Process data are set to the value '0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13892

Parameter | Name: **C13892 | PCP - length**	Data type: UNSIGNED_8 Index: 10683_d = $29BB_h$

Setting of the number of the parameter data words to be used (PCP)
The setting is activated ...
- if all DIP switches (S205) are in the "OFF" position;
- after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards.

▸ Setting the number of parameter data words (PCP) (📖 36)

Selection list (Lenze setting printed in bold)	
0	No PCB data
1	**1 word**
2	2 words
4	4 words

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13893

Parameter | Name: **C13893 | Process data length**	Data type: UNSIGNED_8 Index: 10682_d = $29BA_h$

Setting of the number of the process data words to be used (PD)
The setting is activated ...
- if all DIP switches (S205) are in the "OFF" position;
- after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards.

▸ Setting the number of process data words (PD) (📖 35)

Selection list (Lenze setting printed in bold)	
0	No process data
1	1 word
2	**2 words**
3	3 words
4	4 words
5	5 words
6	6 words
7	7 words
8	8 words
9	9 words
10	10 words

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13894

Parameter | Name: **C13894 | Baud rate**	Data type: UNSIGNED_8 Index: 10681_d = $29B9_h$

Setting of the baud rate
The setting is activated ...
- if all DIP switches (S205) are in the "OFF" position;
- after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards.

▸ Setting the baud rate (📖 37)

Selection list (Lenze setting printed in bold)	
0	**500 kbps**
1	2.00 Mbps

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13900

Parameter | Name: **C13900 | Firmware product type**	Data type: VISIBLE_STRING Index: 10675_d = $29B3_h$

Display of the product type (string with a length of 8 bytes)
The following identification code is output: "E94AFYIB".

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13901

Parameter | Name: **C13901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 10674_d = $29B2_h$

Display of the compilation date of the firmware (string with a length of 20 bytes)
The date ("MMM TT JJJJ") and time ("hh:mm:ss") are output, e.g. "Mar 21 2005 12:31:21".

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13902

Parameter | Name: **C13902 | Firmware version**	Data type: VISIBLE_STRING Index: 10673_d = $29B1_h$

Display of the firmware version (string with a length of 11 bytes)
The identification code is output, e.g. "01.00.00.00".

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13920

Parameter | Name: **C13920 | Display DIP switch setting**		Data type: BITFIELD_8 Index: 10655_d = $299F_h$
Display of the current DIP switch position • The set PCP data length is displayed in C13860/1. • The set process data length is displayed in C13860/2. • The set baud rate is displayed in C13863. ▸ Possible settings through DIP switch (🕮 34)		
Value is bit coded:		**Information**
Bit 0	DIP 8	Baud rate
Bit 1	DIP 7	No function
Bit 2	DIP 6	Number of parameter data words (PCP)
Bit 3	DIP 5	
Bit 4	DIP 4	Number of process data words (PD)
Bit 5	DIP 3	
Bit 6	DIP 2	
Bit 7	DIP 1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

13.2 Table of attributes

The table of attributes contains information required for communicating with the controller via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index by which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (one parameter element only)
			A	Array variable (multiple parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte with sign
			INTEGER_16	2 bytes, with sign
			INTEGER_32	4 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes, without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ no decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible when the controller is inhibited	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13850	All words to master	10725	29E5	A	16	UNSIGNED_16	1	☑		
C13851	All words from master	10724	29E4	A	16	UNSIGNED_16	1	☑		
C13852	All words to standard device	10723	29E3	A	16	UNSIGNED_16	1	☑		
C13853	All words from standard device	10722	29E2	A	16	UNSIGNED_16	1	☑		
C13860	Active settings	10715	29DB	A	2	UNSIGNED_8	1	☑		
C13861	Bus status	10714	29DA	E	1	BITFIELD_16		☑		
C13862	Counter	10713	29D9	A	4	UNSIGNED_16	1	☑		
C13863	Active baud rate	10712	29D8	E	1	UNSIGNED_8	1	☑		
C13880	Reaction on communication failure	10695	29C7	A	1	UNSIGNED_8	1	☑	☑	
C13881	Monitoring time for Interbus communication failure	10694	29C6	E	1	UNSIGNED_16	1	☑	☑	
C13885	Delete process data	10690	29C2	E	1	UNSIGNED_8	1	☑	☑	
C13892	PCP - length	10683	29BB	E	1	UNSIGNED_8	1	☑	☑	
C13893	Process data length	10682	29BA	E	1	UNSIGNED_8	1	☑	☑	
C13894	Baud rate	10681	29B9	E	1	UNSIGNED_8	1	☑	☑	
C13900	Firmware product type	10675	29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware compilation date	10674	29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware version	10673	29B1	E	1	VISIBLE_STRING		☑		
C13920	Display DIP switch setting	10655	299F	E	1	BITFIELD_8		☑		

14 Index

Lenze Drives GmbH
Postfach 10 13 52
D-31763 Hameln
Germany

+49 (0)51 54 / 82-0

+49 (0)51 54 / 82-28 00

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
32699 Extertal
Germany

00 80 00 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS84AYCIB ▪ 13362260 ▪ EN 4.0 ▪ TD17

10 9 8 7 6 5 4 3 2 1

Accessories

PROFIBUS®

E84AYCPM

Inverter Drives 8400 ________________________

Communication Manual EN

13422193

Lenze

1 About this documentation

This documentation exclusively describes the E84AYCPM communication module (PROFIBUS).

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **"Inverter Drives 8400" hardware manual**.

The hardware manual contains safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Typical applications are explained with the help of examples.

The theoretical connections are only explained in so far as they are necessary for comprehending the function of the communication module.

This documentation does not describe the software of other manufacturers. No responsibility is taken for corresponding information given in this documentation. Information on how to use the software can be obtained from the documents of the control system (master).

All brand names used in this documentation are trademarks of their respective owners.

Tip!

Detailed information about PROFIBUS can be found on the website of the PROFIBUS & PROFINET user organisation:

www.profibus.com

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.

Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.Lenze.com

Validity information

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
PROFIBUS communication module	E84AYCPM	VA	01.00

Screenshots/application examples

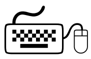
All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the Engineering tools installed (»Engineer«, »STEP7«), the screenshots in this documentation may differ from the actual screen display.

1.1 Document history

Version			Description
1.0	11/2007	TD17	First edition
2.0	11/2008	TD17	General revision
3.0	02/2010	TD17	• Update of chapter structure • General revision
4.0	11/2010	TD17	General revision
5.0	11/2011	TD17	• New layout • New: Going online with »Engineer« via TCI (📖 35) • General corrections

1.2 Conventions used

This manual uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Text colour blue	All pieces of information that only apply to or from a specific software version of the inverter are highlighted accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »Engineer«...
Window	*italics*	The *message window*... / The *Options* dialog box ...
Variable name		Setting *bEnable* to TRUE...
Control element	**Bold**	The **OK** button ... / The **Copy** command ... / The Properties tab ... / The **Name** input field ...
Sequence of menu commands		If several successive commands are required for executing a function, the individual commands are separated from each other by an arrow: Select the command **File** ➜ **Open** to...
Hyperlink	underlined	Optically highlighted reference to another topic. Can be activated with a mouse-click in this online documentation.
Symbols		
Page reference	(📖 6)	Optically highlighted reference to another page. Can be activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Inverter	Lenze inverter of the "Inverter Drives 8400" product series
Standard device	
Code	Parameters which serve to parameterise or monitor the inverter. This term is usually called "index".
Subcode	If a code contains several parameters, these are stored in subcodes. This manual uses a slash "/" as a separator between code and subcode (e.g. "C118/3"). This term is usually called "subindex".
GSD / GSE	Device data base file (device description for PROFIBUS stations)
HW	Hardware
Lenze setting	Settings with which the device is preconfigured ex works.
Basic setting	
	PROFIBUS® (Process Field Bus) is a widely-used fieldbus system for the automation of machines and production plants. PROFIBUS® is a registered trademark and patented technology licensed by the PROFIBUS & PROFINET International (PI) user organisation.
PDO	Process data object
PLC	Programmable Logic Controller (German designation: SPS - Speicherprogrammierbare Steuerung)
»STEP7«	Siemens software for programming and configuring PROFIBUS Siemens control systems
SW	Software
TCI	Tool Calling Interface

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Structure of the safety instructions:

Danger!

(characterises the type and severity of danger)

Note

(describes the danger and informs how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to other documents

2 Safety instructions

Note!

Always observe the specified safety measures to avoid severe injury to persons and damage to property!

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application instructions

Danger!

If you disregard the following basic safety measures, this can cause severe injury to persons and damage to material assets.

Lenze drive and automation components ...

- must only be used as directed.
  ▶ Application as directed (📖 13)
- must never be commissioned in the event of visible damage.
- must never be technically modified.
- must never be commissioned before they have been completely mounted.
- must never be operated without the covers required.
- can - depending on their degree of protection - have live, moving or rotating parts during and after operation. Surfaces can be hot.

For Lenze drive components ...

- use only the accessories approved.
- use only original spare parts from the manufacturer.

Observe all specifications given in the attached and associated documentation.

- This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
  ▶ Features (📖 14)
- The procedural notes and circuit details described in this document are only proposals. It is up to the user to check whether they can be adapted to the particular applications. Lenze does not take any responsibility for the suitability of the procedures and circuit proposals described.

Only qualified personnel may work with and on Lenze drive and automation components. According to IEC 60364 and CENELEC HD 384, these are persons ...

- who are familiar with the installation, assembly, commissioning and operation of the product.
- who have the corresponding qualifications for their work.
- who know all regulations for the prevention of accidents, directives and laws applicable on site and are able to apply them.

2.2 Device- and application-specific safety instructions

- During operation, the communication module must be firmly connected to the standard device.
- Only use cables corresponding to the given specifications.
  ▶ Bus cable specification (📖 28)

Documentation for the standard device, control system, plant/machine

All other measures prescribed in this documentation must also be implemented.
Observe the safety instructions and application notes specified in the documentation.

2.3 Residual hazards

Protection of persons

If the Inverter Drives 8400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not ensured without implementing external measures.

▶ Protective insulation (📖 16)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.

▶ Installation (📖 21)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module for use in conjunction with the following Lenze standard devices:

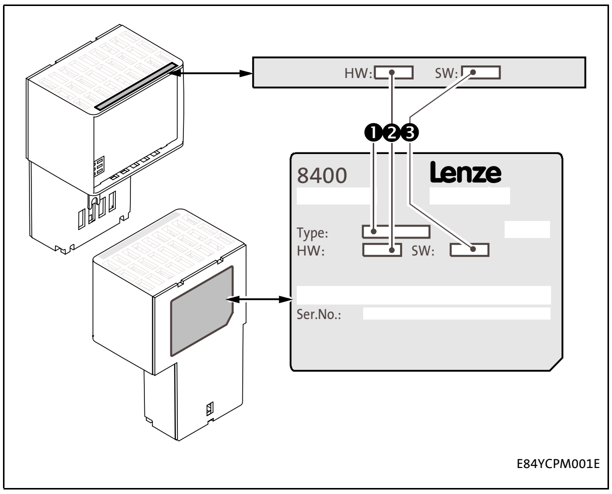
Product series	Type designation	From software version
Inverter Drives 8400 StateLine	E84AVSCxxxxx	01.00
Inverter Drives 8400 HighLine	E84AVHCxxxxx	01.00
Inverter Drives 8400 TopLine	E84AVTCxxxxx	01.00

- is a device intended for use in industrial power systems.
- is only to be operated under the operating conditions specified in this documentation.
- may only be used in PROFIBUS networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation as well as the hardware and software version of the communication module are indicated on the nameplate:



1 Type designation (type)

E84	Product series
A	Version
Y	Module identification: extension module
C	Module type: communication module
PM	PROFIBUS
V/S	V: coated version
	S: standard version

2 Hardware version (HW)

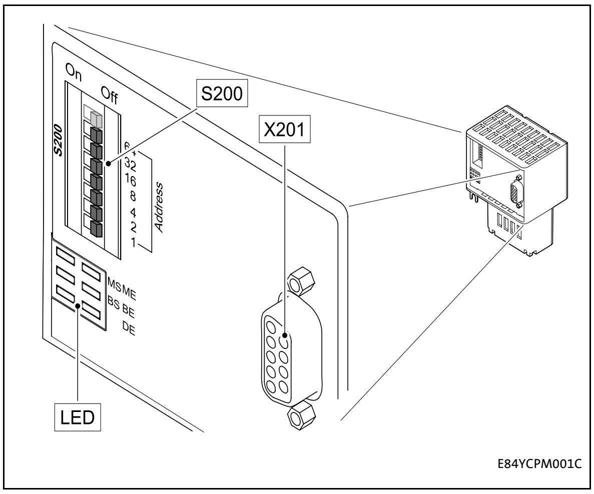
3 Software version (SW)

[3-1] Identification data

3.3 Features

- Interface module for the PROFIBUS communication system for connection to the expansion slots of the Inverter Drives 8400
- Support of parameter data channels DRIVECOM (DP-V0) and PROFIDrive (DP-V1)
- A maximum of 16 process data words per direction can be exchanged.
- The communication module is supplied with voltage via the standard device.
- Bus coupling via remote bus according to the RS485 standard
- Automatic detection of the baud rate (9.6 kbps to 12 Mbps)
- Setting of the station address is possible via DIP switch or code.
- Access to all Lenze parameters

3.4 Terminals and interfaces



S200	DIP switches for setting the station address ▸ Setting the station address (📖 32)
X201	PROFIBUS connection • 9-pin Sub-D socket ▸ Network topology (📖 25) ▸ PROFIBUS connection (📖 29)
MS **ME** **BS** **BE** **DE**	5 LED status displays for diagnostics ▸ Module status displays (📖 85) ▸ Fieldbus status displays (📖 86)

[3-2] E84AYCPM communication module (PROFIBUS)

4 Technical data

"Inverter Drives 8400" hardware manual

Here you can find the **ambient conditions** and information on the **electromagnetic compatibility (EMC)** which also apply to the communication module.

4.1 General data and operating conditions

Area	Values
Order designation	• E84AYCPMV (coated version) • E84AYCPMS (standard version)
Communication profile	• PROFIBUS DP-V0 (DRIVECOM) • PROFIBUS DP-V1 (PROFIdrive)
Communication medium	RS485
Interface	9-pin Sub-D socket
Network topology	• Line (without repeater) • Tree/line (with repeater)
Bus device type	PROFIBUS slave
Number of slaves	• Max. 31 (without repeater) • Max. 125 (with repeater)
Max. cable length	1200 m (depending on the selected baud rate and the cable type used)
PNO identification number	0x0A89
Baud rate for cable type A (EN 50170)	9.6 kbps ... 12 Mbps (automatic detection)
Conformities, approvals	• CE • UL

4.2 Protective insulation

Danger!

Dangerous voltage

If the Inverter Drives 8400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not ensured without implementing external measures.

Possible consequences:

Death or severe injury

Protective measures:

If protection against accidental contact is required for the control terminals of the inverter and the terminals of the plugged-in device modules, ...

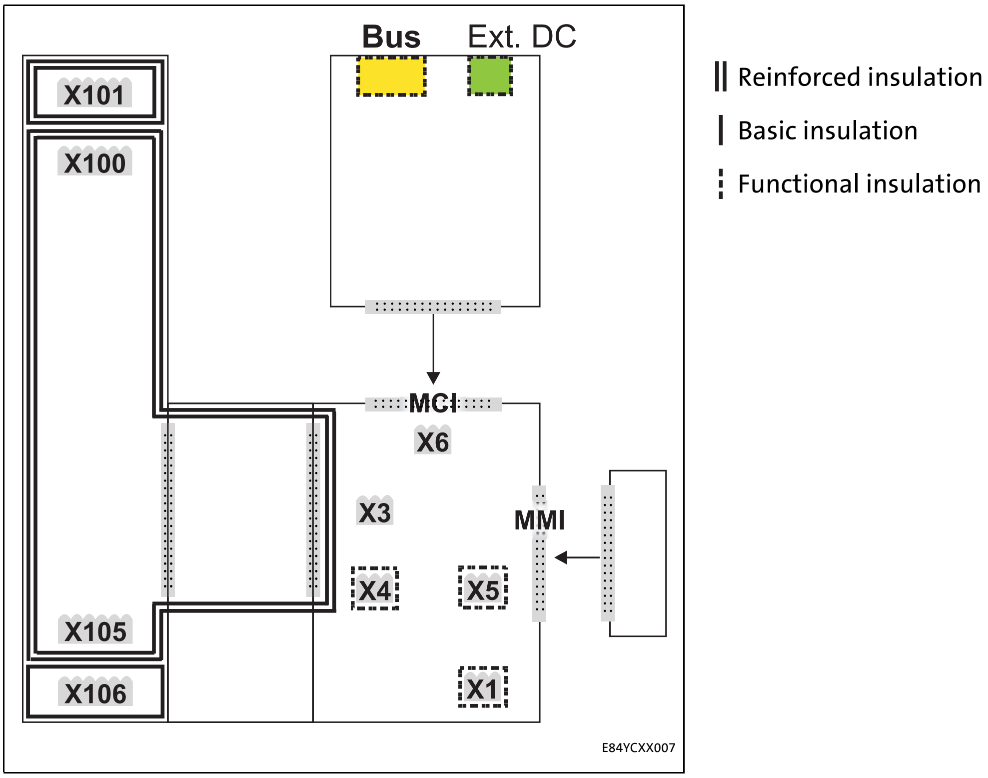
- a double isolating distance must be provided.
- the components to be connected must be provided with a second isolating distance.

Note!

The existing protective insulation in the Inverter Drives 8400 is implemented according to EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Inverter Drives 8400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection
X100	Mains / DC bus connection
X101	Relay contact
X105	Motor/brake resistor
X106	Motor PTC
X1	System bus (CANopen)
X3	Analog inputs/outputs
X4	Digital outputs
X5	Digital inputs
X6	Diagnostics
MCI	Slot for communication module
MMI	Slot for memory module

Example

Which type of protective insulation is used between the bus terminal of the device module in the MCI slot and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the device module's bus terminal is "functionally insulated".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between the mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.3 Protocol data

Area	Values
Process data words (PCD)	1 ... 16 words (16 bits/word)
Cyclic parameter data channel (DP-V0)	4 words
Acyclic parameter data channel (DP-V1)	Max. 240 bytes
PROFIBUS user data length	1 ... 16 words process data channel + 4 words parameter data channel

4.4 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in a PROFIBUS network depend on ...

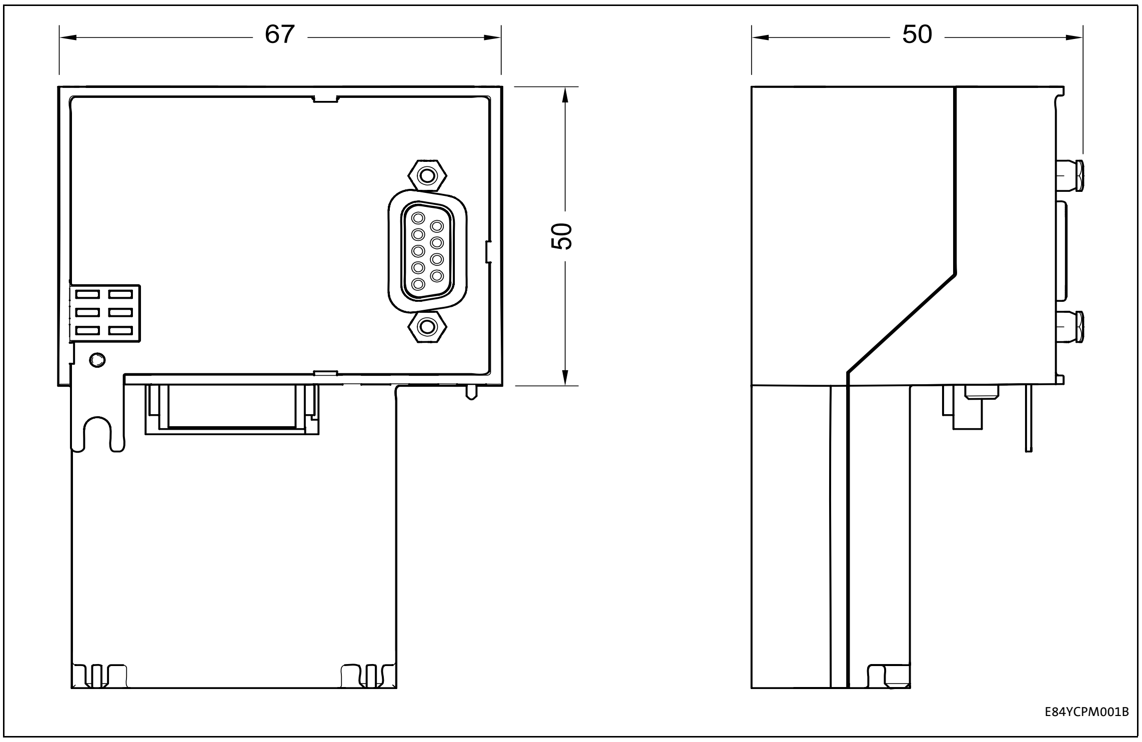
- the processing time in the inverter;
- the transmission delay time (baud rate / telegram length);
- the nesting depth of the network.

Processing time in the inverter

Data	Processing time	
Process data	Approx. 2 ms + 0 ... 1 ms + 1 ... x ms	update cycle processing time in the module application task runtime of the technology application used (tolerance)
Parameter data	Approx. 30 ms + 20 ms tolerance (typical) For some codes, the processing time may be longer (see software manual// »Engineer« online help for Inverter Drives 8400).	

There are no interdependencies between parameter data and process data.

4.5 Dimensions



[4-2] Dimensions

Dimensions in mm

5 Installation

Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:
- The communication module is defective.
- Communication via the fieldbus is not possible or faulty.

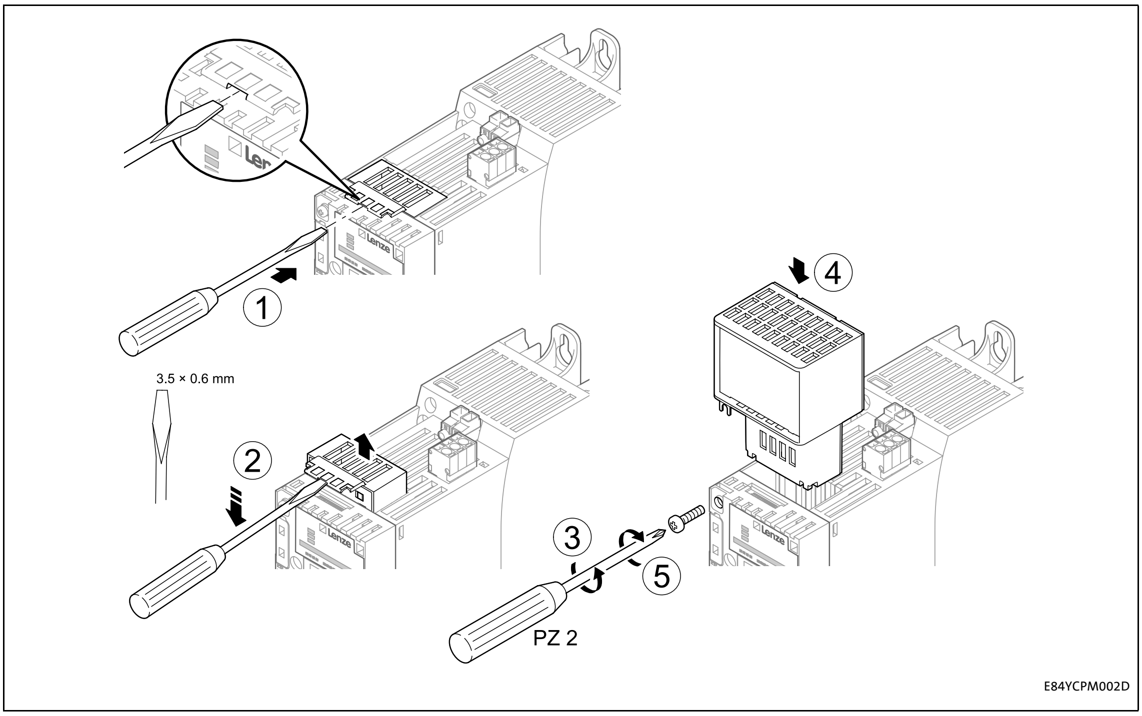
Protective measures

Discharge electrostatic charges before touching the module.

5.1 Mechanical installation

The communication module can be plugged into the MCI slot or unplugged while the inverter is switched on. When the module is plugged in, it is detected automatically, and a plausibility check regarding the function and version is carried out.

5.1.1 Mounting for 0.25 kW and 0.37 kW standard devices

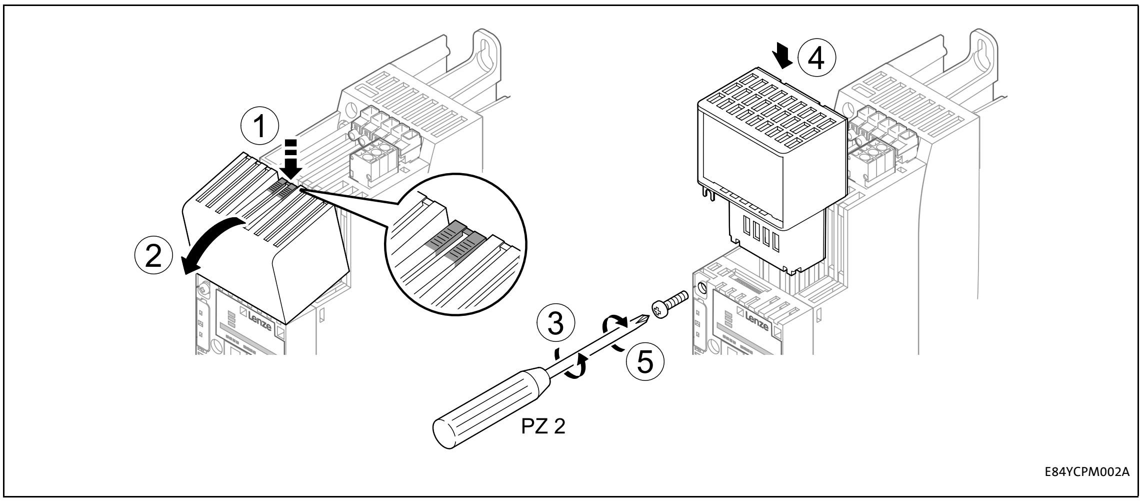


[5-1] Mounting for 0.25 kW and 0.37 kW standard devices

Mounting steps

1. Pry the cover out of the MCI slot using a screwdriver and remove it (1, 2).
2. Loosen the securing screw for the communication module on the standard device (3).
3. Insert the communication module into the MCI slot of the standard device (4).
4. Fasten the securing screw (5).

5.1.2 Mounting for standard devices of 0.55 kW and more

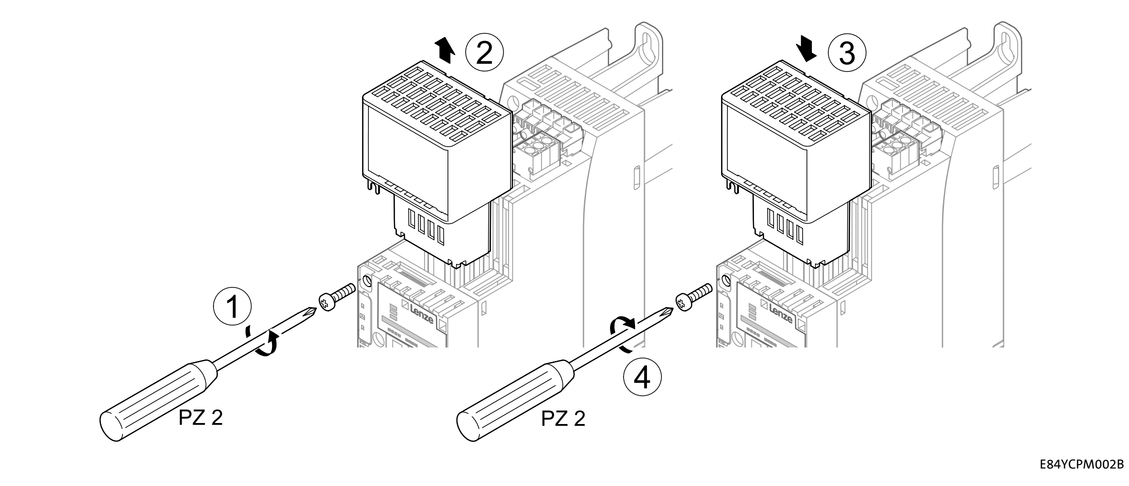


[5-2] Mounting for standard devices of 0.55 kW and more

Mounting steps

1. Slightly press on the area indicated in the illustration at the top of the cover for the standard device's MCI slot (1).
2. Tilt the cover forward and remove it from the standard device (2).
3. Loosen the securing screw for the communication module on the standard device (3).
4. Insert the communication module into the MCI slot of the standard device (4).
5. Fasten the securing screw (5).

5.1.3 Replacing the communication module



[5-3] Replacing the communication module

Mounting steps

1. Loosen the securing screw for the communication module on the standard device (1).
2. Remove the communication module from the MCI slot of the standard device (2).
3. Insert the new communication module into the MCI slot of the standard device (3).
4. Fasten the securing screw (4).

5.2 Electrical installation

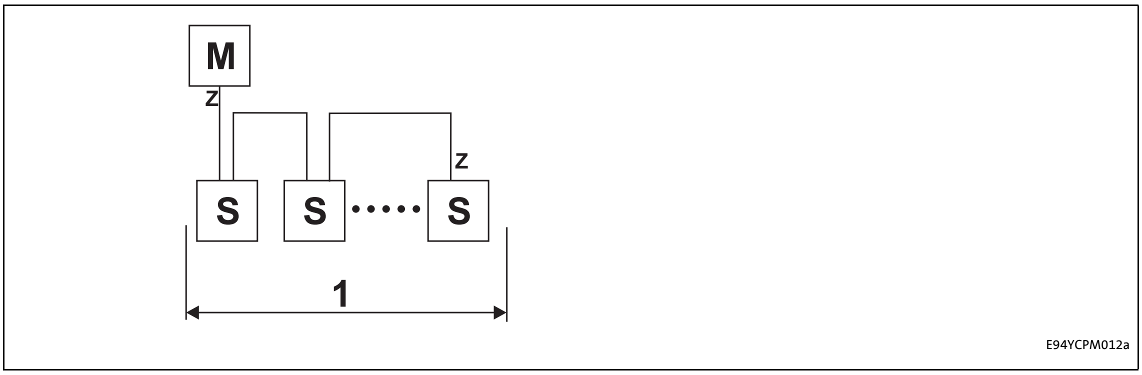
Documentation for the standard device, control system, plant/machine
Observe the notes and wiring instructions given in the documentation.

5.2.1 Network topology

The following examples show two simple RS485 networks.

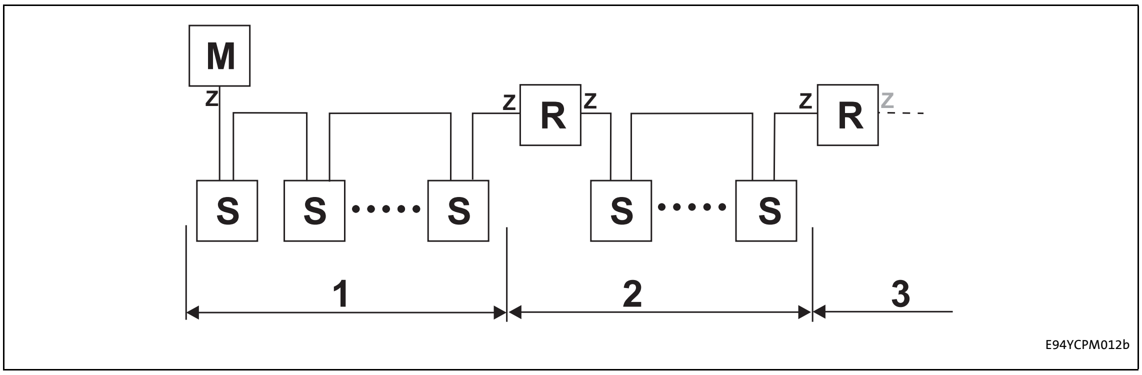
Every segment of the network must be terminated at both ends. The bus terminators of the PROFIBUS are marked with a "Z" in the below examples.

In the case of an RS485 network of only one segment, the PROFIBUS master (M) with the integrated bus terminator starts the segment while the bus terminating resistor in the connector of the last PROFIBUS station (S) must be activated.



[5-4] RS485 network with one segment

An RS485 network consisting of several segments contains repeaters (R) for coupling the segments. The repeaters are provided with integrated bus terminating resistors.



[5-5] RS485 network with a repeater

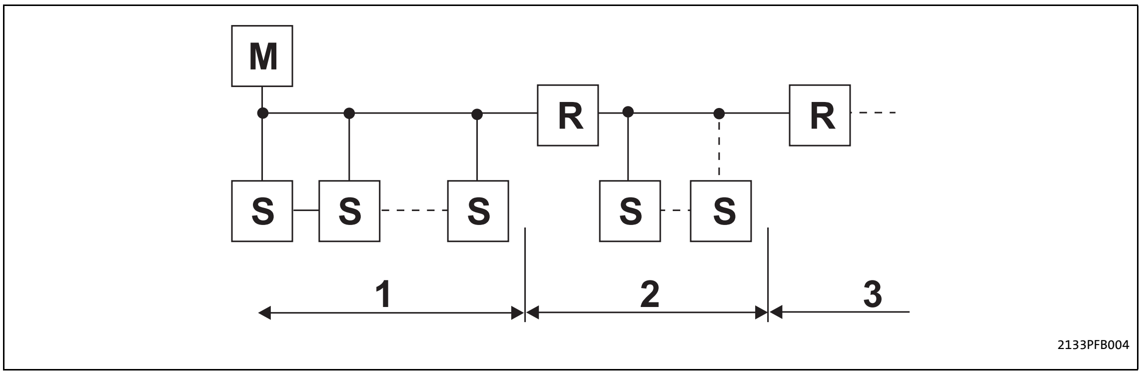
If no repeater is to be used at the end of the segment, the bus terminating resistor must be activated in the connector of the last device. The bus termination is supplied by the station itself.

Stop!

The bus terminator must always be supplied. Otherwise, the bus can get unstable.

▸ Activating the bus terminating resistor (📖 27)

Number of stations



[5-6] Number of stations

Segment	Master (M)	Slave (S)	Repeater (R)
1	1	31	-
	2	30	-
2	-	30	1
3	-	30	1

Tip!

Repeaters do not have a station address. When calculating the maximum number of stations, they reduce the number of stations by 1 on each side of the segment.

Repeaters can be used to build up line and tree topologies. The maximum total bus system expansion depends on ...

- the baud rate used;
- the number of repeaters used.

5.3 Activating the bus terminating resistor

The PROFIBUS must be terminated by a bus terminating resistor at the first and last physical bus station.

The bus terminating resistor in the bus connector of the bus cable is activated by means of a switch.

PROFIBUS cables with integrated bus terminating resistor are offered by several cable manufacturers.

Note!

If you want to disconnect individual bus stations, ensure that the bus terminators at the cable ends remain active.

Please observe that the bus termination is not active any longer if ...

- the bus connector has been disconnected;
- the voltage supply of the Inverter Drive 8400 has been switched off.

5.3.1 Bus cable specification

Note!

Only use cables which meet the listed specifications of the PROFIBUS user organisation.

Area	Values
Cable resistance	135 ... 165 Ω/km, (f = 3 ... 20 MHz)
Capacitance per unit length	≤ 30 nF/km
Loop resistance	< 110 Ω/km
Core diameter	> 0.64 mm
Core cross-section	> 0.34 mm^2
Cores	Twisted in pairs, insulated and shielded

Bus cable length

The length of the bus cable depends on the baud rate and cable type used. The data in the following table applies to PROFIBUS cables of "FC-Standard Cable" cable type .

Baud rate	Length
9.6 ... 93.75 kbps	1200 m
187.5 kbps	1000 m
500 kbps	400 m
1500 kbps	200 m
3000 ... 12000 kbps	100 m

Note!

The baud rate depending of the data volume, cycle time and number of stations should only be selected as high as required for the application.

Tip!

We recommend taking the use of optical fibres into consideration for high baud rates.

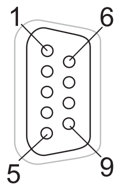
Advantages of optical fibres:

- External electromagnetic interferences have no effect on the transmission path.
- Bus lengths of several kilometres are also possible with higher baud rates.
- The bus length is ...
  - independent of the baud rate;
  - dependent on the optical fibre used.

5.3.2 PROFIBUS connection

The 9-pole Sub-D socket **X201** serves to connect the communication module to the bus system.

Assignment of the 9-pin Sub-D socket X201

View	Pin	Assignment	Description
	1	Not assigned	-
	2	Not assigned	-
	3	RxD/TxD-P	Data line B (received data/transmitted data, plus)
	4	RTS	Request To Send (received data/transmitted data, no differential signal)
	5	M5V2	Data ground (ground to 5 V)
	6	P5V2	5 V DC / 30 mA (bus termination)
	7	Not assigned	-
	8	RxD/TxD-N	Data line A (received data/transmitted data, minus)
	9	Not assigned	-

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The codes of the inverter and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

▶ Parameter reference (📖 95)

6.1 Before initial switch-on

Stop!

Before switching on the inverter for the first time, check ...

- the entire wiring for completeness, short circuit and earth fault.
- whether the bus system is terminated by means of a bus terminating resistor at the first and last physical bus station.
  ▶ Activating the bus terminating resistor (📖 27)

6.2 Configuration of the controller (master)

The controller (master) must be configured before communication with the communication module is possible.

Configuration for the controller (master) and the DP-V0 parameter data channel

For configuring the PROFIBUS you must read the device description file of the communication module into the master.

The device description file for the E84AYCPM communication module (PROFIBUS) can be found in the Download area at:

www.Lenze.com

The following language variants of the device description file can be used:

- LENZ0A89.GSD (source file, English)
- LENZ0A89.GSG (German)
- LENZ0A89.GSE (English)

Defining the user data length

The user data length is defined during the initialisation phase of the master.

The Inverter Drives 8400 support the configuration of a maximum of 16 process data words (max. 32 bytes). The optional activation of the cyclic parameter data channel additionally occupies 4 process data word (8 bytes).

The user data lengths for process input data and process output data are the same.

Description of the device data base file

Selection text	Parameter data with consistency	Process data		Assigned IO memory
		with consistency	without consistency	
DRIVECOM-PAR (cons) + PCD (**n**W cons)	Yes	**n** words	-	**4** + **n** words
PCD (**n**W cons)	-	**n** words	-	**n** words
PCD (**n**W)	-	-	**4** words	**4** words
n = 1 ... 16 process data words				

Example of selecting the device data base file

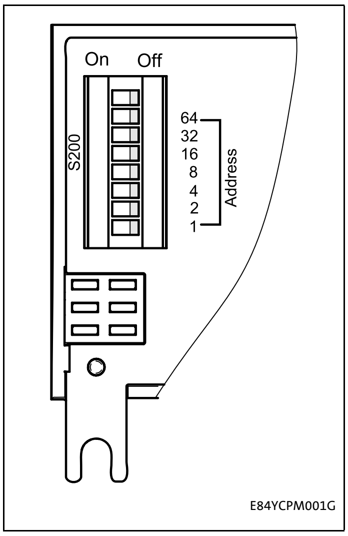
```
DRIVECOM-PAR (cons) + PCD (8W cons)
```

- "Drivecom-PAR (cons)" = DP-V0 parameter data channel (4 words)
- "PCD (8W cons)" = 8 process data words

Tip!

A detailed description of consistency is given in the chapter "Consistent parameter data" (📖 80).

6.3 Setting the station address



[6-1] DIP switch

The station address can be set via DIP switches **1 ... 64** or via the »Engineer« (code C13899).

The unlabelled DIP switch (topmost position) does not have any function.

Lenze setting: all switches in OFF position

The station addresses must differ from each other if several networked PROFIBUS stations are used.

The station address can be set via DIP switches **1 ... 64** or via the »Engineer« (code C13899).

	Setting the station address via ...	
	DIP switch	**C13899**
Condition	At least one switch **1 ... 64 = ON**	• Switches **1 ... 64 = OFF** • All switches **1 ... 64 = ON** (invalid value "127")

The housing labelling indicates the valencies of the individual DIP switches for setting the station address.

DIP switch	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Switch position	OFF	OFF	**ON**	OFF	**ON**	**ON**	**ON**
Value	0	0	16	0	4	2	1
Station address	**= sum of the valencies = 16 + 4 + 2 + 1 = 23** DIP switch positions for setting the station address (📖 107)						

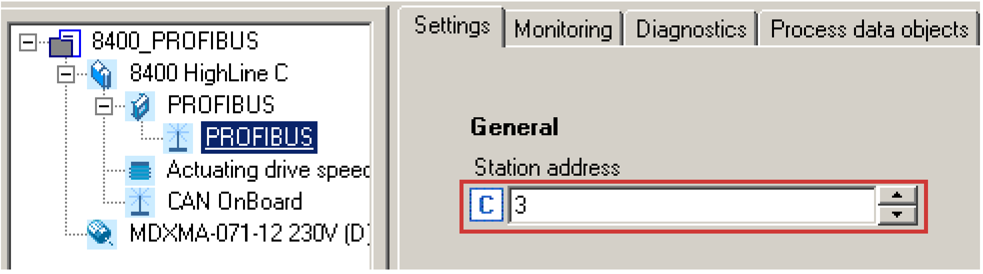
- Valid address range: 1 ... 126 (max. 126 slave stations)
- C13920: Display of the current address setting of the switches
- C13864: Display of the station address active on the PROFIBUS

Note!

Switch off the voltage supply of the communication module and then on again in order to activate changed settings.

Setting the station address via the »Engineer«

In the »Engineer«, the station address can be set via the **Settings** tab.



Impermissible addresses are displayed in red in the **Station address** (code C13899).

Save the changed settings with device command **C00002/11** (save all parameter sets).

6.4 Initial switch-on

Documentation for the standard device

Observe the safety instructions and residual hazards stated.

Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

After communication has been established, the externally supplied module operates independently of the power on/off state of the standard device.

Protection against uncontrolled restart

After a fault (e.g. short-time mains failure), the restart of a drive is not always wanted and - in some cases - even not allowed.

In the Lenze setting of the Inverter Drives 8400, the restart protection is activated.

The restart behaviour of the inverter can be set via **C00142** ("Autostart Option"):

C00142 = 9 (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- Bit 0 (inhibited at power-on) and bit 3 (inhibited at undervoltage) are set.
- The drive starts in a controlled mode by explicitly enabling the inverter: LOW-HIGH edge at digital input X4/RFR.

C00142 = 8 (enabled)

- To enable the device directly at power-on, bit 0 must be set to zero (FALSE).
- An uncontrolled restart of the drive is possible.

6.5 Going online with »Engineer« via TCI

Via Tool Calling Interfaces (TCI) you can connect to a TCI-capable integrated development environment and parameterise and diagnose your field devices without having to exit the integrated development environment.

You cannot set the TCI communication path directly in the »Engineer«. The selection is carried out by the »STEP7« Siemens software.

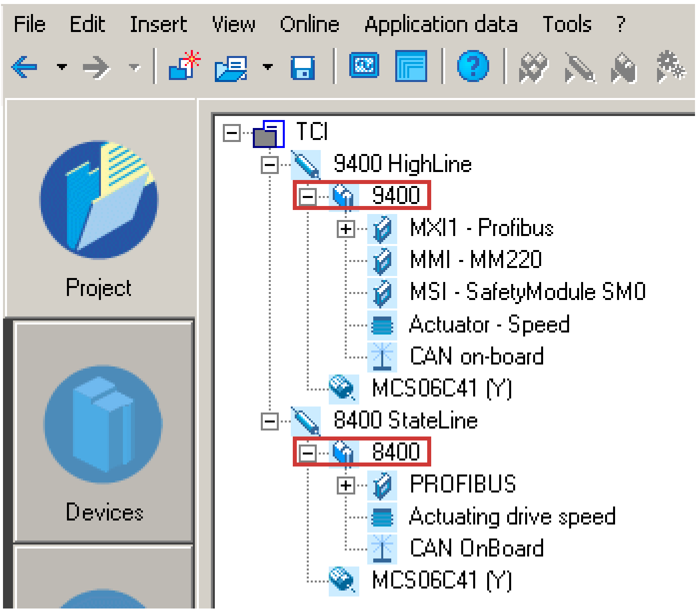
The TCI function requires a PN/DP-CPU. Information on the Siemens PLC types that are equipped with the TCI function is provided via the Siemens Support at:

http://support.automation.siemens.com

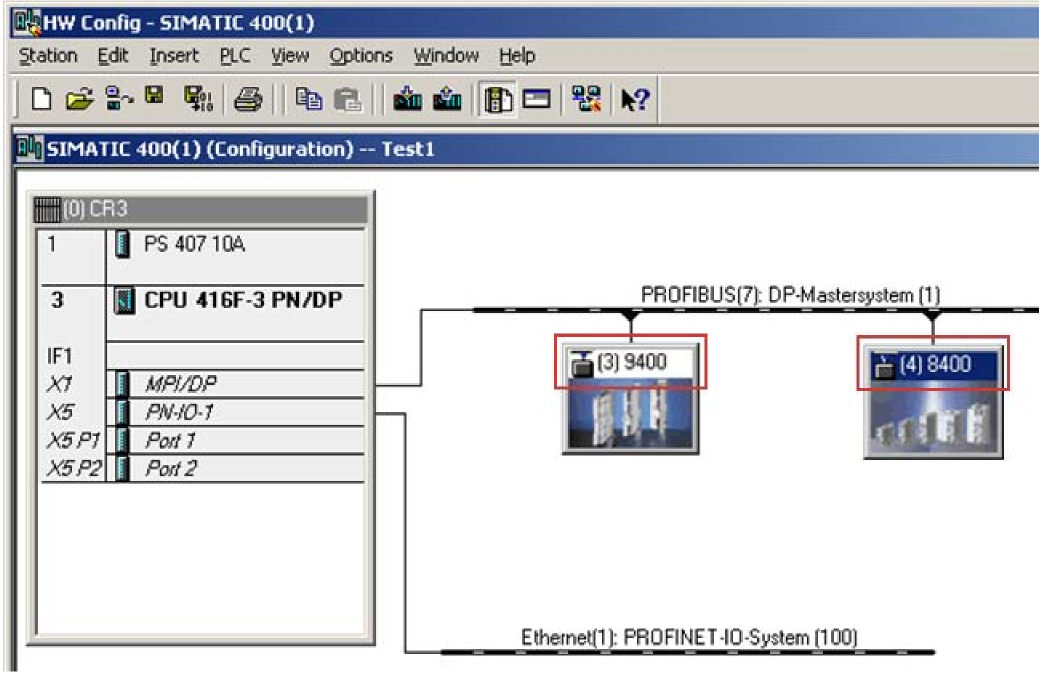
How to configure TCI communication:

1. Allocate names for the individual axes in the »Engineer« project.

   In our case, "9400" was allocated for 9400 HighLine, and "8400" for 8400 StateLine:

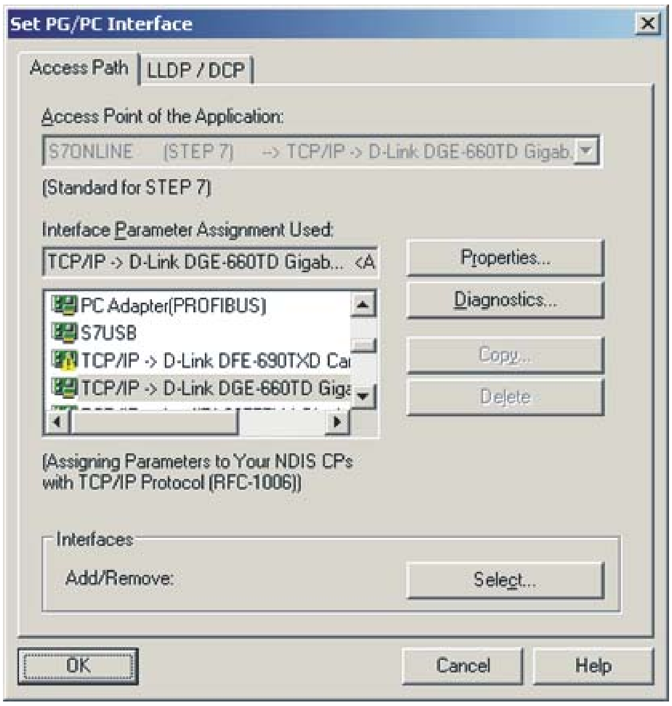


2. In »STEP7« in the »HW Konfig« ...
   - create the Lenze PROFIBUS stations with the corresponding station addresses and
   - create a PROFIBUS network.

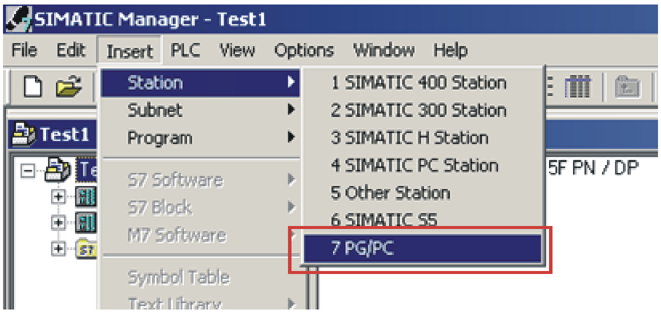


   Here a Servo Drive 9400 (address 3) and an Inverter Drive 8400 (address 4) are operated on the PROFIBUS.
   - The names of the PROFIBUS slaves in the »HW Konfig« must be identical to those of the corresponding Lenze axes in the »Engineer« (here "9400" and "8400").
   - The selection of the process data configuration has no impact on TCI communication.

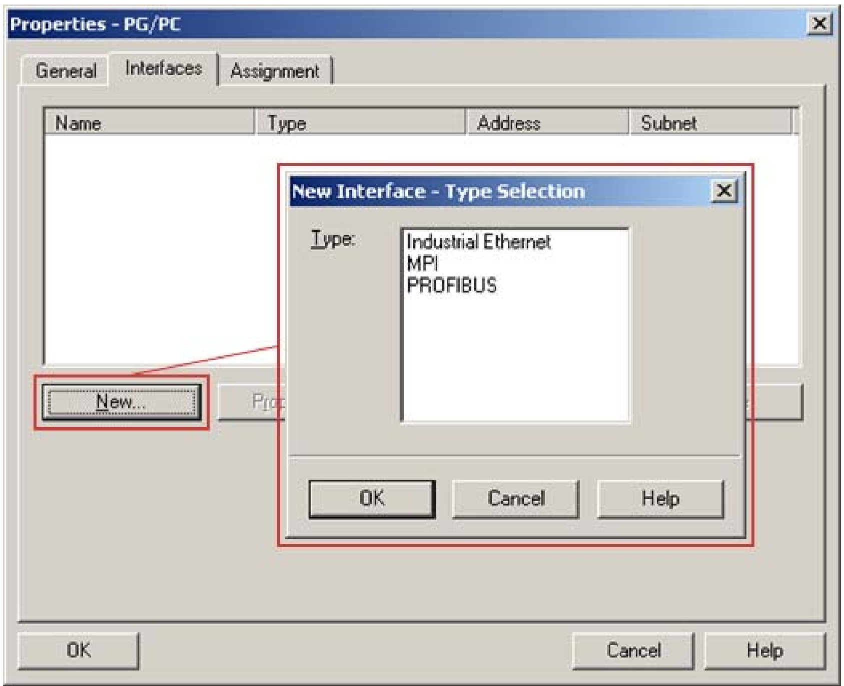
3. Establish an Ethernet connection to the PROFIBUS CPU.



4. Load the »STEP7« project to the CPU.
5. Use the menu command **Insert → Station → 7 PG/PC** to integrate a PG/PC station into the »STEP7« project.

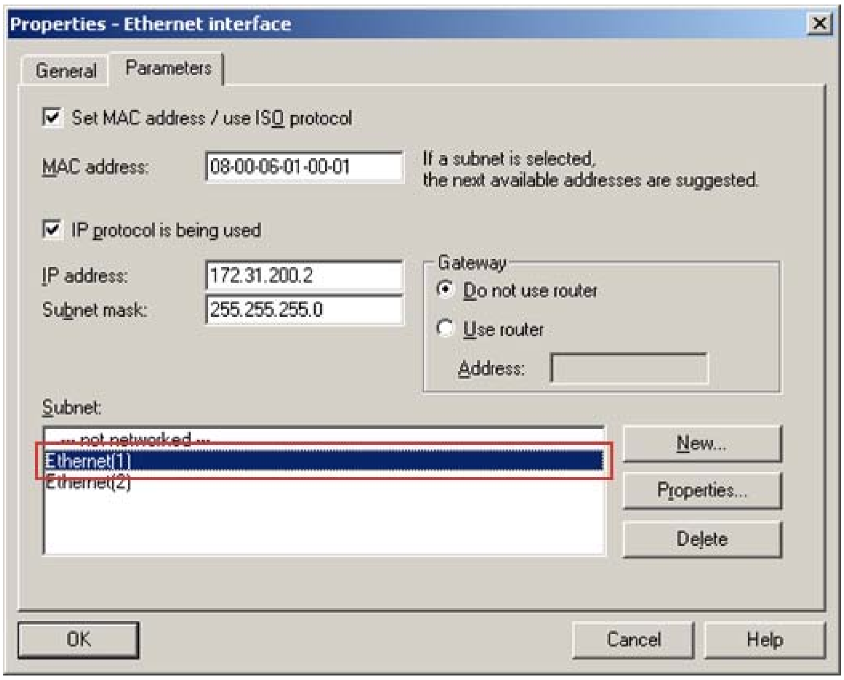


6. By double-clicking the PG/PC station inserted, open its "Properties" dialog.
7. Under the **Interfaces** tab, select a new Ethernet interface and confirm the selection with **OK**.



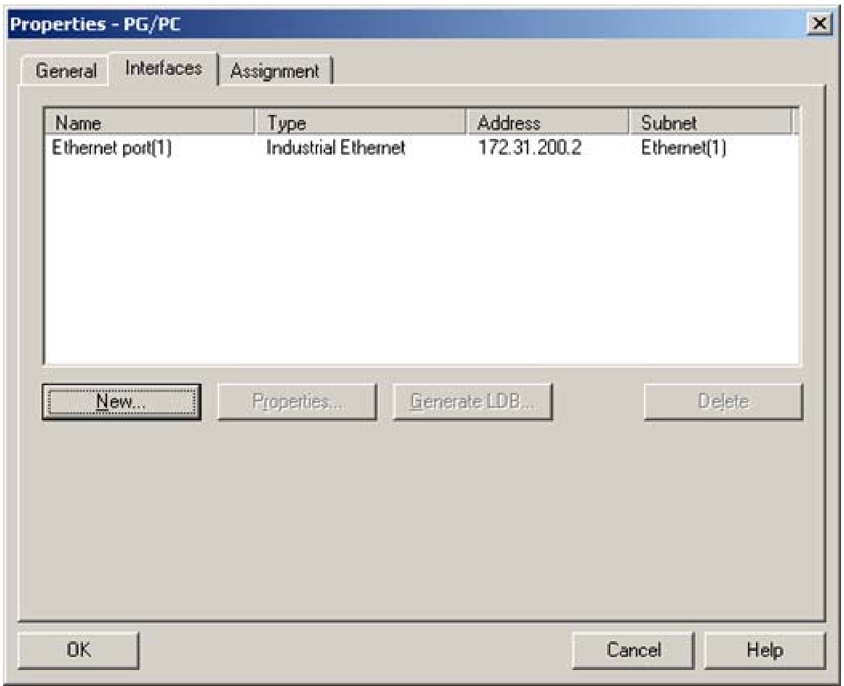
8. Select the Ethernet connection which you are using to go online with »STEP7« (the same Ethernet connection that has been configured in the »HW Konfig«).

   In our case this is the Ethernet(1) connection:

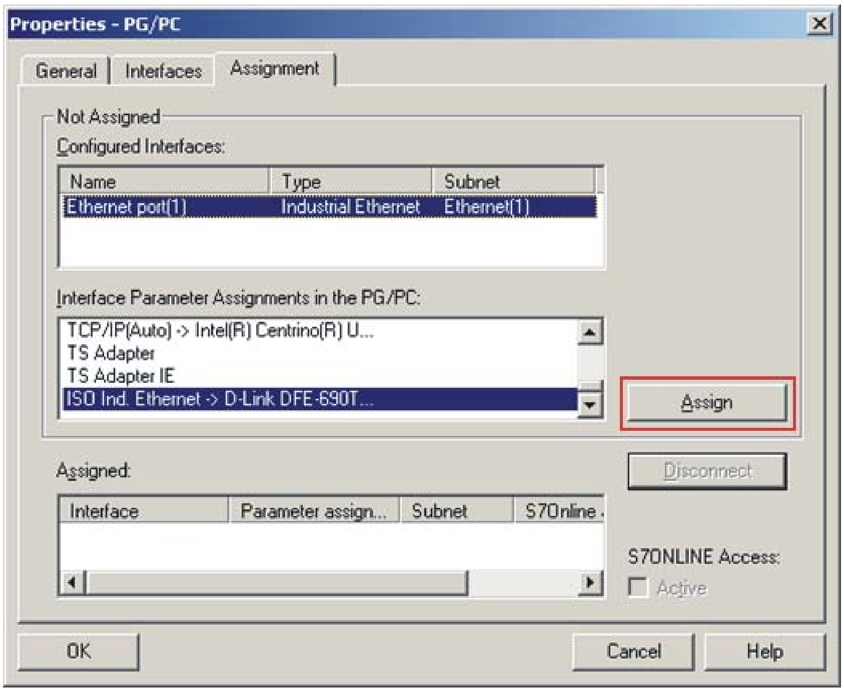


9. Confirm the selection with **OK**.

   The connection has been accepted.

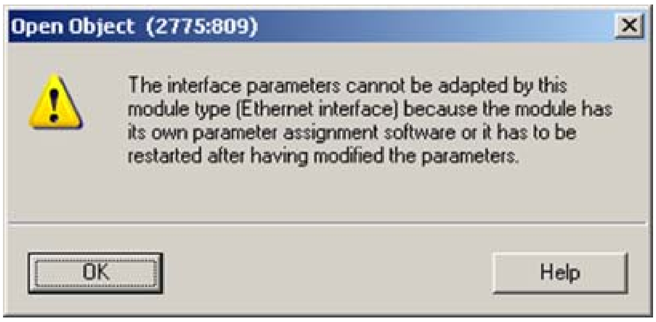


10. Select the actual PG/PC connection under the **Assignment** tab.

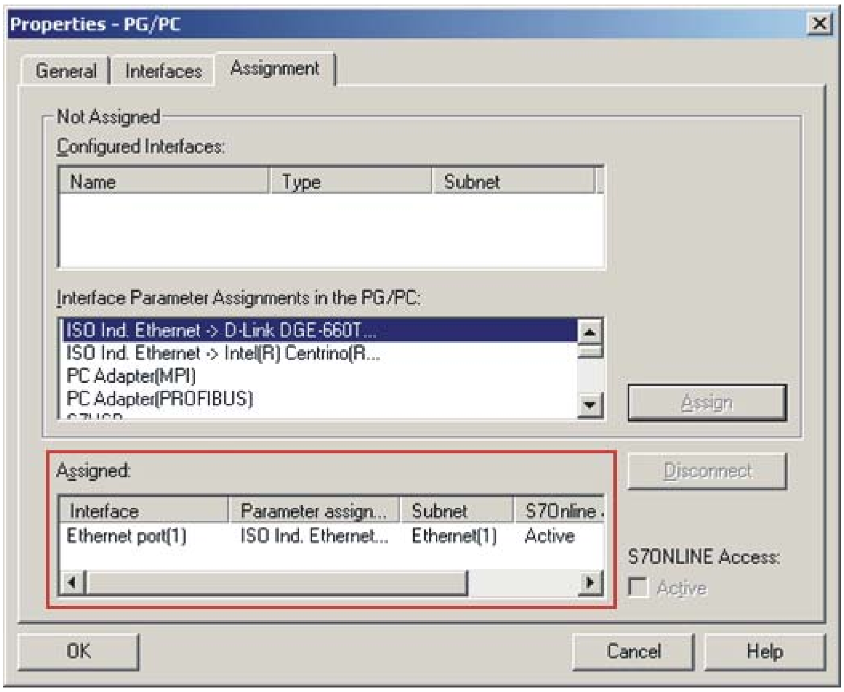


The connection highlighted is assigned by means of the **Assign** button.

11. Confirm the following message with **OK**.

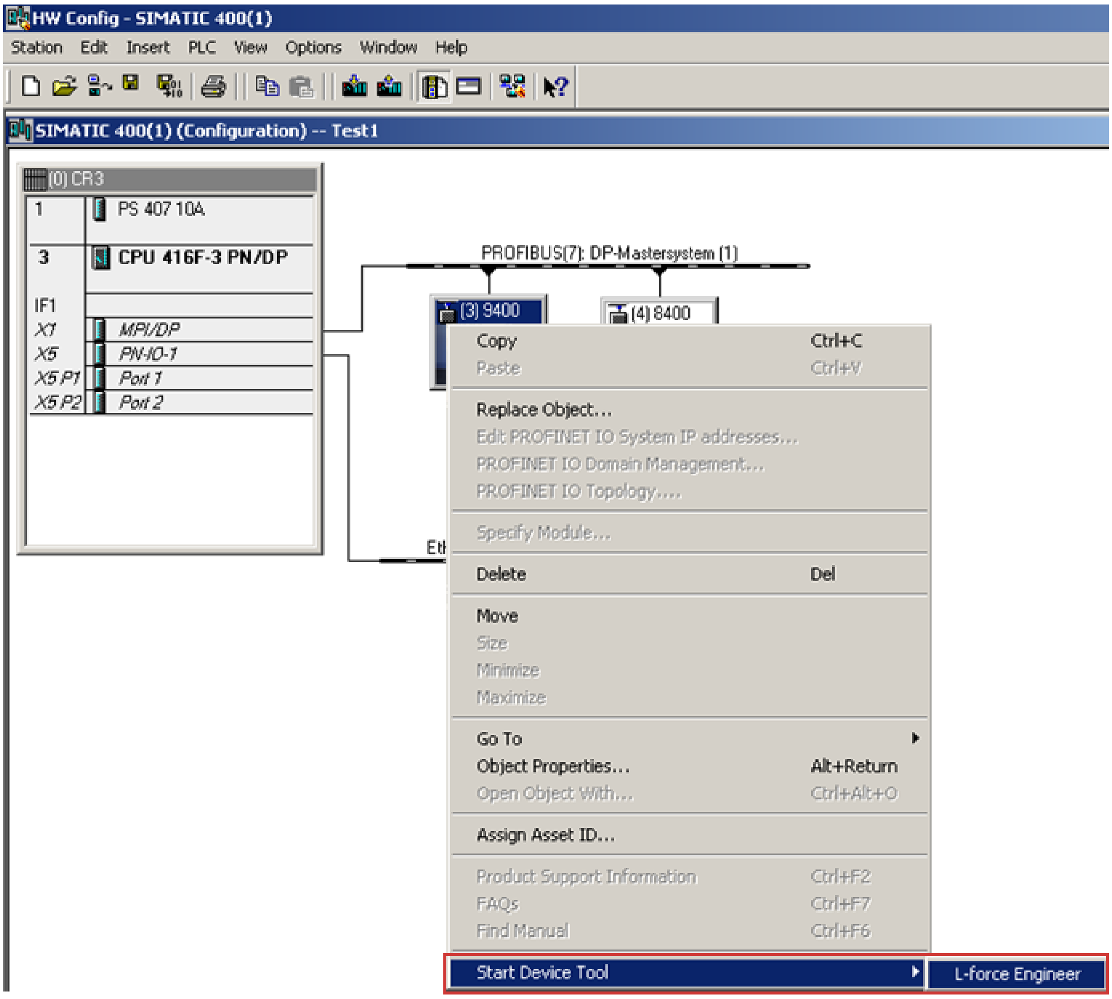


12. After the assignment, the connection appears in the "Assigned" display area. Close the dialog with **OK**.

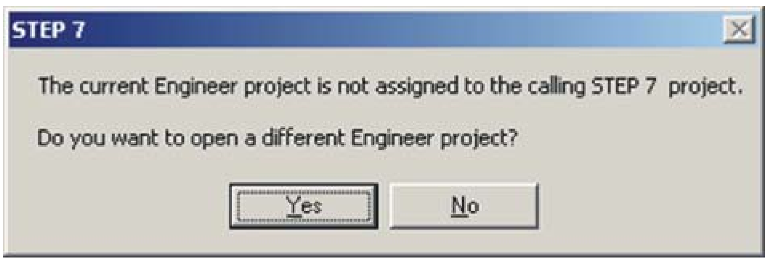


13. In the »STEP7« project, the PG/PC station is marked with a yellow arrow. (The connection selected is active.)

14. Start the transfer of the TCI communication parameters in the »HW Konfig« using the right mouse button and the menu command **Start Device Tool→ L-force Engineer**.



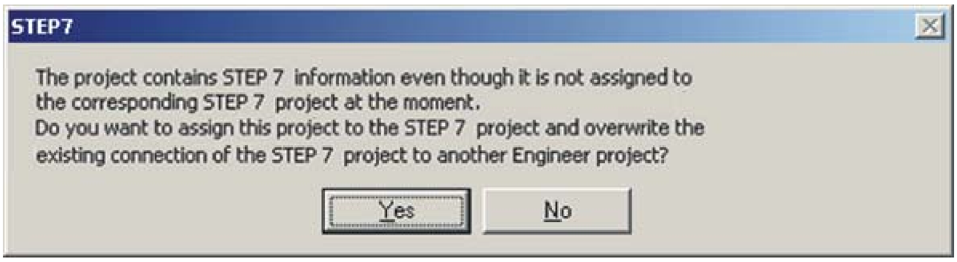
15. If the »Engineer« has already been started with the applicable project, the following message will appear:



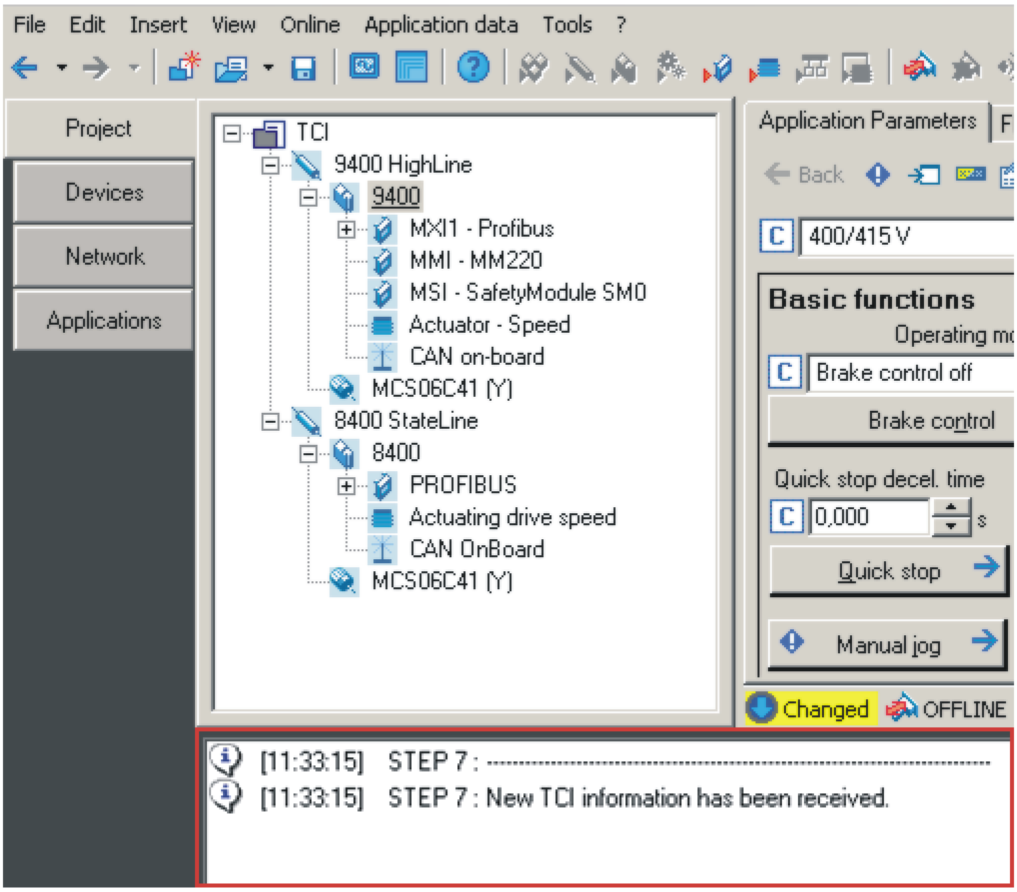
- The message says that the »Engineer« project is not set to a TCI communication path and provides information about whether this action is to be executed now.
- If you confirm the message with **Yes**, the applicable TCI communication parameter settings of the »STEP7« project are transferred to the »Engineer«.

If the »Engineer« has not already been started, it is started automatically now and you have to open the applicable project.

If the project selected has not been set to a TCI communication path yet, this can now be executed by clicking **Yes**:

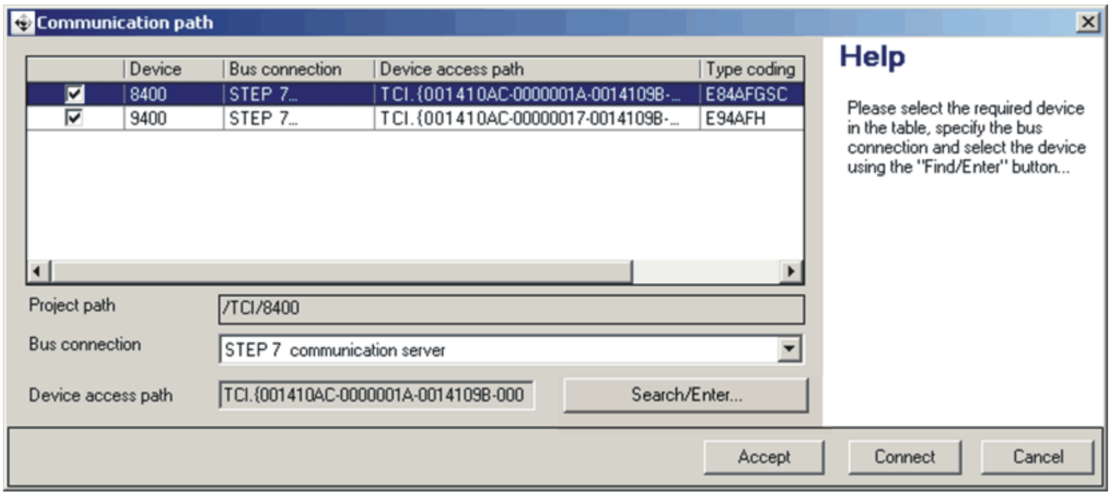


16. The transfer of the TCI communication parameters is documented in the »Engineer« message window.



Here the communication settings have been carried out successfully. The individual PROFIBUS addresses in the respective codes have been adapted to the »STEP7« project.

17. If you now call the "Go online" function of the »Engineer«, the TCI communication settings are displayed as follows:



- "STEP7 Communication Server" appears as bus connection.
- The device access path contains a very long string.
- Use the **Connect** button to establish an online connection.
- By means of the **Search/Enter** button, you can update the TCI communication parameters.

7 Data transfer

The PROFIBUS master and inverter communicate through the exchange of data telegrams via PROFIBUS. The user data area of the data telegram contains parameter data or process data. In the inverter, different communication channels are assigned to the parameter data and process data.

Communication channels

The process data channel serves to transfer process data.

- The process data serve to control the inverter.
- The controller (master) can directly access the process data. In the PLC, for instance, the data are directly saved to the I/O area.
- Process data are not saved in the inverter.
- Process data are transferred cyclically between the controller and the inverters (permanent exchange of current input / output data).
- Process data are, for instance, setpoints, actual values, control words and status words.
- The Inverter Drives 8400 can exchange a maximum of 16 process data word (16 bits/word) per direction.

Note!

Observe the direction of the information flow!
- Process input data (Rx data):
  - Process data from the inverter (slave) to the master
- Process output data (Tx data):
  - Process data from the master to the inverter (slave)

The parameter data channel serves to transfer parameter data.

- The parameter data channel provides access to all Lenze codes.
- In general, the parameter data transfer is not time-critical.
- Parameter data are, for instance, operating parameters, diagnostic information, and motor data.
- Parameter changes must be saved by means of code **C00002** of the Inverter Drive 8400.

8 Process data transfer

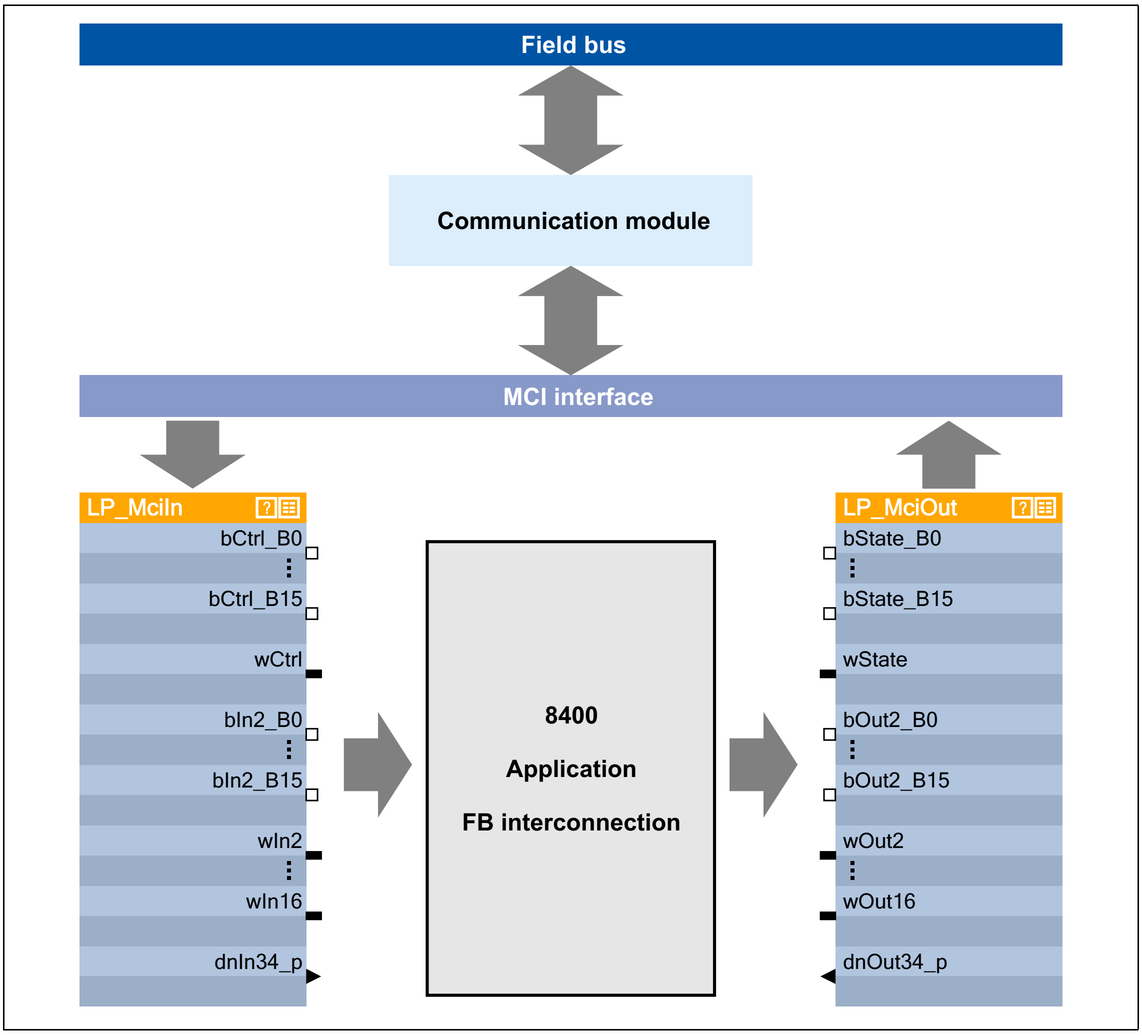
8.1 Access to process data / PDO mapping

Note!

PDO mapping and the objects required for this purpose are not supported in the software version 01.00 of the E84AYCPM communication module.

The process data (MCI-PDOs) are transferred via the MCI interface.

- A maximum of 16 words for each direction is exchanged.
- The process data are accessed via the **LP_MciIn** and **LP_MciOut** port blocks. These port blocks are also referred to as process data channels.
- The port/function block interconnection of the process data objects (PDO) takes place via the Lenze »Engineer«.



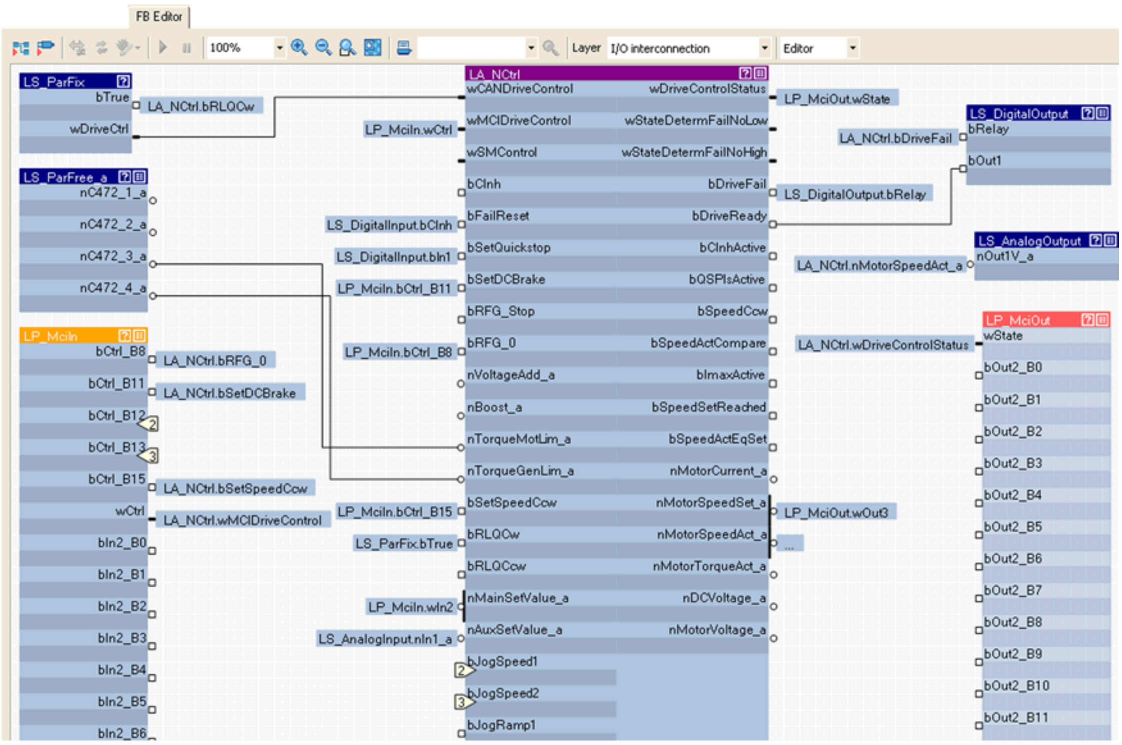
[8-1] External and internal data transfer between the bus system, inverter, and function block interconnection

Software manual/ »Engineer« online help for Inverter Drives 8400
Here you'll find some detailed information about the port/function block interconnection in the »Engineer« and about port blocks.

8.2 Preconfigured port interconnection of the process data objects (PDO)

The preconfigured port interconnection of the process data objects is activated by setting standard device code **C00007 = "40: MCI"**.

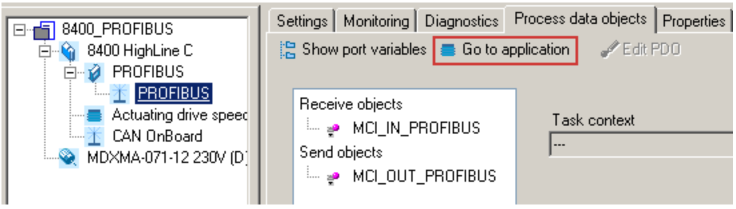
It is possible to display the port blocks "LP_MciIn" and "LP_MciOut" as well as the preconfigured interconnections in the »FB Editor« :



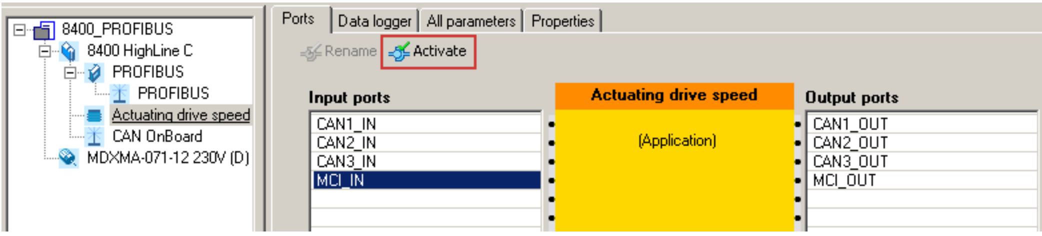
8.3 Free configuration of the port interconnection of process data objects (PDO)

How to freely configure the port interconnection in the »Engineer«:

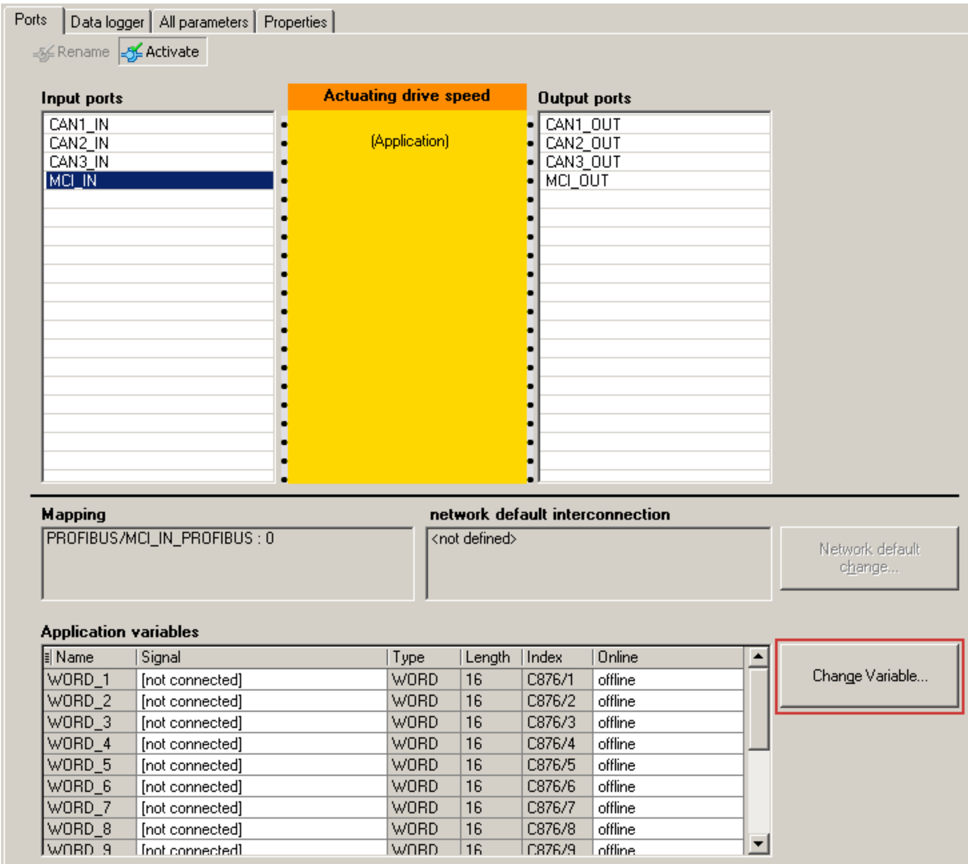
1. Go to the **Process data objects** tab and click **Go to application**.



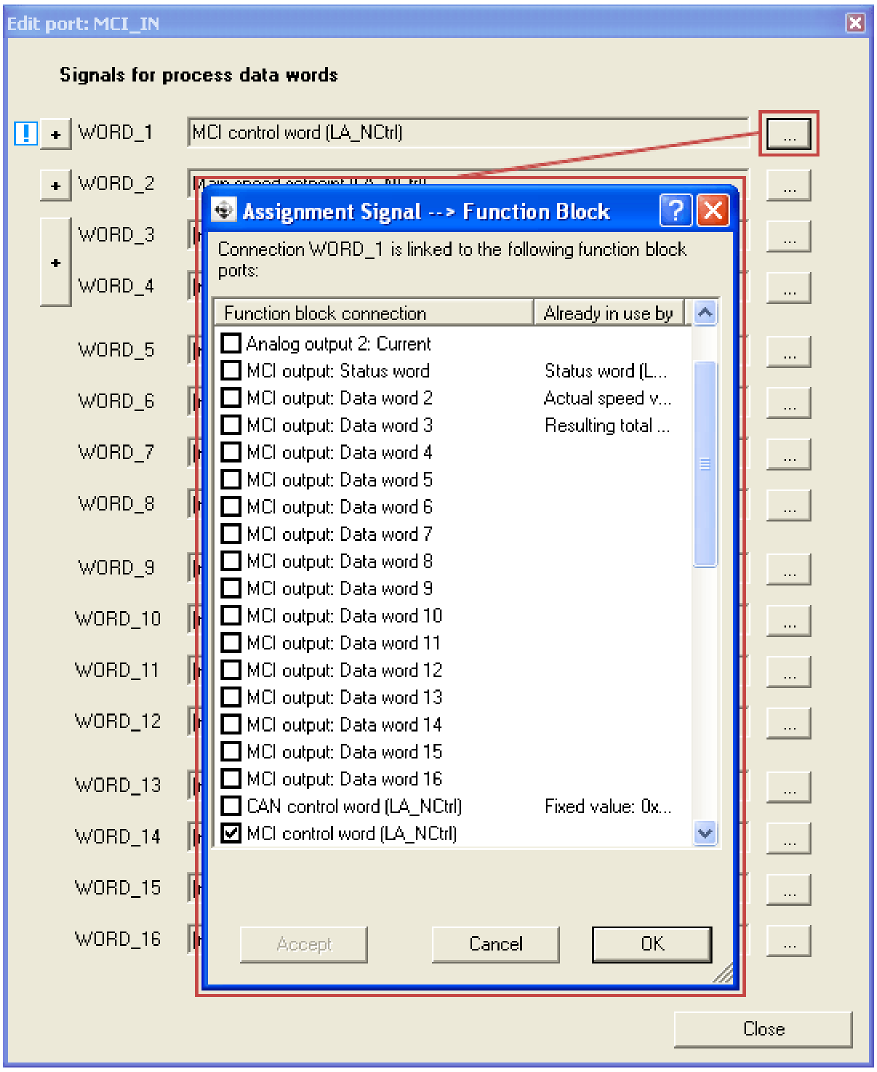
2. Go to the **Ports** tab and select the port block "MCI_IN" or "MCI_OUT" via mouse-click. Activate it by clicking **Activate**.



3. Click the **Edit port ...** button.

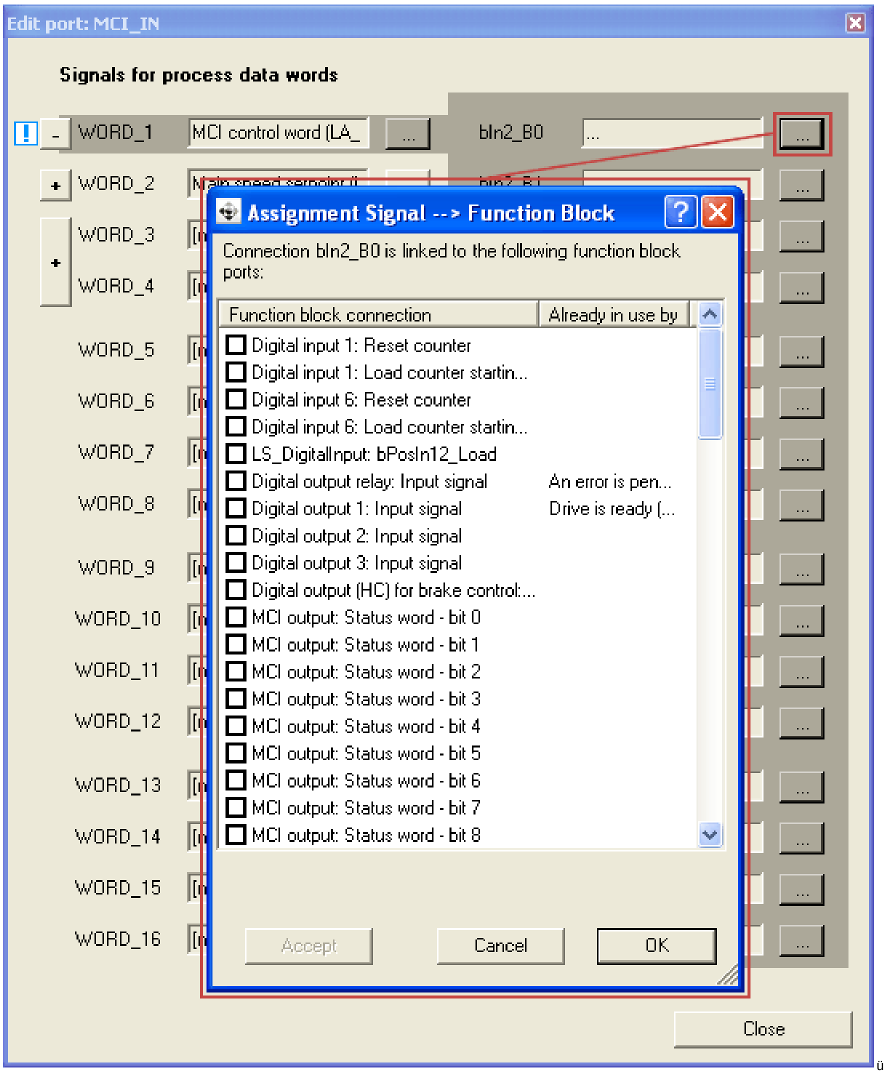


4. Via the ... button, you can assign signals to the process data words in the *Assignment Signal --> Function Block* dialog box.
   → Select the signals and then confirm the selection with **OK**.



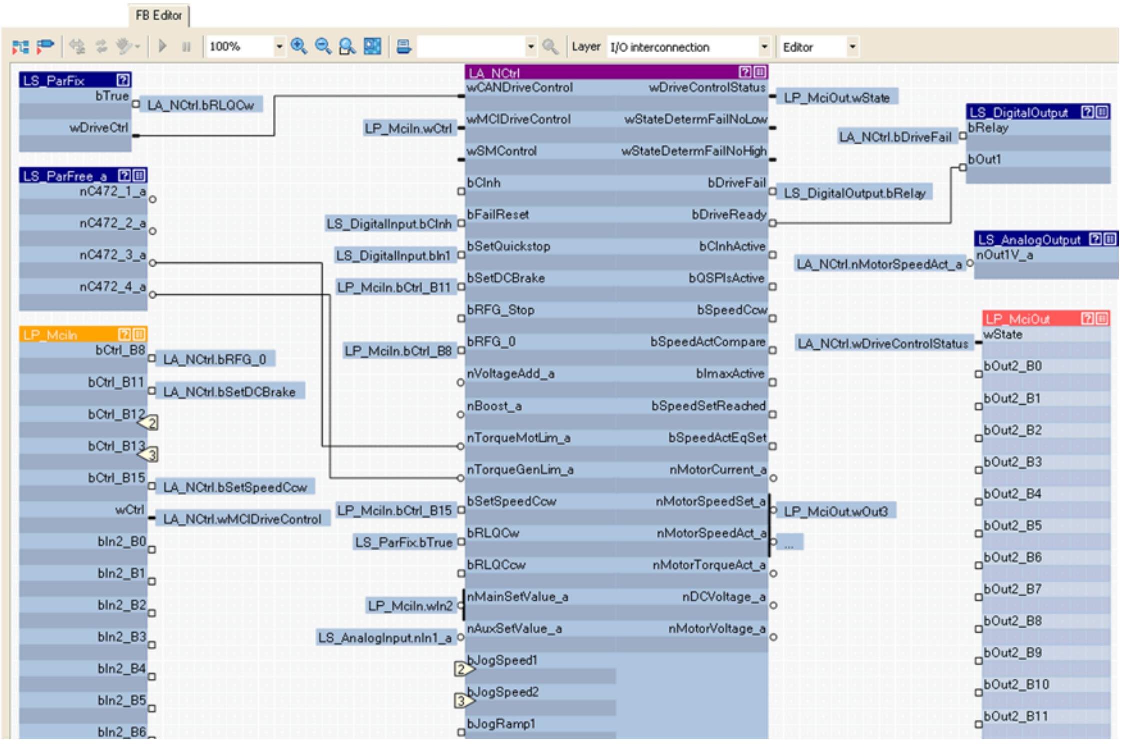
For the process data words WORD_1 and WORD_2, you can also assign signals to the individual control bits and status bits via the [+] and [...] buttons.

➜ Select the signals and then confirm the selection with **OK**.



Tip!

If the port blocks "LP_MciIn" and "LP_MciOut" are activated (see step 1), they are displayed in the »FB Editor«. Here you can also assign signals to the process data words.



9 Parameter data transfer

The E84AYCPM communication module supports the cyclic and acyclic transmission of parameter data:

- Cyclic DP-V0 parameter data are based on the DRIVECOM profile.
  If the DP-V0 parameter data channel is active, it additionally occupies 4 words of the input data and the output data.
- Acyclic DP-V1 parameter data are based on the PROFIdrive profile.

9.1 Addressing of the parameter data

The parameter data are addressed via codes which can be found in this documentation and in the corresponding documentation of your inverter.

▶ Parameter reference (📖 95)

Addressing of Lenze parameters

In the case of the DP-V0 parameter data channel, the parameters of a device are not addressed directly via Lenze code numbers, but via indices (bytes 3 + 4) and subindices (byte 2).

The conversion is made via an offset (24575 / 0x5FFF):

- PROFIBUS-DP $\text{index}_{\text{dec}}$ = 24575 - Lenze code number
- PROFIBUS-DP $\text{index}_{\text{hex}}$ = 0x5FFF - Lenze code $\text{number}_{\text{hex}}$

Example of C00105 (quick stop deceleration time):

- PROFIBUS-DP $\text{index}_{\text{dec}}$ = 24575 - 105 = 24470
- PROFIBUS-DP $\text{index}_{\text{hex}}$ = 0x5FFF - 0x69 = 0x5F96

The parameter values are entered into the user data (bytes 5 to 8) of the telegram.

9.2 DRIVECOM parameter data channel (DP-V0)

The DRIVECOM parameter data channel (DP-V0) ...

- enables parameter setting and diagnosing of the inverter.
- provides access to all Lenze parameters (codes).
- additionally occupies 4 words (16 bits/word) of the input and output data words in the master.
- is identical for both transmission directions.

9.2.1 Telegram structure (overview)

The telegram of the parameter data channel consists of a total of 8 bytes:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

The individual bytes are described in detail in the following subchapters.

9.2.2 Byte 1: Service

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

Request and response control for the parameter data channel

7	6	5	4	3	2	1	0

[9-1] Method of counting for bits 0 ... 7

Bit 0 ... 2: Request	
Read/write request from the master to the inverter	
000	No request
001	Read request ▸ Reading parameter data from the inverter (📖 54)
010	Write request (write data to the inverter) ▸ Writing parameter data to the inverter (📖 54)
100	Data transfer abort by the master ▸ Data transfer abort by the master (📖 55)

Bit 3
Reserved

Bit 4/5: Data length	
Data length ≤ 4 bytes in the telegram bytes 5 ... 8 (data 1 ... 4 / error 1 ... 4)	
00	1 byte
01	2 bytes
10	3 bytes
11	4 bytes

Bit 6: Handshake
Indicates a new request. • The state of this (toggle) bit is changed by the master for every new request. • The inverter copies the bit into its response telegram.

Bit 7: Status	
Status information from the inverter to the master when sending the request confirmation. This status bit informs the master whether the request has been carried out without errors.	
0	Request completed without errors.
1	Request not completed because of an error. • The set status bit indicates that the telegram is an "error telegram". The data of bytes 5 ... 8 (data/error) must be interpreted as an error message. ▸ Error codes (📖 58)

9.2.2.1 Reading parameter data from the inverter

General procedure:

1. Define the user data area of the inverter, i.e. define the location of the DP user data in the controller (observe manufacturer-specific information).
2. Enter the address of the required parameter in the "Index" and "Subindex" fields (DP output data).
3. Request in the service byte = read request.
   The handshake bit in the service byte must be changed (DP output data).
4. Check whether the handshake bit in the service byte is the same for the DP input data and the DP output data.
   - If the handshake bit is the same, the response has been received.
   - It is useful to implement a time monitoring tool.
5. Check whether the status bit in the service byte is set:
   - Status bit is not set: The "Data/Error" field contains the required Parameter value (data) (📖 57).
   - Status bit is set: The read request has not been executed correctly. The "Data/Error" field contains the Error codes (📖 58).

9.2.2.2 Writing parameter data to the inverter

General procedure:

1. Define the user data area of the inverter, i.e. define the location of the DP user data in the controller (observe manufacturer-specific information).
2. Enter the address of the required parameter in the "Index" and "Subindex" fields (DP output data).
3. Enter the parameter value in the "Data/Error" field.
4. Request in the service byte = write request.
   The handshake bit in the service byte must be changed (DP output data).
5. Check whether the handshake bit in the service byte is the same for the DP input data and the DP output data.
   - If the handshake bit is the same, the response has been received.
   - It is useful to implement a time monitoring tool.
6. Check whether the status bit in the service byte is set:
   - Status bit is not set: The write request has been executed correctly.
   - Status bit is set: The write request has not been executed correctly. The "Data/Error" field contains the Error codes (📖 58).

9.2.2.3 Abort of data transfer by the inverter

The error telegram is used to abort the transfer.

- The error telegram is marked by a set status bit in the service byte.
- The telegram can either be the response to an "Initiate Read/Write Service" or to a "Read/Write Segment Service".

Inverter response in the event of an error:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1
1t110000	SIDX	IDXH	IDXL	Error Class	Error code	Additional Code High	Additional Code Low

9.2.2.4 Data transfer abort by the master

The master can use this error telegram to abort a running segment transmission.

- The error telegram is marked by a set status bit in the service byte.
- The service byte also contains the request code "4" (100_{bin}).
- Bit 4 and bit 5 in the service byte (data length) are without meaning.
- Additional information (subindex, index, error information) is not transmitted.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1txx0100	0	0	0	0	0	0	0

Inverter response in the case of correct execution:

The inverter confirms the error telegram of the master by also sending an error telegram.

- The error telegram is marked by a set status bit in the service byte.
- In the case of correct execution, the telegram contains the error information "0x00000000" in bytes 5 ... 8.
- Additional information (subindex, index) is not transmitted.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	SIDX	IDXH	IDXL	Error Class	Error code	Additional Code High	Additional Code Low
1t110000	0	0	0	0	0	0	0

9.2.3 Byte 2: Subindex

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	**Subindex**	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

Additional addressing via the subindex is required for those codes of the Inverter Drives 8400 that contain a subcode (see code table).

9.2.4 Bytes 3 + 4: Index

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	**Index High byte**	**Index Low byte**	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

The parameter (Lenze code) is selected via these two bytes according to the formula:

- **Index = 24575 - Lenze code number**

(Also see "Addressing of Lenze parameters" (51))

Example:

The parameter C00105 (quick stop (QSP) deceleration time) is to be addressed:

- Index = 24575 - 105 = 24470 = 0x5F96
- The entries in bytes 3 + 4 for this example would be:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	**0x5F**	**0x96**	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

9.2.5 Bytes 5 ... 8: Parameter value / error information

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	**Data 4 / Error 4**	**Data 3 / Error 3**	**Data 2 / Error 2**	**Data 1 / Error 1**

The state of status bit 7 in the service byte determines the meaning of this data field:

Status bit	Meaning of bytes 5 ... 8
0	Bytes 5 ... 8 contain the parameter value (data 1 ... 4).
1	Bytes 5 ... 8 contain an error message (error 1 ... 4) due to an invalid access. Error codes (58)

Parameter value (data)

Note!

Strings or data blocks cannot be transmitted.

Depending on the data format, the length of the parameter value is between 1 and 4 bytes.

Data are saved in the Motorola format, i.e. first the high byte (high word), then the low byte (low word):

Byte 5	Byte 6	Byte 7	Byte 8
High byte	Low byte	High byte	Low byte
High word		Low word	
Double word			

Principle for the assignment of bytes 5 ... 8 with parameter values of different lengths:

Byte 5	Byte 6	Byte 7	Byte 8
Parameter value (length 1)	0x00	0x00	0x00
Parameter value (length 2)		0x00	0x00
Parameter value (length 4)			

9.2.6 Error codes

The following error messages may appear:

Byte 8	Byte 7	Byte 6	Byte 5	Meaning
Error 1	Error 2	Error 3	Error 4	
0x06	0x03	0x00	0x00	No right to access
0x06	0x05		0x11	Invalid subindex
0x06	0x05		0x12	Data length too large
0x06	0x05		0x13	Data length too small
0x06	0x07		0x00	Object does not exist
0x06	0x08		0x00	Data types do not comply with each other
0x08	0x00		0x00	Request cannot be executed
0x08	0x00		0x20	Request cannot be executed at the moment
0x08	0x00		0x22	Request cannot be executed due to the device status / The parameter can only be changed in the case of a controller inhibit
0x08	0x00		0x30	Value ranged exited
0x08	0x00		0x31	Parameter value too high
0x08	0x00		0x32	Parameter value too low
0x08	0x00		0x80	Hardware error

9.2.7 Telegram examples

9.2.7.1 Read request: Querying the heatsink temperature

The heatsink temperature of the inverter is to be read.

- Code to be read: C00061
- Heatsink temperature: 43 °C

Byte 1: Service (request)

Request = 0t110001$_{bin}$

- Bit 0 ... 2 = 001$_{bin}$ for read request
- Bit 3 = 0 (reserved)
- Bit 4/5 = 01$_{bin}$ for 2-byte data length (only relevant for the response telegram)
- Bit 6 = handshake bit (t ≡ status is changed in the response telegram)
- Bit 7 = status bit (only relevant for the response telegram)

Byte 2: Subindex

Subindex = 0 because code C00061 does not contain any subindices.

Bytes 3 + 4: Index

Index = 24575 - code number = 24575 - 61 = 24514 = 0x5FC2

- Byte 3 (high byte) = 0x5F
- Byte 4 (low byte) = 0xC2

Bytes 5 ... 8: Data

The response telegram contains the value of code C00061:

Data 3 + 4 = 43 [°C] x 1 (internal factor) = 43 = 0x002B

Result:

Request telegram from master to drive:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x01	0x00	0x5F	0xC2	0x00	0x00	0x00	0x00
0t000001$_{bin}$	00000000$_{bin}$	01011111$_{bin}$	11000010$_{bin}$	00000000$_{bin}$	00000000$_{bin}$	00000000$_{bin}$	00000000$_{bin}$
Waiting for change of handshake bit 6 in service byte 1 of the response.							

Response telegram from drive to master (for correct execution):

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x11	0x00	0x5F	0xC2	0x00	0x2B	0x00	0x00
0t010001$_{bin}$	00000000$_{bin}$	01011111$_{bin}$	11000010$_{bin}$	00000000$_{bin}$	00101011$_{bin}$	00000000$_{bin}$	00000000$_{bin}$

9.2.7.2 Write request: Setting the deceleration time for quick stop (QSP)

In the inverter, the deceleration time for quick stop (QSP) is to be set to 50 ms.

- Code to be written: C00105

Byte 1: Service (request)

Request = $0t110010_{bin}$

- Bit 0 ... 2 = 010_{bin} for write request
- Bit 3 = 0 (reserved)
- Bit 4/5 = 11_{bin} for 4-byte data length
- Bit 6 = handshake bit (t ≡ status is changed in the response telegram)
- Bit 7 = status bit (only relevant for the response telegram)

Byte 2: Subindex

Subindex = 0 because code C00105 does not contain any subindices.

Bytes 3 + 4: Index

Index = 24575 - code number = 24575 - 105 = 24470 = 0x5F96

- Byte 3 (high byte) = 0x5F
- Byte 4 (low byte) = 0x96

Bytes 5 ... 8: Data

The parameter value of 0.05 s to be set is multiplied by the code-specific factor of "1000" and entered in the user data:

Data 1 ... 4 = 0.05 [s] x 1000 (internal factor) = 50 = 0x00000032

Result:

Request telegram from master to drive:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x72	0x00	0x5F	0x96	0x00	0x00	0x00	0x32
$0t110010_{bin}$	00000000_{bin}	01011111_{bin}	10010110_{bin}	00000000_{bin}	00000000_{bin}	00000000_{bin}	00110010_{bin}
Waiting for change of handshake bit 6 in service byte 1 of the response							

Response telegram from drive to master (for correct execution):

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x40	0x00	0x5F	0x96	0x00	0x00	0x00	0x32
$0t000000_{bin}$	00000000_{bin}	01011111_{bin}	10010110_{bin}	00000000_{bin}	00000000_{bin}	00000000_{bin}	00110010_{bin}

9.3 PROFIdrive parameter data channel (DP-V1)

Data communication with PROFIBUS-DP-V0 is characterised by cyclic diagnostics and cyclic process data and parameter data transfer.

An optional service expansion is the acyclic parameter data transfer of PROFIBUS-DP-V1. This service does not impair the functionality of the standard services under PROFIBUS-DP-V0.

PROFIBUS-DP-V0 and PROFIBUS-DP-V1 can be operated simultaneously in the same network. This enables the step-by-step expansion or retrofitting of a system.

The services of PROFIBUS-DP-V1 can be used by the class 1 master (PLC) and the class 2 master (diagnostics master, etc.).

The integration of the acyclic service into the fixed bus cycle depends on the corresponding configuration of the class 1 master:

- With configuration, a time slot is reserved.
- Without configuration the acyclic service is *appended* when a class 2 master acyclically accesses a DP-V1 slave.

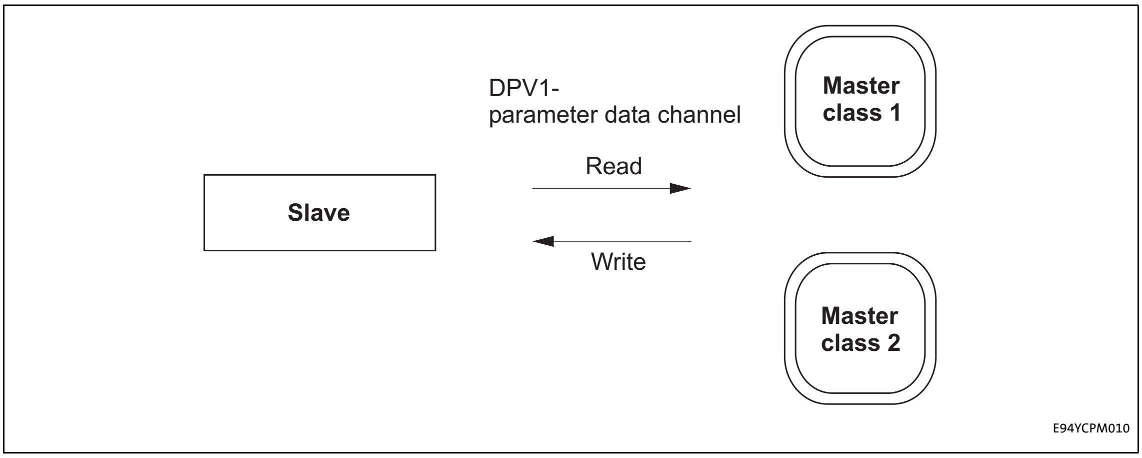
Features

- Parameter number and subindex addresses with a width of 16 bits each.
- Several parameter requests can be combined to one request (multi-parameter request).
- There is always only one parameter request in process (no pipelining).
- A parameter request/response must fit into a data block (max. 240 bytes). Requests/responses cannot be split into several data blocks.
- No spontaneous messages are transferred.
- There are only acyclic parameter requests.
- Profile-specific parameters can be read independently of the slave state.

9.3.1 Connection establishment between master and slave

A class 1 master can always request parameters from a slave if the slave is in the "Data_Exchange" state.

In addition to the class 1 master, a class 2 master can establish a communication connection to the slave:

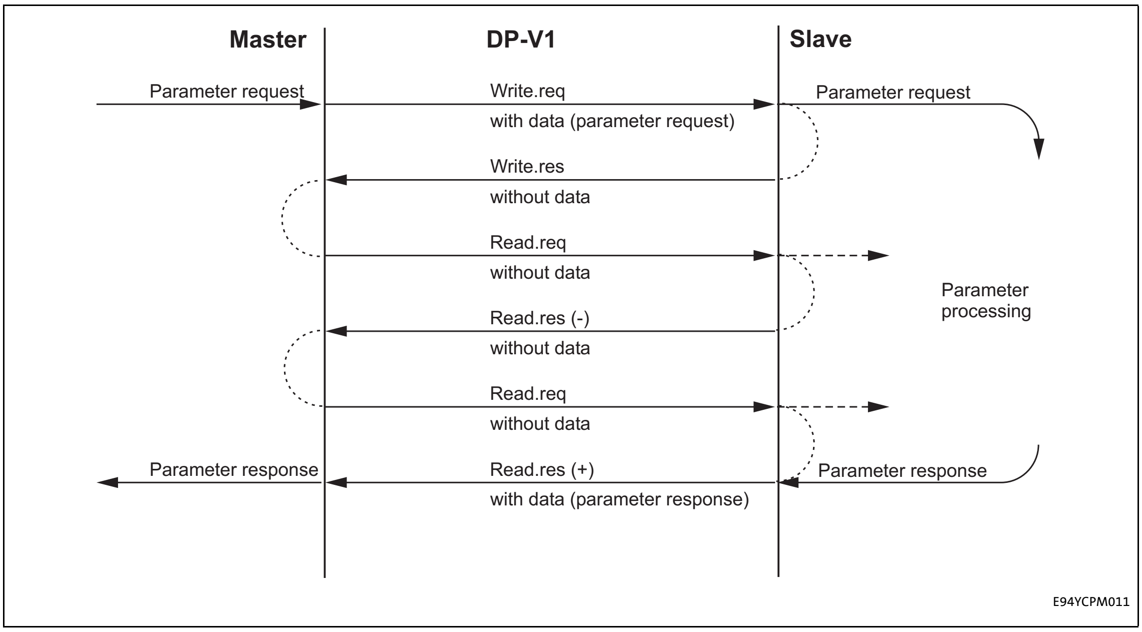


[9-2] Data communication via the DP-V1 parameter data channel

9.3.2 Acyclic data transfer

Note!

A parameter request refers to one or several parameter(s) (multi-parameter request).

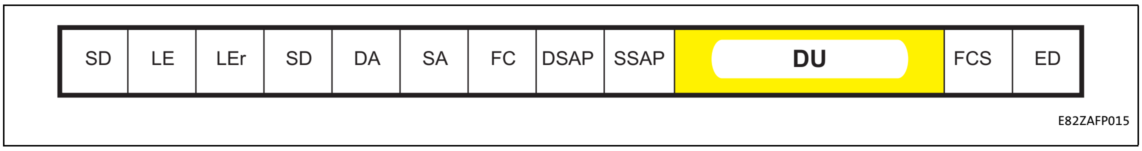


[9-3] Transmission directions

Explanation

- A "Write.req" is used to pass the data set (DB47) to the slave in the form of a parameter request.
- With "Write.res" the master receives the confirmation for the receipt of the message.
- The master requests the response of the slave with "Read.req".
- The slave responds with "Read.res (-)" if processing has not yet been completed.
- After parameter processing, the parameter request is completed by transmitting the parameter response to the master with "Read.res (+)".

9.3.3 Telegram structure



[9-4] PROFIBUS data telegram

The data unit (DU) contains the DP-V1 header and the parameter request or the parameter response.

The following subchapters describe the parameter request and the parameter response in detail.

Note!

The DP-V1 header consists of:

- Function identification
- Slot number
- Data set
- Length of the user data

Please refer to the corresponding PROFIBUS specification for further information on the DP-V1 header.

Assignment of the user data depending on the data type

Depending on the data type used, the user data are assigned as follows:

Data type	Length	User data assignment				
		Byte 1	Byte 2	Byte 3	Byte 4	Byte ...
String	x bytes					
U8	1 byte		0x00			
U16	2 bytes	High byte	Low byte			
U32	4 bytes	High word		Low word		
		High byte	Low byte	High byte	Low byte	

9.3.3.1 Reading parameter data from the inverter

Note!

- When a read request is processed, no parameter value is written to the slave.
- In the case of a multi-parameter read request, the parameter attribute, index, and subindex are repeated with the number "n" of the requested parameters.
- A read request must not exceed the maximum data length of 240 bytes.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is specified by the master
Request identification	U8	0x01: Request parameters for reading
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 (For array elements: Enter the number of array elements required.)

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

9.3.3.2 Response to a correctly executed read request

Note!

Responses to a read request do not contain parameter attributes, indices and subindices.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x01: Parameter has been read
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	0x01 or number of requested subindices/parameters (with several subindices/parameters only the parameter value is repeated). In the case of string codes, the number of characters is entered here.

Parameter value

Byte 7	Byte 8	Byte 9	Byte 10
Value			

Field	Data type	Values
Value	String	Any (length > 4 bytes possible)
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.3.3.3 Response to a read error

Note!

In the case of a multi-parameter request, correct and possible faulty messages are summarised in one telegram. They have the following data contents:

Correct message

- Format: data type of the value requested
- Number of values: as described in the chapter "Reading parameter data from the inverter" (📖 65).
- Parameter value: value requested

Faulty message

- Format: 0x44
- Number of values: 0x01 or 0x02
- Error code without additional information (for number of values = 0x01) or
- Error code with additional information (for number of values = 0x02)

A faulty access to a parameter "n" is indicated at the nth position in the response telegram of a multi-parameter request.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x81: Parameter has not been read • The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7	Byte 8	Byte 9	Byte 10
Error code		Additional information (if available)	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF ▸ Error codes (📖 74)
Additional information (if available)	U16	

9.3.3.4 Writing parameter data to the inverter

Note!

When a multi-parameter write request is transferred, the ...
- Parameter attribute
- Index and subindex

and then the ...
- Parameter format
- Parameter value

... are repeated with the number "n" of the parameters addressed.

A write request must not exceed the maximum data length of 240 bytes.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is defined by the master.
Request identification	U8	0x02: Write parameter
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters addressed)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 (For array elements: Enter the number of array elements required.)

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

Parameter format

Byte 11	Byte 12
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	0x01 or number of written subindices/parameters (with several subindices/parameters only the parameter value is repeated). In the case of string codes, the number of characters is entered here.

Parameter value

Byte 13	Byte 14	Byte 15	Byte 16
Value			

Field	Data type	Values
Value	String	Any (length > 4 bytes possible)
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.3.3.5 Response to a correctly executed write request

Note!

In the case of a multi-parameter request, correct and possible faulty messages are summarised in one telegram. They have the following data contents:

Correct message

- Format: 0x40
- Number of values: 0x00

Faulty message

- Format: 0x44
- Number of values: 0x01 or 0x02
- Error code without additional information (for number of values = 0x01) or with additional information (for number of values = 0x02)

A faulty access to a parameter "n" is indicated at the nth position in the response telegram of a multi-parameter request.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x02: Parameter has been written
Axis	U8	0x00 or 0x01
Number of indices	U8	0xn (n = number of parameter addressed)

9.3.3.6 Response to a write error

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x82: Parameter has not been written • The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters addressed)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7	Byte 8	Byte 9	Byte 10
Error code		Additional information (if available)	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF ▸ Error codes (📖 74)
Additional information (if available)	U16	

9.3.4 Error codes

Error code	Description	Explanation	Additional information
0x0000	Impermissible parameter number	Access to unavailable parameter	-
0x0001	Parameter value cannot be changed	Change access to a parameter value that cannot be changed	Subindex
0x0002	Lower or upper value limit exceeded	Change access with value beyond the value limits	Subindex
0x0003	Faulty subindex	Access to unavailable subindex	Subindex
0x0004	No array	Access with subindex to non-indicated parameter	-
0x0005	Wrong data type	Change access with value that does not match the data type of the parameter	-
0x0006	No setting permitted (only resettable)	Change access with value unequal to 0 where this is not permitted	Subindex
0x0007	Description element cannot be changed	Change access to a description element that cannot be changed	Subindex
0x0008	Reserved	(PROFIdrive profile V2: PPO-Write requested in the IR is not available)	-
0x0009	Description data not available	Access to unavailable description (parameter value is available)	-
0x000A	Reserved	(PROFIdrive profile V2: Wrong access group)	-
0x000B	No parameter change rights	Change access without parameter change rights	-
0x000C	Reserved	(PROFIdrive profile V2: Wrong password)	-
0x000D	Reserved	(PROFIdrive profile V2: Text in the cyclic traffic cannot be read)	-
0x000E	Reserved	(PROFIdrive profile V2: Name in the cyclic traffic cannot be read)	-
0x000F	No text array available	Access to unavailable text array (parameter value is available)	-
0x0010	Reserved	(PROFIdrive profile V2: Missing PPO-Write)	-
0x0011	Request cannot be executed due to the operating state	Access is not possible due to temporary reasons not specified here	-
0x0012	Reserved	(PROFIdrive profile V2: Other error)	-
0x0013	Reserved	(PROFIdrive profile V2: date in the cyclic traffic cannot be read)	-
0x0014	Value impermissible	Change access with the value that is inside the value limits but not permissible for other permanent reasons (parameters with defined individual values)	Subindex
0x0015	Response too long	The length of the current response exceeds the maximum transmittable length	-
0x0016	Parameter address impermissible	Impermissible or non-supported value for attribute, number of subindices, parameter number, or subindex, or a combination	-
0x0017	Format impermissible	Write request: Impermissible or non-supported format of parameter data	-
0x0018	Number of values not consistent	Write request: Number of values of the parameter data do not match the number of subindices in the parameter address	-
0x0019 ... 0x0064	Reserved	-	-

Error code	Description	Explanation	Additional information
0x0065	Manufacturer-specific	-	-
...			
0x00FF			

9.3.5 Telegram examples

9.3.5.1 Read request: Querying the heatsink temperature

The heatsink temperature of the inverter is to be read.

- Code to be read: C00061
- Heatsink temperature: 43 °C

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x01	0x00	0x01
	Request parameters for reading		

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0xC2	0x00	0x00
Index = 24575 - code no. = 24575 - 61 = 24514 = 0x5F C2		No subindex	

Parameter response to a correctly executed read request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x01	0x00	0x01
	Parameter has been read		

Byte 5	Byte 6
Format	Number of values
0x03	0x01
Integer16	1 value

Byte 7	Byte 8
Value	
High byte	Low byte
0x00	0x2B
Value read = 0x 00 2B = 43 x 1 (internal factor) = 43 [°C]	

Parameter response to a read error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x81	0x00	0x01
	Parameter has not been read		

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Error	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning, see the "Error codes" (📖 74) chapter	

9.3.5.2 Write request: Setting the deceleration time for quick stop (QSP)

In the inverter, the deceleration time for quick stop (QSP) is to be set to 50 ms.

Code to be written: C00105

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x02	0x00	0x01
	Write parameter	Axis 0	1 index

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0x96	0x00	0x00
Index = 24575 - code no. = 24575 - 105 = 24470 = 0x5F 96		No subindex	

Byte 11	Byte 12
Format	Number of values
0x43	0x01
Double word	1 value

Byte 13	Byte 14	Byte 15	Byte 16
Value			
High word: high byte	High word: low byte	Low- word: high byte	Low word: low byte
0x00	0x00	0x00	0x32
Value to be written = 0.05 [s] x 1000 (internal factor) = 50 = 0x00 00 00 32			

Parameter response to a correctly executed write request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x02	0x00	0x01
	Parameter has been written		1 index

Parameter response to a read error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x82	0x00	0x01
	Parameter has not been written		1 index

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Error	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning, see the Error codes (📖 74)	

9.4 Consistent parameter data

In the PROFIBUS communication system, data are permanently exchanged between the control system (CPU + PROFIBUS master) and the inverter via the plugged-on slave interface module. Both the PROFIBUS master and the CPU (central processing unit) of the control system access a joint memory: the dual port memory (DPM).

The DPM permits a data exchange in both directions (write/read):

It could happen that a slower PROFIBUS master writing would be overtaken by a faster CPU reading within a cycle time without any further data organisation.

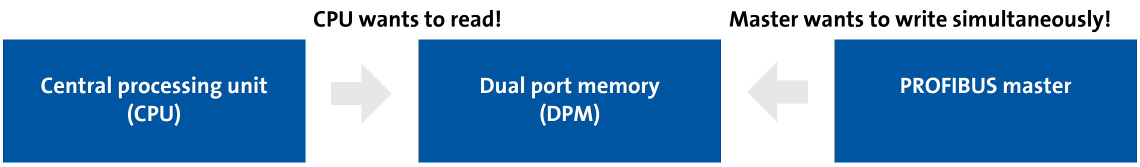
In order to avoid such an impermissible state, the parameter data to be transmitted must be marked as "consistent".

Data communication with consistent data

With consistency, either "reading" or "writing" is possible when the master and the CPU simultaneously access the memory:

- The PROFIBUS master transfers data only as a complete data set.
- The CPU can only access completely updated data sets.
- The PROFIBUS master cannot read or write data as long as the CPU accesses consistent data.

The result becomes clear from the example below:



1. As the master can only write if the CPU does not read, the master waits until the data are read completely by the CPU.
2. The master only writes a complete data set into DPM.

Configuring consistent data

Note!

Consistency is achieved by an appropriate PROFIBUS master configuration.

For this purpose, refer to the documentation for your configuring software.

10 Monitoring

10.1 Permanent interruption of PROFIBUS communication

If PROFIBUS communication is interrupted permanently, e.g. by cable breakage or failure of the PROFIBUS master, no process data are transmitted to the slave being in the "Data Exchange" state.

After the watchdog monitoring time determined by the master has expired, the response parameterised in C13880/1 is executed in the inverter (slave).

The process data are treated according to the setting in C13885. (The data sent last by the master can be used or can be set to zero.)

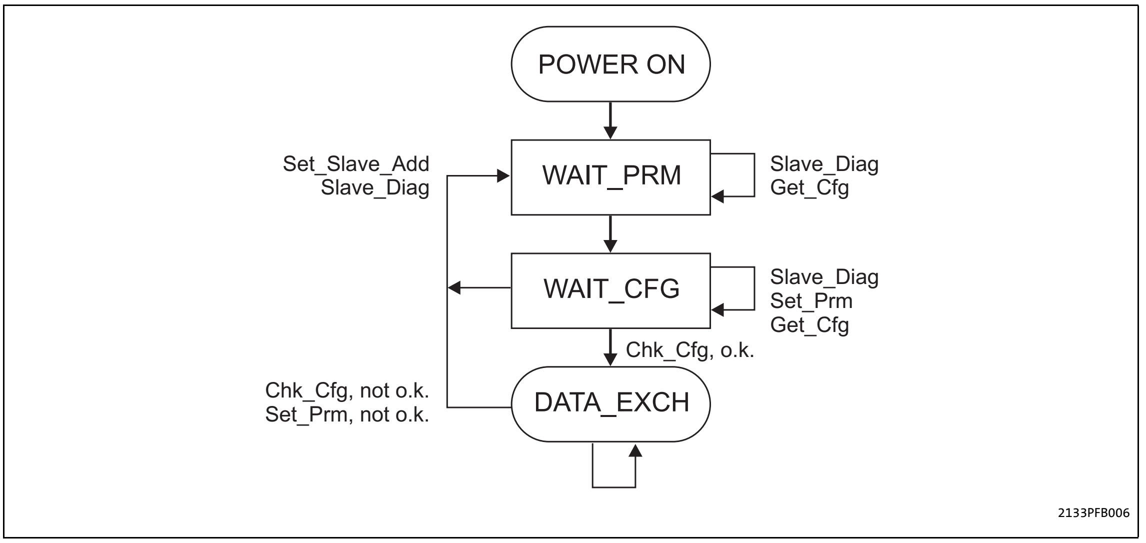
Preconditions for a inverter (slave) response

- A monitoring time of 1 ... 65534 ms for the "Data_Exchange" status (C13881) is set.
  A value of "65535 ms" (Lenze setting) deactivates the monitoring.
- A response for the slave is set in C13880/1 (Lenze setting "No response").
- The slave is in the "Data_Exchange" state.
- The watchdog monitoring time is configured correctly in the master.

If one of these preconditions is not met, the response to the absence of cyclic process data telegrams from the master is not executed.

▶ Settings and displays in the »Engineer« (📖 83)

10.2 Short-time interruption of PROFIBUS communication



[10-1] DP states (Decentralized Peripherals) for short-time interruption of communication

The master detects the communication fault and, only after a few microseconds, transfers the slave to the "WAIT_PRM" status of the DP state machine (see fig. [10-1]).

Only after the state chain of the DP state machine ending in the "Data_Exchange" state (DATA_EXCH) has been passed through, the watchdog monitoring time calculated for the slave (in milliseconds) continues to run.

Note!

The watchdog monitoring time does not continue running if the slave does not reach the "Data_Exchange" state due to repeated communication errors (e.g. caused by loose contact).

Additional monitoring for the data exchange

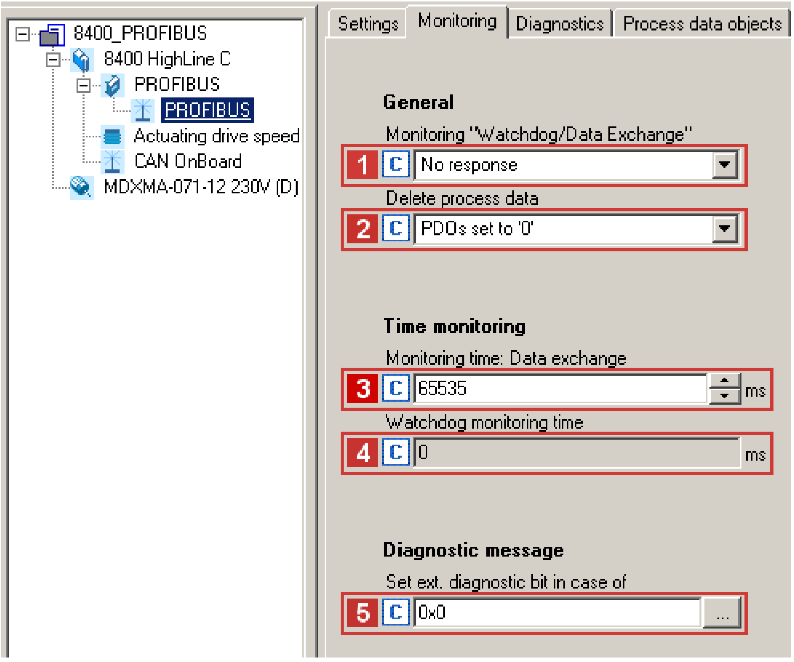
For this reason an additional monitoring function for the data exchange is available with C13881, which is activated when "Data_Exchange" is exited and the parameterised time (0 ... 65535 ms) has expired. The active monitoring triggers the response parameterised in C13880/1.

Note!

Observe the following condition for the time setting:

Monitoring time for the data exchange (C13881) ≤ watchdog monitoring time of the PROFIBUS (C13882/1).

10.3 Settings and displays in the »Engineer«



On the **Monitoring** tab of the »Engineer«, you can set or display the following parameters:

Parameter		Description
1	**Reaction on communication fault** (C13880/1)	The response set here takes place if the PROFIBUS station ... • does not receive a message from the master within the watchdog monitoring time (displayed in C13882/1) if there is an active connection; • recognises that it is not in the "Data_Exchange" status anymore. Please see also the information on 3.
2	**Clear process data** (C13885)	Selection of the process data which the inverter will process in the event of a PROFIBUS failure in order to maintain internal communication. The process data sent last by the master can be used or the process data can be set to zero.
3	**Monitoring time: Data exchange** (C13881)	After the monitoring time set here has elapsed, the response set in 1 takes place for the data exchange. • The value "65535" deactivates the monitoring function. • The monitoring time set here must be smaller than the watchdog monitoring time 4. • A change in monitoring is effective immediately. ▸ Permanent interruption of PROFIBUS communication (📖 81)
4	**Monitoring time: Watchdog** (C13882/1)	Display of the watchdog monitoring time determined by the PROFIBUS master • Monitoring starts with the receipt of the first telegram. • When a value of "0" is displayed, the monitoring function is deactivated. • A change in the watchdog monitoring time in the master is immediately effective. ▸ Permanent interruption of PROFIBUS communication (📖 81)
5	**Set ext. diagnostic bit upon** (C13886)	Bit-coded selection of the error responses in the standard device causing the external diagnostic bit ("diag bit") to be set (see PROFIBUS specification; bit 3 of byte 1 of the DP diagnostic messages). • The diagnostic bit is sent to the PROFIBUS master where it is evaluated separately. • The diagnostic bit is always set when a system error occurs. • The Lenze setting "0" means that the diagnostic bit is not set for the following error responses. • An advanced diagnostic message is always sent.

11 Diagnostics

For diagnosing faults of the PROFIBUS module, the LEDs on the front panel are provided. Furthermore you can query the current bus status via code C13861.

11.1 LED status displays

Note!

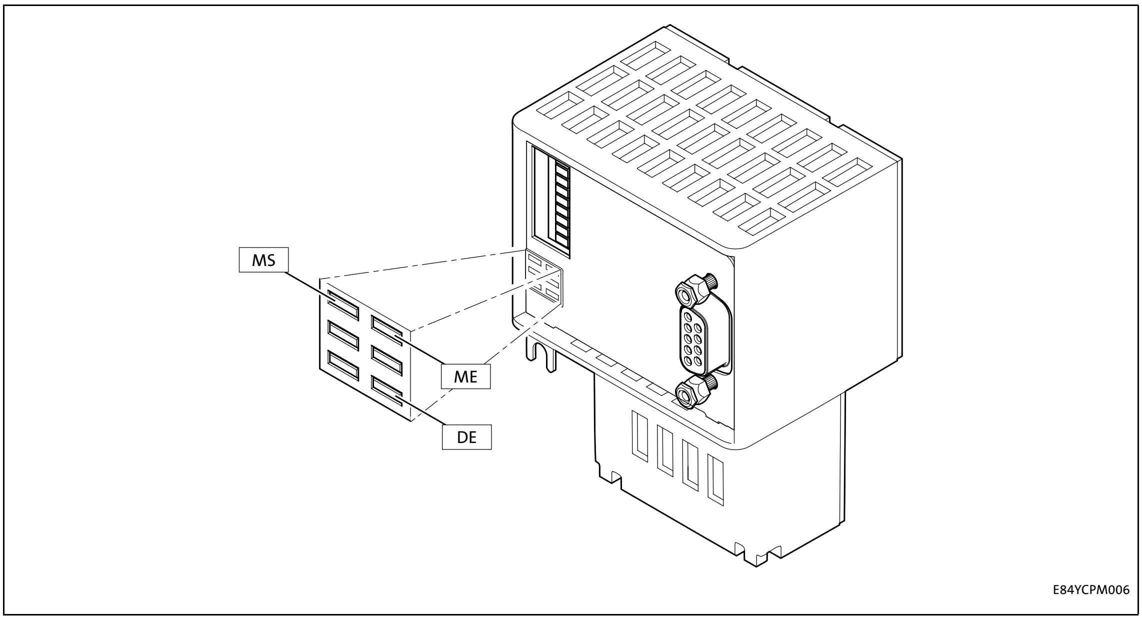
During normal operation, the LED **BS** (📖 86) blinks and the LED **MS** (📖 86) is lit permanently.

The following status displays are distinguished:

- Module status displays (📖 85)
- Fieldbus status displays (📖 86)

11.1.1 Module status displays

The LEDs **MS**, **ME** and **DE** indicate the module status.

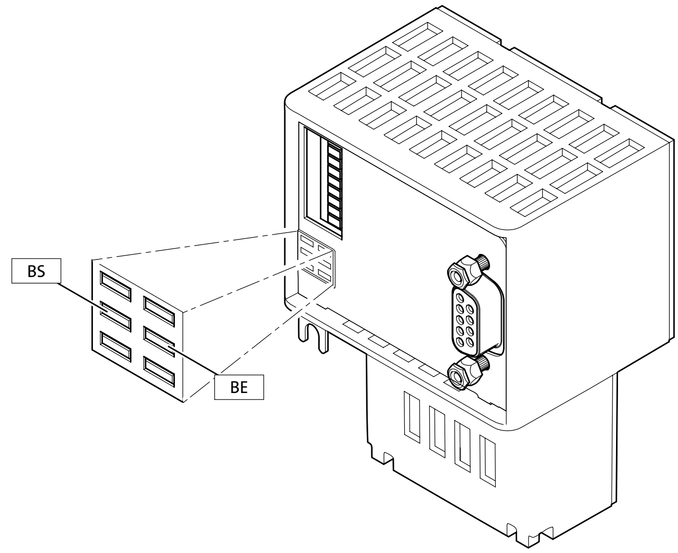


[11-1] LED status displays MS, ME, and DE

LED	Colour	State	Description
MS	Green	On	
			The communication module is supplied with voltage and has established a connection to the standard device.
		Blinking	
			The communication module is supplied with voltage, but has not yet established a connection to the standard device. (Standard device is switched off, initialising or not present.)
ME	Red	On	
			An error concerning the communication module has occurred.
DE	Red	On	
			The communication module is not accepted by the standard device or the standard device is not active. (See notes in the documentation for the standard device.)

11.1.2 Fieldbus status displays

The LEDs **BS** and **BE** indicate the fieldbus status.

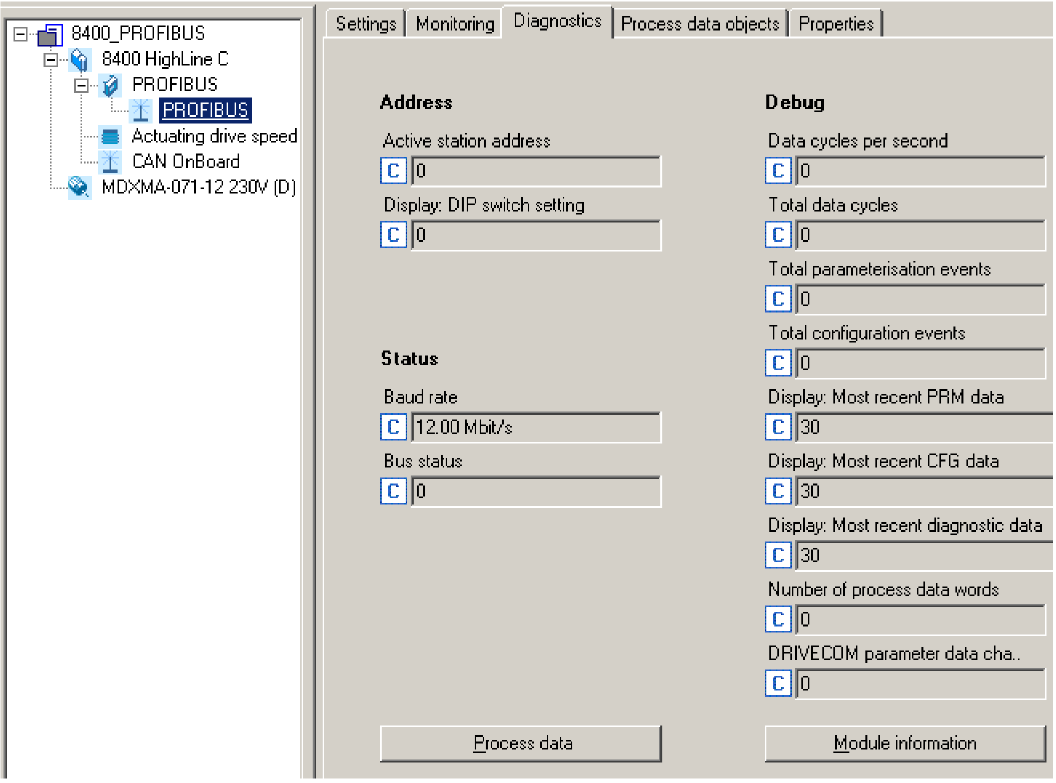


[11-2] LED status displays BS and BE

LED	Colour	State	Description
BS	Green	Off	The communication module is not active on the fieldbus or is being initialised.
		Blinking	
			The communication module is in the DATA_EXCH state ("Data_Exchange"). Data are exchanged via PROFIBUS.
BE	Red	Blinking	
			Incorrect setting for the station address. The communication module is initialised and internally operates with the respective default values.
		On	
			Bus error/fault is active (e.g. bus cable unplugged).

11.2 Diagnosing with the »Engineer«

In the »Engineer«, the **Diagnostics** tab displays various pieces of PROFIBUS diagnostic information.



Querying the current bus status

Code C13861 displays the current PROFIBUS status in a bit-coded form:

Bit assignment				Description	
Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	Reserved	
		Bit 5	**Bit 4**	**Status of the DP state machine (DP-STATE)**	
		0	0	WAIT_PRM	The slave waits for a parameter data telegram after acceleration. Other types of telegrams will not be processed. Data exchange is not yet possible.
		0	1	WAIT_CFG	The slave waits for the configuration telegram that specifies the number of input and output bytes. The master informs the slave about the number of I/O bytes that will be transferred.
		1	0	DATA_EXCH	If the parameter settings as well as the configuration have been accepted by the firmware and by the application, the slave state changes to DATA_EXCH ("Data Exchange", exchange of user data with the master).
		1	1	Not possible	
		Bit 7	**Bit 6**	**Status of the watchdog state machine (WD-STATE)**	
		0	0	BAUD_SEARCH	The PROFIBUS slave is able to automatically detect the baud rate.
		0	1	BAUD_CONTROL	After recognising the correct baud rate, the slave status changes to BAUD_CONTROL and the baud rate is monitored.
		1	0	DP_CONTROL	The DP_CONTROL status serves for response monitoring of the master.
		1	1	Not possible	
Bit 11	**Bit 10**	**Bit 9**	**Bit 8**	**PROFIBUS baud rate detected**	
0	0	0	0	12 Mbps	
0	0	0	1	6 Mbps	
0	0	1	0	3 Mbps	
0	0	1	1	1.5 Mbps	
0	1	0	0	500 kbps	
0	1	0	1	187.5 kbps	
0	1	1	0	93.75 kbps	
0	1	1	1	45.45 kbps	
1	0	0	0	19.2 kbps	
1	0	0	1	9.6 kbps	
Bit 15	**Bit 14**	**Bit 13**	**Bit 12**	Reserved	

11.3 Advanced diagnostic message

Errors in the inverter and its plugged-in modules are transmitted to the PROFIBUS master in the form of advanced diagnostic messages.

Structure of the diagnostic message

Byte	Description
1	Bit 0: Station does not exist (set by the master). Bit 1: Slave is not ready for data exchange. Bit 2: Configuration data do not correspond. Bit 3: Slave has extended diagnostic data. Bit 4: Requested function is not supported by the slave. Bit 5: Slave response is invalid (set by the master) Bit 6: Incorrect parameter setting Bit 7: Slave has been parameterised by another master (set by the master).
2	Bit 0: Slave must be parameterised again. Bit 1: Static diagnostics Bit 2: Permanently set to "1". Bit 3: Watchdog active Bit 4: Freeze command received. Bit 5: Sync command received. Bit 6: Reserved Bit 7: Slave is deactivated (set by the master).
3	Bit 7: Diagnostics overflow - amount of diagnostic information present in the slave is too large to fit into one telegram.
4	Bits 0 ... 7: Master address after parameterisation ("0xFF" without parameterisation)
5	Bits 0 ... 7: ID number (high byte)
6	Bits 0 ... 7: ID number (low byte)
7	Header • The header contains the block length of the advanced diagnostics including the header byte. • In this case, the value of the entry is "0x0A" (bytes 7 ... 16 = 10 bytes).
8	Status_Type The value of this entry is fixed. For the following bit assignment it is "0x81": • Bit 7 = 1: "status" • Bit 0 = 1: "status message" • Value of all other bits = 0
9	Slot_Number The value of the slot number is "0x00".
10	Specifier • An indicated error is entered in the specifier with the identification "0x1" (status coming). • An eliminated error is entered in the specifier with the identification "0x02" (status going). • If no errors are indicated, the entry in the specifier has the value "0x00" (no further differentiation).
11	Reserved
12	
13 ... 16	Error code of the Inverter Drive 8400 • Code **C00165** can be used to read out the contents of the fault memory. • Detailed information regarding the error codes of the Inverter Drive 8400 can be found in the documentation of the inverter.

Example: "Short circuit (OC1)" error in the Inverter Drive 8400

Byte	Value [hex]	Description
1	x	Standard data (PRM_Fault)
...		
6		
7	0A	Block length of the advanced diagnostics = 10 bytes
8	81	Status message
9	00	Slot 0
10	01	Status coming
11	00	
12	00	
13	0B	Error message 0x11C4000B "Short circuit (OC1)" • Error type: "Warning locked" • Subject area: 0x11C4 (current) • Error ID: 0x000B The error number "0x11C4000B" indicates the following: In the "Current" subject area, an overcurrent has been detected. The error response to this is a "Warning locked", which must be unlocked separately after the error has been eliminated.
14	00	
15	C4	
16	11	

12 Error messages

This chapter supplements the error list contained in the software manual and in the »Engineer« online help for Inverter Drives 8400 by the error messages of the communication module.

Software manual/online help for Inverter Drives 8400

Here you can find general information on diagnostics & fault analysis and on error messages.

12.1 Short overview of the PROFIBUS error messages

The following table lists all PROFIBUS error messages in numerical order of the error number. Furthermore the preset error response and – if available – the parameters for setting the error response are specified.

Tip!

When you click the cross-reference in the first column, you will see a detailed description (causes and remedies) of this error message.

Error number			Error text	Error type	Adjustable in
hex	dec (subject area no.)	dec (error no.)			
0x01bc3100	444	12544	Connection to 8400 standard device lost	Error	-
0x01bc5531	444	21809	Memory: No access	Error	-
0x01bc5532	444	21810	Memory: Read error	Error	-
0x01bc5533	444	21811	Memory: Write error	Error	-
0x01bc6010	444	24592	Restart after watchdog reset	Error	-
0x01bc6011	444	24593	Internal error	Error	-
0x01bc6100	444	24832	Internal error	Error	-
0x01bc6101	444	24833	Internal error	Error	-
0x01bc6110	444	24848	Internal error	Error	-
0x01bc641f	444	25631	Invalid parameter set	Error	-
0x01bc6420	444	25632	Error: Lenze settings loaded	Error	-
0x01bc8130	444	33072	Profibus watchdog: Monitoring time elapsed	No response	C13880/1
0x01bc8131	444	33073	Profibus: Data_Exchange state exited	No response	C13880/1
0x01bc8132	444	33074	Profibus Watchdog: DP-V1 MSC2 monitoring time exceeded	No response	C13880/2

12.2 Possible causes and remedies

This chapter lists all PROFIBUS error messages in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

Connection to 8400 standard device lost [0x01bc3100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Network cable (plug) is defective. • Network cable is not connected to the PROFIBUS terminal X201. • Voltage supply is interrupted.	Check cables and terminals. Connect network cable to the PROFIBUS terminal X201.

Memory: No access [0x01bc5531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access to memory was not possible.	Repeat the download of the application (including module)

Memory: Read error [0x01bc5532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameter could not be read.	Repeat the download of the application (including module)

Memory: Write error [0x01bc5533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Parameter could not be written.	Repeat the download of the application (including module)

Restart after watchdog reset [0x01bc6010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module defective.	If this occurs repeatedly, contact the Lenze service.

Internal error [0x01bc6011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module defective.	If this occurs repeatedly, contact the Lenze service.

Internal error [0x01bc6100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error.	If this occurs repeatedly, contact the Lenze service.

Internal error [0x01bc6101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The communication module carries out an automatic software reset and reinitialises itself.	If this occurs repeatedly, contact the Lenze service.

Internal error [0x01bc6110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module defective.	If this occurs repeatedly, contact the Lenze service.

Invalid parameter set [0x01bc641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Repeat the download of the application (including module)

Error: Lenze settings loaded [0x01bc6420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set was not successful.	Repeat the download of the application (including module).

Profibus watchdog: Monitoring time elapsed [0x01bc8130]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
Permanent interruption of communication to the PROFIBUS master. Also see the chapter "Permanent interruption of PROFIBUS communication" (🕮 81).	Check cables and terminals.

Profibus: Data_Exchange state exited [0x01bc8131]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
Data exchange via PROFIBUS has been stopped. Also see the chapter "Permanent interruption of PROFIBUS communication" (🕮 81).	Check cables and terminals. The slave must receive new parameterisation and configuration files from the master in order to be able to exchange data again.

Profibus watchdog: DP-V1 MSC2 monitoring time exceeded [0x01bc8132]

Response (Lenze setting printed in bold)	**Setting:** C13880/2 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
Permanent interruption of communication to C2-PROFIBUS master. Also see the chapter "Permanent interruption of PROFIBUS communication" (🕮 81).	Check cables and terminals.

13 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the software manual and in the »Engineer« online help for Inverter Drives 8400 by the parameters of the E84AYCPM communication module (PROFIBUS).

Software manual/»Engineer« online help for Inverter Drives 8400
Here you can find general information on parameters.

13.1 Parameters of the communication module

This chapter lists the parameters of the E84AYCPM communication module (PROFIBUS) in numerically ascending order.

C13850

Parameter | Name: **C13850 | All words to master**			Data type: UNSIGNED_16 Index: 10725_d = $29E5_h$
Display of the process data words transferred from the communication module to the PROFIBUS master. In subcodes 1 ... 16, all process data words to the master are displayed. Only the process data words configured are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master**			Data type: UNSIGNED_16 Index: 10724_d = $29E4_h$
Display of the process data words transferred from the PROFIBUS master to the communication module. In subcodes 1 ... 16, all process data words to the master are displayed. Only the process data words configured are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to standard device**			Data type: UNSIGNED_16 Index: 10723_d = $29E3_h$
Display of process data words 1 ... 16 which are transferred from the communication module to the standard device. In subcodes 1 ... 16, all process data words from the communication module are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words from standard device**			Data type: UNSIGNED_16 Index: 10722_d = $29E2_h$
Display of process data words 1 ... 16 which are transferred from the standard device to the communication module. In subcodes 1 ... 16, all process data words from the standard device are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13853/1			
...			
C13853/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13860

Parameter | Name: **C13860 | Settings**			Data type: UNSIGNED_8 Index: 10715_d = $29DB_h$
Display of the current configuration data.			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13860/1			Reserved
C13860/2			Number of process data words (1 ... 16 words)
C13860/3			DRIVECOM parameter data channel • 0: Not active • 1: Active
C13860/4			Reserved
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Bus status**			Data type: UNSIGNED_16 Index: 10714_d = $29DA_h$
Bit-coded display of the current bus state. ▸ Querying the current bus status (📖 88)			
Display area (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13862

Parameter | Name: **C13862 | Bus counter**			Data type: UNSIGNED_16 Index: 10713_d = $29D9_h$
When the maximum count value of 65535 is reached, the counter starts again with 0.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13862/1			Data cycles per second
C13862/2			Total data cycles
C13862/3			Total parameterisation events
C13862/4			Total configuration events
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13863

Parameter | Name: **C13863 | Baud rate**		Data type: UNSIGNED_8 Index: 10712_d = $29D8_h$
Display of the baud rate		
Selection list (read only)		
0	12.00 Mbps	
1	6.00 Mbps	
2	3.00 Mbps	
3	1.50 Mbps	
4	500.00 kbps	
5	187.50 kbps	
6	93.75 kbps	
7	45.45 kbps	
8	19.20 kbps	
9	9.60 kbps	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13864

Parameter | Name: **C13864 | Active station address**			Data type: UNSIGNED_8 Index: 10711$_d$ = 29D7$_h$
Display of the active station address If all DIP switches 1 ... 64 are in the "OFF" position (Lenze setting), the station address set in C13899 becomes active and is displayed here after switching on. ▸ Setting the station address (📖 32)			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13865

Parameter | Name: **C13865 | Display: Most recent PRM data**	Data type: OCTET_STRING Index: 10710$_d$ = 29D6$_h$
Display of the last parameter data sent by the PROFIBUS master with the "Set-Prm" telegram (ASCII string with 24 characters)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13866

Parameter | Name: **C13866 | Display: Most recent CFG data**	Data type: OCTET_STRING Index: 10709$_d$ = 29D5$_h$
Display of the last configuration data sent by the PROFIBUS master with the "Chk-Cfg" telegram (ASCII string with 22 characters)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13867

Parameter | Name: **C13867 | Display: Most recent diagnostic data**	Data type: OCTET_STRING Index: 10708$_d$ = 29D4$_h$
Display of the last diagnostic data sent to the PROFIBUS master (ASCII string with 16 characters) ▸ Advanced diagnostic message (📖 89)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13880

Parameter | Name: **C13880 | Reaction on communication fault**			Data type: UNSIGNED_8 Index: 10695_d = $29C7_h$
Monitoring response to a communication fault on the PROFIBUS A change in the monitoring response is effective immediately. ▸ Permanent interruption of PROFIBUS communication (🕮 81)			
Selection list			
0	No response		
1	Error		
3	Quick stop by trouble		
4	Warning locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	
C13880/1	0: No response	The response set here for the "Watchdog/Data Exchange" monitoring function is executed if the bus station ... • does not receive a message from the master within the watchdog monitoring time (displayed in C13882/1) if there is an active connection. • detects that it is no longer in the "Data_Exchange" status. Please see also the notes given under C13881.	
C13880/2	0: No response	The response set here for the "DPV1 MSAC2" monitoring is executed if the bus station does not receive any "DPV1 MSAC2" message from the master within the monitoring time (displayed in C13882/2 if there is an active connection) and the MSAC2 connection is stopped by the slave. **Note:** We recommend only setting "information" as response so that no drive-relevant response is executed.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13881

Parameter | Name: **C13881 | Monitoring time: Data exchange**			Data type: UNSIGNED_16 Index: 10694_d = $29C6_h$
If the "Data Exchange" state is exited, the response parameterised under C13880/1 is carried out when the monitoring time for data exchange set here has expired. • A value of "65535" in this code deactivates the monitoring function. • A change in monitoring is effective immediately. • The value set here for the monitoring time must be smaller than the watchdog monitoring time (C13882/1). ▸ Permanent interruption of PROFIBUS communication (🕮 81)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**65535 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13882

Parameter | Name: **C13882 | Monitoring time: Watchdog**			Data type: UNSIGNED_32 Index: 10693_d = $29C5_h$
Display of the watchdog monitoring time determined by the PROFIBUS master • A change in the watchdog monitoring time is immediately effective. • Monitoring starts with the receipt of the first telegram. • When a value of "0" is displayed, the monitoring function is deactivated. ▸ Permanent interruption of PROFIBUS communication (📖 81)			
Display area (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C13882/1			Watchdog monitoring time
C13882/2			DP-V1 MSC2
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Clear process data**		Data type: UNSIGNED_8 Index: 10690_d = $29C2_h$
Selection of the process data which the inverter will process in the event of a PROFIBUS failure in order to maintain internal communication.		
Selection list (Lenze setting printed in bold)		
0	Use of most recent master PDOs	
1	**PDOs are set to the value '0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13886

Parameter | Name: **C13886 | Set ext. diagnostic bit by**		Data type: BITFIELD_8 Index: 10689_d = $29C1_h$
Bit-coded selection of the error responses in the standard device causing the external diagnostic bit ("diag bit") to be set (see PROFIBUS specification; bit 3 of byte 1 of the DP diagnostic messages). • The diagnostic bit is sent to the PROFIBUS master where it is evaluated separately. • The diagnostic bit is always set when a system error occurs. • The Lenze setting "0" means that the diagnostic bit is not set for the following error responses. • An advanced diagnostic message is always sent.		
Value is bit-coded:		
Bit 0	Error	
Bit 1	Trouble	
Bit 2	Quick stop by trouble	
Bit 3	Warning locked	
Bit 4	Warning	
Bit 5	Reserved	
Bit 6	Reserved	
Bit 7	Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13887

Parameter | Name: **C13887 | Suppress signalling diag. mess. upon**		Data type: BITFIELD_8 Index: 10688_d = $29C0_h$
Selection of the error responses not causing a diagnostic request to the PROFIBUS master. The Lenze setting "0" means that for each of the following error responses a diagnostic request is signalled.		
Value is bit-coded:		
Bit 0	Error	
Bit 1	Trouble	
Bit 2	Quick stop by trouble	
Bit 3	Warning locked	
Bit 4	Warning	
Bit 5	Reserved	
Bit 6	Reserved	
Bit 7	Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13899

Parameter | Name: **C13899 | Station address**			Data type: UNSIGNED_8 Index: 10676_d = $29B4_h$
Optional setting of the station address (instead of setting via DIP switches 1 ... 64) • The station address set here only becomes effective if the DIP switches 1 ... 64 have been set to OFF prior to mains switching. • The active station address is displayed under C13864. **Note:** A change of the station address will not be effective until the " Save parameter set" device command has been executed and another mains switching for the communication module/inverter has been performed. ▸ Setting the station address (📖 32)			
Setting range (min. value | unit | max. value)			**Lenze setting**
3		126	**3**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13900

Parameter | Name: **C13900 | Firmware product type**	Data type: VISIBLE_STRING Index: 10675_d = $29B3_h$
Display of the product type (string with a length of 8 bytes) The following identification code is displayed: "E84AFYPM".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 10674_d = $29B2_h$
Display of the compilation date of the firmware (string with a length of 20 bytes) The date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware version**	Data type: VISIBLE_STRING Index: 10673_d = $29B1_h$
Display of the firmware version (string with a length of 5 bytes) Example: "01.00"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13920

Parameter | Name: **C13920 | Display: DIP switch setting**				Data type: UNSIGNED_8 Index: 10655_d = $299F_h$
Display of the current DIP switch setting • The displayed value corresponds to the sum of the individual DIP switch values 1 ... 64. • The active station address is displayed under C13864. ▸ Setting the station address (📖 32)				
Display area **(min. value | unit | max. value)**				
0		255		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT				

13.2 Table of attributes

The table of attributes contains information required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter designation	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index by which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable #(only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ no decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible if controller inhibit is set	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13850	All words to master	10725	29E5	A	16	UNSIGNED_16	1	☑		
C13851	All words from master	10724	29E4	A	16	UNSIGNED_16	1	☑		
C13852	All words to standard device	10723	29E3	A	16	UNSIGNED_16	1	☑		
C13853	All words from standard device	10722	29E2	A	16	UNSIGNED_16	1	☑		
C13860	Settings	10715	29DB	A	4	UNSIGNED_8	1	☑		
C13861	Bus status	10714	29DA	E	1	UNSIGNED_16	1	☑		
C13862	Bus counter	10713	29D9	A	4	UNSIGNED_16	1	☑		
C13863	Baud rate	10712	29D8	E	1	UNSIGNED_8	1	☑		
C13864	Active station address	10711	29D7	E	1	UNSIGNED_8	1	☑		
C13865	Display: Most recent PRM data	10710	29D6	E	1	OCTET_STRING		☑		
C13866	Display: Most recent CFG data	10709	29D5	E	1	OCTET_STRING		☑		
C13867	Display: Most recent diagnostic data	10708	29D4	E	1	OCTET_STRING		☑		
C13880	Reaction on communication fault	10695	29C7	A	2	UNSIGNED_8	1	☑	☑	
C13881	Monitoring time: Data exchange	10694	29C6	E	1	UNSIGNED_16	1	☑	☑	
C13882	Monitoring time: Watchdog	10693	29C5	A	2	UNSIGNED_32	1	☑		
C13885	Clear process data	10690	29C2	E	1	UNSIGNED_8	1	☑	☑	
C13886	Set ext. diagnostic bit upon	10689	29C1	E	1	BITFIELD_8		☑	☑	
C13887	Suppress signalling diag. mess. upon	10688	29C0	E	1	BITFIELD_8		☑	☑	
C13899	Station address	10676	29B4	E	1	UNSIGNED_8	1	☑	☑	
C13900	Firmware product type	10675	29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware compilation date	10674	29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware version	10673	29B1	E	1	VISIBLE_STRING		☑		
C13920	Display: DIP switch setting	10655	299F	E	1	UNSIGNED_8	1	☑		

13.3 Implemented PROFIdrive objects (DP-V1)

I-918

Index | Name: **0x918 | Display of station address**			Data type: U16
Display of the station address set			
Display area (min. value | unit | max. value)			
1		126	
☑ Read access ☐ Write access			

I-963

Index | Name: **0x963 | Baud rate**		Data type: U16
Display of the PROFIBUS baud rate		
Selection list (read only)		
0	9.6 kbps	
1	19.2 kbps	
2	93.75 kbps	
3	187.5 kbps	
4	500 kbps	
6	1.5 Mbps	
7	3 Mbps	
8	6 Mbps	
9	12 Mbps	
10	31.25 kbps	
11	45.45 kbps	
☑ Read access ☐ Write access		

I-964

Index | Name: **0x964 | Device identification**		Data type: U16
Display of identification data		
Subindex	Display	**Info**
0x964/0	262	Manufacturer: Lenze
0x964/1	8400	Device type
0x964/2	xxyy	Software version, e.g. 0100 (V 01.00)
0x964/3	yyyy	Firmware date (year), e.g. 2007
0x964/4	ddmm	Firmware date (day/month), e.g. 0506 (5th June)
☑ Read access ☐ Write access		

I-974

Index | Name: **0x974 | Maximum time per DPV1 parameter access**		Data type: U16
Display of access statistics		
Subindex	Display	**Info**
0x974/0	240 bytes	Maximum block length
0x974/1	40	Maximum number of parameter accesses
0x974/2	0	Maximum time per access
☑ Read access ☐ Write access		

14 DIP switch positions for setting the station address

The station address results from the sum of the binary valencies of switches 1 ... 64.

The following table shows the switch positions for the valid address range 1 ... 126.

▸ Setting the station address (📖 32)

Station address	DIP switch						
	64	32	16	8	4	2	1
1	OFF	OFF	OFF	OFF	OFF	OFF	**ON**
2	OFF	OFF	OFF	OFF	OFF	**ON**	OFF
3	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**
4	OFF	OFF	OFF	OFF	**ON**	OFF	OFF
5	OFF	OFF	OFF	OFF	**ON**	OFF	**ON**
6	OFF	OFF	OFF	OFF	**ON**	**ON**	OFF
7	OFF	OFF	OFF	OFF	**ON**	**ON**	**ON**
8	OFF	OFF	OFF	**ON**	OFF	OFF	OFF
9	OFF	OFF	OFF	**ON**	OFF	OFF	**ON**
10	OFF	OFF	OFF	**ON**	OFF	**ON**	OFF
11	OFF	OFF	OFF	**ON**	OFF	**ON**	**ON**
12	OFF	OFF	OFF	**ON**	**ON**	OFF	OFF
13	OFF	OFF	OFF	**ON**	**ON**	OFF	**ON**
14	OFF	OFF	OFF	**ON**	**ON**	**ON**	OFF
15	OFF	OFF	OFF	**ON**	**ON**	**ON**	**ON**
16	OFF	OFF	**ON**	OFF	OFF	OFF	OFF
17	OFF	OFF	**ON**	OFF	OFF	OFF	**ON**
18	OFF	OFF	**ON**	OFF	OFF	**ON**	OFF
19	OFF	OFF	**ON**	OFF	OFF	**ON**	**ON**
20	OFF	OFF	**ON**	OFF	**ON**	OFF	OFF
21	OFF	OFF	**ON**	OFF	**ON**	OFF	**ON**
22	OFF	OFF	**ON**	OFF	**ON**	**ON**	OFF
23	OFF	OFF	**ON**	OFF	**ON**	**ON**	**ON**
24	OFF	OFF	**ON**	**ON**	OFF	OFF	OFF
25	OFF	OFF	**ON**	**ON**	OFF	OFF	**ON**
26	OFF	OFF	**ON**	**ON**	OFF	**ON**	OFF
27	OFF	OFF	**ON**	**ON**	OFF	**ON**	**ON**
28	OFF	OFF	**ON**	**ON**	**ON**	OFF	OFF
29	OFF	OFF	**ON**	**ON**	**ON**	OFF	**ON**
30	OFF	OFF	**ON**	**ON**	**ON**	**ON**	OFF
31	OFF	OFF	**ON**	**ON**	**ON**	**ON**	**ON**
32	OFF	**ON**	OFF	OFF	OFF	OFF	OFF
33	OFF	**ON**	OFF	OFF	OFF	OFF	**ON**
34	OFF	**ON**	OFF	OFF	OFF	**ON**	OFF
35	OFF	**ON**	OFF	OFF	OFF	**ON**	**ON**
36	OFF	**ON**	OFF	OFF	**ON**	OFF	OFF
37	OFF	**ON**	OFF	OFF	**ON**	OFF	**ON**

Station address	DIP switch						
	64	32	16	8	4	2	1
38	OFF	**ON**	OFF	OFF	**ON**	**ON**	OFF
39	OFF	**ON**	OFF	OFF	**ON**	**ON**	**ON**
40	OFF	**ON**	OFF	**ON**	OFF	OFF	OFF
41	OFF	**ON**	OFF	**ON**	OFF	OFF	**ON**
42	OFF	**ON**	OFF	**ON**	OFF	**ON**	OFF
43	OFF	**ON**	OFF	**ON**	OFF	**ON**	**ON**
44	OFF	**ON**	OFF	**ON**	**ON**	OFF	OFF
45	OFF	**ON**	OFF	**ON**	**ON**	OFF	**ON**
46	OFF	**ON**	OFF	**ON**	**ON**	**ON**	OFF
47	OFF	**ON**	OFF	**ON**	**ON**	**ON**	**ON**
48	OFF	**ON**	**ON**	OFF	OFF	OFF	OFF
49	OFF	**ON**	**ON**	OFF	OFF	OFF	**ON**
50	OFF	**ON**	**ON**	OFF	OFF	**ON**	OFF
51	OFF	**ON**	**ON**	OFF	OFF	**ON**	**ON**
52	OFF	**ON**	**ON**	OFF	**ON**	OFF	OFF
53	OFF	**ON**	**ON**	OFF	**ON**	OFF	**ON**
54	OFF	**ON**	**ON**	OFF	**ON**	**ON**	OFF
55	OFF	**ON**	**ON**	OFF	**ON**	**ON**	**ON**
56	OFF	**ON**	**ON**	**ON**	OFF	OFF	OFF
57	OFF	**ON**	**ON**	**ON**	OFF	OFF	**ON**
58	OFF	**ON**	**ON**	**ON**	OFF	**ON**	OFF
59	OFF	**ON**	**ON**	**ON**	OFF	**ON**	**ON**
60	OFF	**ON**	**ON**	**ON**	**ON**	OFF	OFF
61	OFF	**ON**	**ON**	**ON**	**ON**	OFF	**ON**
62	OFF	**ON**	**ON**	**ON**	**ON**	**ON**	OFF
63	OFF	**ON**	**ON**	**ON**	**ON**	**ON**	**ON**
64	**ON**	OFF	OFF	OFF	OFF	OFF	OFF
65	**ON**	OFF	OFF	OFF	OFF	OFF	**ON**
66	**ON**	OFF	OFF	OFF	OFF	**ON**	OFF
67	**ON**	OFF	OFF	OFF	OFF	**ON**	**ON**
68	**ON**	OFF	OFF	OFF	**ON**	OFF	OFF
69	**ON**	OFF	OFF	OFF	**ON**	OFF	**ON**
70	**ON**	OFF	OFF	OFF	**ON**	**ON**	OFF
71	**ON**	OFF	OFF	OFF	**ON**	**ON**	**ON**
72	**ON**	OFF	OFF	**ON**	OFF	OFF	OFF
73	**ON**	OFF	OFF	**ON**	OFF	OFF	**ON**
74	**ON**	OFF	OFF	**ON**	OFF	**ON**	OFF
75	**ON**	OFF	OFF	**ON**	OFF	**ON**	**ON**
76	**ON**	OFF	OFF	**ON**	**ON**	OFF	OFF
77	**ON**	OFF	OFF	**ON**	**ON**	OFF	**ON**
78	**ON**	OFF	OFF	**ON**	**ON**	**ON**	OFF
79	**ON**	OFF	OFF	**ON**	**ON**	**ON**	**ON**
80	**ON**	OFF	**ON**	OFF	OFF	OFF	OFF

Station address	DIP switch						
	64	**32**	**16**	**8**	**4**	**2**	**1**
81	**ON**	OFF	**ON**	OFF	OFF	OFF	**ON**
82	**ON**	OFF	**ON**	OFF	OFF	**ON**	OFF
83	**ON**	OFF	**ON**	OFF	OFF	**ON**	**ON**
84	**ON**	OFF	**ON**	OFF	**ON**	OFF	OFF
85	**ON**	OFF	**ON**	OFF	**ON**	OFF	**ON**
86	**ON**	OFF	**ON**	OFF	**ON**	**ON**	OFF
87	**ON**	OFF	**ON**	OFF	**ON**	**ON**	**ON**
88	**ON**	OFF	**ON**	**ON**	OFF	OFF	OFF
89	**ON**	OFF	**ON**	**ON**	OFF	OFF	**ON**
90	**ON**	OFF	**ON**	**ON**	OFF	**ON**	OFF
91	**ON**	OFF	**ON**	**ON**	OFF	**ON**	**ON**
92	**ON**	OFF	**ON**	**ON**	**ON**	OFF	OFF
93	**ON**	OFF	**ON**	**ON**	**ON**	OFF	**ON**
94	**ON**	OFF	**ON**	**ON**	**ON**	**ON**	OFF
95	**ON**	OFF	**ON**	**ON**	**ON**	**ON**	**ON**
96	**ON**	**ON**	OFF	OFF	OFF	OFF	OFF
97	**ON**	**ON**	OFF	OFF	OFF	OFF	**ON**
98	**ON**	**ON**	OFF	OFF	OFF	**ON**	OFF
99	**ON**	**ON**	OFF	OFF	OFF	**ON**	**ON**
100	**ON**	**ON**	OFF	OFF	**ON**	OFF	OFF
101	**ON**	**ON**	OFF	OFF	**ON**	OFF	**ON**
102	**ON**	**ON**	OFF	OFF	**ON**	**ON**	OFF
103	**ON**	**ON**	OFF	OFF	**ON**	**ON**	**ON**
104	**ON**	**ON**	OFF	**ON**	OFF	OFF	OFF
105	**ON**	**ON**	OFF	**ON**	OFF	OFF	**ON**
106	**ON**	**ON**	OFF	**ON**	OFF	**ON**	OFF
107	**ON**	**ON**	OFF	**ON**	OFF	**ON**	**ON**
108	**ON**	**ON**	OFF	**ON**	**ON**	OFF	OFF
109	**ON**	**ON**	OFF	**ON**	**ON**	OFF	**ON**
110	**ON**	**ON**	OFF	**ON**	**ON**	**ON**	OFF
111	**ON**	**ON**	OFF	**ON**	**ON**	**ON**	**ON**
112	**ON**	**ON**	**ON**	OFF	OFF	OFF	OFF
113	**ON**	**ON**	**ON**	OFF	OFF	OFF	**ON**
114	**ON**	**ON**	**ON**	OFF	OFF	**ON**	OFF
115	**ON**	**ON**	**ON**	OFF	OFF	**ON**	**ON**
116	**ON**	**ON**	**ON**	OFF	**ON**	OFF	OFF
117	**ON**	**ON**	**ON**	OFF	**ON**	OFF	**ON**
118	**ON**	**ON**	**ON**	OFF	**ON**	**ON**	OFF
119	**ON**	**ON**	**ON**	OFF	**ON**	**ON**	**ON**
120	**ON**	**ON**	**ON**	**ON**	OFF	OFF	OFF
121	**ON**	**ON**	**ON**	**ON**	OFF	OFF	**ON**
122	**ON**	**ON**	**ON**	**ON**	OFF	**ON**	OFF
123	**ON**	**ON**	**ON**	**ON**	OFF	**ON**	**ON**

Station address	DIP switch						
	64	**32**	**16**	**8**	**4**	**2**	**1**
124	**ON**	**ON**	**ON**	**ON**	**ON**	OFF	OFF
125	**ON**	**ON**	**ON**	**ON**	**ON**	OFF	**ON**
126	**ON**	**ON**	**ON**	**ON**	**ON**	**ON**	OFF

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

Lenze Drives GmbH
Postfach 10 13 52
D-31763 Hameln
Germany
+49 (0)51 54 / 82-0
+49 (0)51 54 / 82-28 00
Lenze@Lenze.de
www.Lenze.com

Service

Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
00 80 00 / 24 4 68 77 (24 h helpline)
+49 (0)51 54 / 82-11 12
Service@Lenze.de