Drive functions

Motion Control

MC 5010

MC 5005

MC 5004

MCS

WE CREATE MOTION

EN

Imprint

Version:
2nd edition, 9-12-2016

This document has been prepared with care.
Dr. Fritz Faulhaber GmbH & Co. KG cannot accept any liability for any errors in this document or for the consequences of such errors. Equally, no liability can be accepted for direct or consequential damages resulting from improper use of the equipment.

The relevant regulations regarding safety engineering and interference suppression as well as the requirements specified in this document are to be noted and followed when using the software.

Subject to change without notice.

The respective current version of this technical manual is available on FAULHABER's internet site:
www.faulhaber.com

Content

Content

Content

Content

1 About this document

1.1 Validity of this document

This document describes:

- Principle of the device controller
- Commissioning and configuring the device
- Operating modes and functions

This document is intended for use by technicians and engineers in the application of controlled electrical drives and industrial communications systems.

All data in this document relate to the standard versions of the drives. Changes relating to customer-specific versions can be found in the according data sheet.

All data in this document relate to the firmware revision G.

1.2 Associated documents

For certain actions during commissioning and operation of FAULHABER products additional information from the following manuals is useful:

Manual	Description
Motion Manager 6	Operating instructions for FAULHABER Motion Manager PC software
Quick start guide	Description of the first steps for commissioning and operation of FAULHABER Motion Controllers
Communications manual	Description of communication with the drive
Technical manual	Instructions for installation and use of the FAULHABER Motion Controller

These manuals can be downloaded in pdf format from the web page www.faulhaber.com/manuals.

1.3 Using this document

- Read the document carefully before undertaking configuration.
- Retain the document throughout the entire working life of the product.
- Keep the document accessible to the operating personnel at all times.
- Pass the document on to any subsequent owner or user of the product.

1.4 List of abbreviations

Abbreviation	Meaning
ADC	Analog-to-Digital Converter
AES	Absolute encoder
AnIn	Analogue input
APC	Analogue Position Control
ATC	Analogue Torque Control
Attr.	Attribute
AVC	Analogue Velocity Control
BL	Brushless DC Servomotor
CAN	Controller Area Network
CiA	CAN in Automation e.V.
CSP	Cyclic Synchronous Position
CST	Cyclic Synchronous Torque
CSV	Cyclic Synchronous Velocity
DC	Direct Current
DigIn	Digital input
EMF	Voltage induced in the Motor (ElectroMotive Force)
HW	Hardware
Ixx	Data Type Integer (whole numbers) with bit size xx
LM	Linear Motor
LSS	Layer Setting Service
PP	Profile Position
PV	Profile Velocity
ro	read only
rw	read-write
PWM	Pulse Width Modulation
SSI	Synchronous Serial Interface for Position Encoder
Sxx	Data type signed (negative and positive numbers) with bit size xx
TTL	Transistor Transistor Logic
Uxx	Data type unsigned (positive numbers) with bit size xx
VM	Voltage Mode
XDC	External Document Converter
XML	Extensible Markup Language

1.5 Symbols and markers

CAUTION!
Hazards due to hot surfaces. Disregard may lead to burns.

- Measures for avoidance

NOTICE!
Risk of damage.

- Measures for avoidance

Instructions for understanding or optimising the operational procedures

✓ Pre-requirement for a requested action

1. First step for a requested action
   ↳ Result of a step
2. Second step of a requested action

↳ Result of an action

- Request for a single-step action

2 Motion Controller in overview

The Motion Controller offers flexible integrated control of DC, BL and LM servomotors.

FAULHABER Series V3.0 Motion Controllers can be configured and operated via USB, RS232, CANopen or an optional EtherCAT interface. Alternatively the Motion Controller can be configured via sequence programs so that it can be controlled without any communications connections, purely using discrete inputs and outputs.

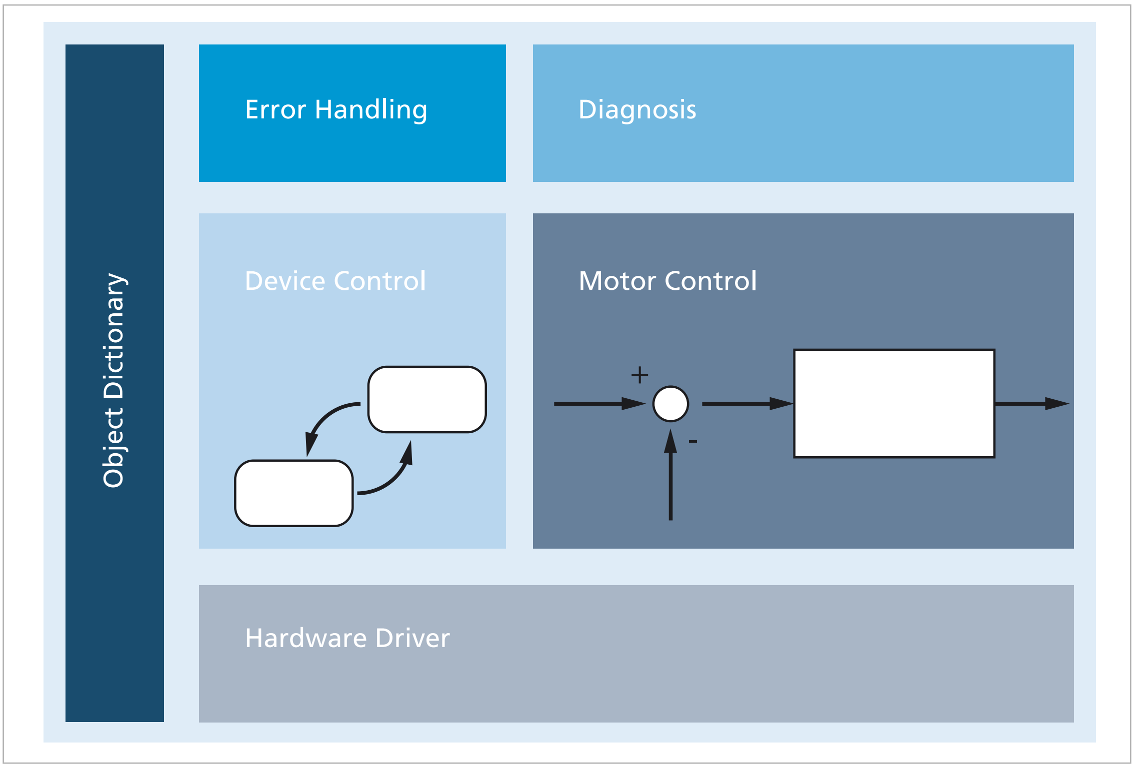


Fig. 1: Motion Controller sub-functions

The FAULHABER Motion Controller has several sub-functions:

- HW driver: The HW driver provides basic functions for accessing the connected hardware. The parameters include the type of the motor encoder or the configuration of the digital inputs.
- Device control: The device control contains the drive state machine, switches the output stage and changes the operating modes. The essential parameters are the controlword and the statusword of the drive and the operating mode.
- Controller: The controller monitors the target and actual values for actuation of the connected motor. The essential parameters are the settings for the controllers and for the profile generator.
- Device diagnostics: The device diagnostics monitor the state of the device and of the connected motor. The essential parameters are the data of the connected motor. The device state is signalled in the device statusword 0x2324.01.
- Error handling: The error handling responds configurably to the errors that are detected.
- Object Dictionary: The object dictionary gathers the parameters together with the setpoint and actual values for the application, so that they are the available to the communications system or the free procedures.

The FAULHABER Motion Manager offers convenient methods for device configuration by means of graphical dialogues. The Motion Manager 6 manual explains the handling and the commissioning using the FAULHABER Motion Manager.

Configuration can be performed either by direct programming or by other configuration tools.

The basic settings of the Motion Controller must be configured during commissioning, to adapt the controller to the connected motor.

Where integrated drive units (Motion Control Systems) are supplied, the basic configuration has already been performed in the factory. The basic configuration still has to be adapted to the application situation. The following settings often have to be adapted to the application situation:

- Operating mode
- Current limitation values
- Motor control parameters
- Function of the digital inputs/outputs

2.1 Components of the Motion Controller

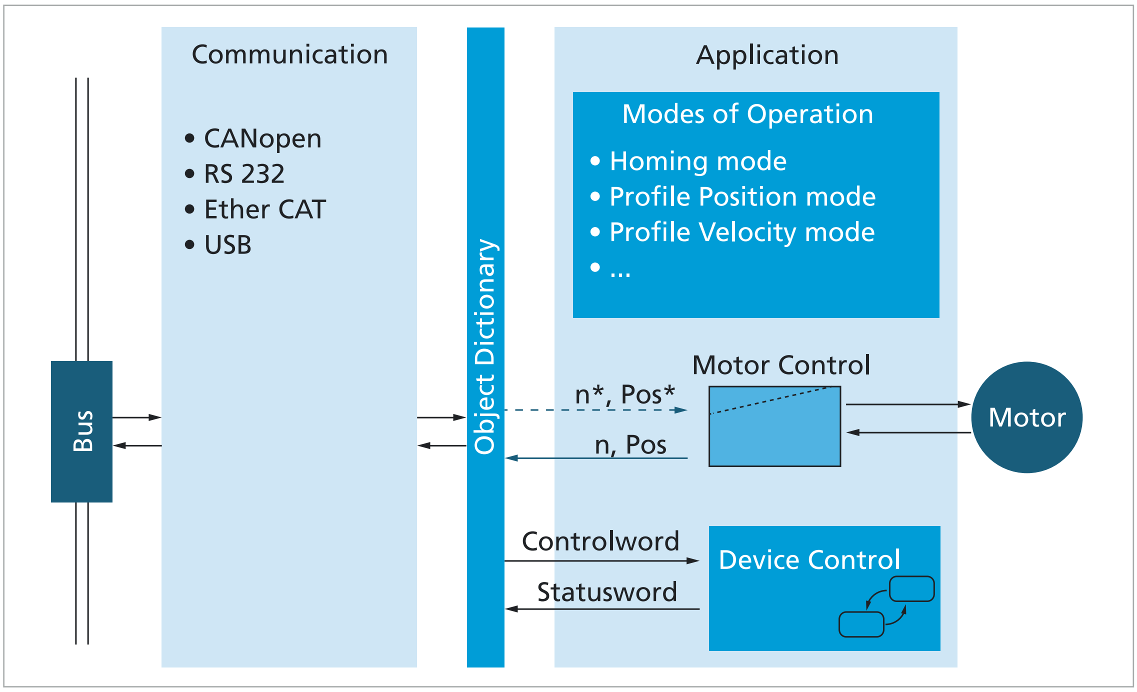


Fig. 2: Basic design of the device controller

Communication services

The master communicates via the bus system and uses the communication services and the object dictionary (see the Communications Manual).

Object Dictionary

The object dictionary contains parameters, set values- and actual values of a drive. The object dictionary is the link between the application (drive functions) and the communication services. All objects in the object dictionary can be addressed by a 16-bit index number (0x1000 to 0x6FFF) and an 8-bit subindex (0x00 to 0xFF).

Index	Assignment of the objects
0x1000 to 0x1FFF	Communication objects
0x2000 to 0x5FFF	Manufacturer-specific objects
0x6000 to 0x6FFF	Objects of the drive profile to CiA 402

The values of the parameters can be changed by the communication side or by the drive side.

Application part

The application part contains drive functions according to CiA 402. The drive functions read parameters from the object dictionary, obtain the setpoints from the object dictionary and return actual values. The parameters from the object dictionary determine the behaviour of the drive.

Tab. 1: Application to CiA 402

Controller components	Description
CiA 402 drive state machine	Represents the behaviour of the drive (see chap. 3.1, p. 14).
Controlword	Controls the behaviour of the drive (see chap. 3.2, p. 16).
Statusword	Reads the behaviour of the drive (see chap. 3.3, p. 19).

2.2 Configuring the drive - general procedure

Steps 1, 2, 3 and 9 are essential in order to commission the drive. In operating modes PP and PV, step 4 must be performed in order to set the profile generators.

FAULHABER Motion Manager 6 offers helpful commissioning wizards for steps 1-3. Appropriate graphical communications dialogues are provided for the further steps.

The other steps allow the end application to be configured.

Procedure for initial commissioning

- ✓ Have a suitable tool to hand (such as the FAULHABER Motion Manager or other configuration tools).
- ✓ The communication settings must be correct, see the Communication Manual.

1. Establishing the connection (see chap. 4.1, p. 23).
2. Configure the motor type and motor data (see chap. 4.2, p. 24).
3. Adapt the motor control parameters and current limitation values to the motor type and application (see chap. 4.3, p. 24).
4. Configure the profile generator (see chap. 4.4, p. 42).
5. Configure the error handling (see chap. 7, p. 136)
6. Configure the digital inputs and outputs (see chap. 4.6, p. 53).
7. Convert the units (see chap. 4.7, p. 67).
8. Configure the actual value source (see chap. 4.8, p. 76).
9. Configure the operating mode (see chap. 5, p. 82).

3 Design of the device controller

3.1 State machine of the drive

During the switch-on and switch-off process, the FAULHABER Motion Controller passes through a state machine with several steps. The sequence corresponds to the process defined in the CiA 402 for CANopen drives.

The transitions are controlled by the controlword (object 0x6040) of the drive.

The behaviour of the drive is represented by a state machine. The controlword controls the transitions, the statusword shows the states.

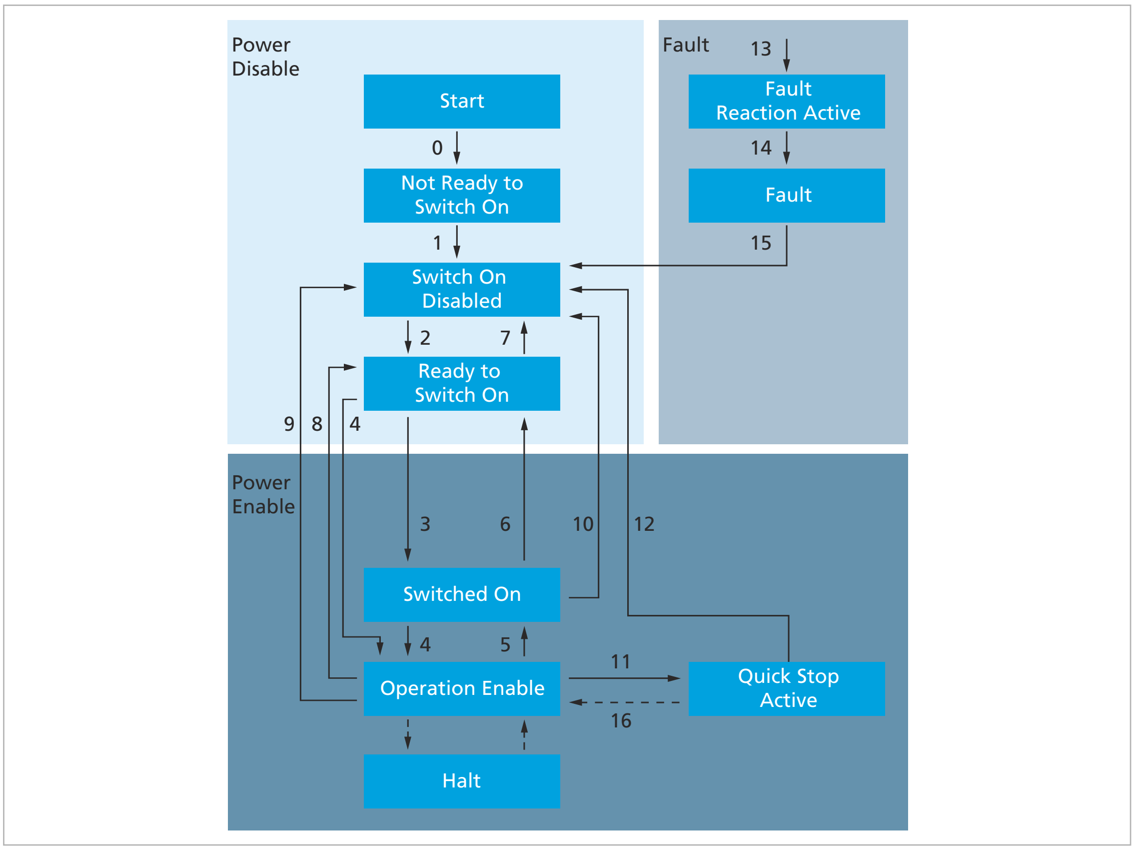


Fig. 3: State machine of the drive

Tab. 2: Command overview

Command	Transitions
Shut Down	2, 6, 8
Switch On	3
Disable Voltage	7, 9, 10, 12
Quick Stop	7, 10, 11
Disable Operation	5
Enable Operation	4, 16
Fault Reset	15

- The *Not Ready to Switch On* state is passed through automatically. The Motion Controller can be configured via the object 0x2503 so that the offsets for the current measurement are readjusted.
- After it has been switched on, the drive is in the *Switch On Disabled* state. The status LED starts to flash green.
- The **Shut Down** command brings the drive into the *Ready to Switch On* state. The option code in the object 0x605B can be used to specify whether the drive should first be brought to a controlled stop.
- The **Switch On** command switches the Motion Controller into the *Switched On* state. The *Switched On* state can be passed through automatically if in the *Ready to Switch On* state the **Enable Operation** command is given directly.
- The **Enable Operation** command brings the drive into the *Operation Enabled* state. The transition is performed only if the supply voltage is within the permissible range. If a digital output is configured for actuation of the holding brake, the holding brake is first released.
- The output stage is enabled in the *Operation Enabled* state. The status LED lights up continuously green. The behaviour of the motor control depends on the set operating mode.
- The **Disable Operation** command returns the drive to the *Switched On* state. All movement commands outstanding at this stage are cancelled. If a holding brake is configured, it is applied before the output stage is switched off. The option code in the object 0x605C can be used to specify whether the drive should first be brought to a controlled stop.
- The **Disable Voltage** command switches the output stage off directly. The motor is not braked. If a holding brake is configured, it is applied before the output stage is switched off. The drive is then in the *Switch On Disabled* state.
- The **Quick Stop** command switches the drive out of the *Operation Enabled* state and into the *Quick Stop Active* state. By means of the option code in the object 0x605A the period for which a motor that is running should remain stopped can be determined. Any outstanding movement commands are discarded when the *Quick Stop Active* state is entered. The brake is not activated if the drive remains in the *Quick Stop Active* state.
- The Halt bit in the controlword allows a drive to be stopped during the course of a movement. The current and following movement jobs are not discarded but merely suspended as long as the halt bit is set. The movement jobs are resumed as soon as the halt bit is unset.
- Sending the **Enable Operation** command again activates the drive again from the *Quick Stop Active* state. This resets the target value, and the position previously attained is retained.
- In response to detection of an error the drive can switch from any operating state into the *Fault* state. The option code in the object 0x605E can be used to specify how a motor that is still running can then be brought to a standstill. After this the output stage will be switched off and a configured holding brake is applied.

3.2 Controlword

The commands for performing a change of state are defined by combinations of bits 0–3 in the controlword. The controlword is located in the object dictionary under index 0x6040.

Controlword

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6040	0x00	Controlword	U16	rw	–	Drive control

Tab. 3: Overview of the bits of the controlword and combination permutations of bits 0-3

Bit	Function	Commands for the device control state machine						
		Shut Down	Switch On	Disable Voltage	Quick Stop	Disable Opera-tion	Enable Opera-tion	Fault Reset
0	Switch On	0	1	X	X	1	1	X
1	Enable Voltage	1	1	0	1	1	1	X
2	Quick Stop	1	1	X	0	1	1	X
3	Enable Operation	X	0	X	X	0	1	X
4	Operation Mode Specific							
5	Operation Mode Specific							
6	Operation Mode Specific							
7	Fault Reset							0 → 1
8	Halt							
9	Change on setpoint value (only in Profile Position mode)							
10	not used							
11	not used							
12	not used							
13	not used							
14	not used							
15	not used							

1 = bit set

0 = bit set

0 → 1 = rising flank, change from 0 to 1

X = bit not used for this command (status irrelevant)

Tab. 4: Meaning of the bits in the controlword

Bit	Function	Description
0	Switch On	0: No voltage present 1: Power supply being activated
1	Enable Voltage	0: Drive switched off 1: Drive ready
2	Quick Stop	0: Quick Stop enabled 1: Quick Stop disabled
3	Enable Operation	0: Operation disabled 1: Operation enabled
7	Fault Reset	0 → 1: Fault Reset
8	Halt	0: Movement can be performed 1: Stop drive

3.2.1 Examples of command sequences

The command sequences for controlling the state machine are explained in the following sections.

3.2.1.1 Enable Operation

Sequence of steps of the transition to bring a drive into the *Operation Enabled* state.

✓ The drive is in the *Switch On Disabled* state.

1. Enter the Shut Down command (controlword = 0x00 06).
   - ↳ The drive switches into the state *Ready to Switch On*.
2. Enter the Switch On command (controlword = 0x00 07).
   - ↳ The drive switches into the *Switched On* state.
3. Enter the Enable Operation command (controlword = 0x00 0F).

↳ The drive is now in the *Operation Enabled* state. In this state the set operating mode can be used, using the respective objects.

3.2.1.2 Resetting the fault state

Sequence of steps of the transition to bring a drive out of the fault state.

1. Enter the Fault Reset command (controlword = 0x00 08).
   - ↳ The drive switches into the state *Switch On Disabled*.
2. Enter the Shut Down command (controlword = 0x00 06).
   - ↳ The drive switches into the state *Ready to Switch On*.
3. Enter the Enable Operation command (controlword = 0x00 0F).

↳ The drive is now in the *Operation Enabled* state. In this state the set operating mode can be used, using the respective objects.

The current state of the state machine of the drive (see Fig. 3) can be read from bits 0 to 6 of the statusword.

Only transitions defined in the current states can be performed. Therefore before a change of state, the evaluation of the statusword must be checked in order to determine the state of the drive.

3.2.2 Actuation of a holding brake

Using the object 0x2312.02, one of the digital outputs can be defined as a control port for a holding brake. During the transition into the *Operation Enabled* state, the holding brake is released and reactuated before the output stage is switched off again.

The delay time to be applied is set via the object 0x2302.03.

3.3 Statusword

The current state of the drive is represented in bits 0–6 of the statusword. The statusword is located in the object dictionary under index 0x6041.

Statusword

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6041	0x00	Statusword	U 16	ro	–	Status display

Tab. 5: Overview of the bits of the statusword and combination permutations of bits 0-6

Bit	Function	State of the device control state machine							
		Not Ready to Switch On	Switch On Disabled	Ready to Switch On	Switched On	Opera-tion Enabled	Quick Stop Active	Fault Reaction Active	Fault
0	Ready to Switch On	0	0	1	1	1	1	1	0
1	Switched on	0	0	0	1	1	1	1	0
2	Operation Enabled	0	0	0	0	1	1	1	0
3	Fault	0	0	0	0	0	0	1	1
4	*Voltage enabled	X	X	X	X	X	X	X	X
5	Quick Stop	X	X	1	1	1	0	X	X
6	Switch On Disabled	0	0	0	0	0	0	0	0
7	Warning								
8	0								
9	Not used								
10	Operation mode specific (see chap. 5, p. 82)								
11	Internal limit active								
12	Operation mode specific (see chap. 5, p. 82)								
13	Operation mode specific (see chap. 5, p. 82)								
14	Configurable								
15	Configurable								

1 = bit set

0 = bit set

X = bit not used for this command (status irrelevant)

* = not implemented. FAULHABER Motion Controllers have no switch for the power supply

Tab. 6: Meaning of the bits in the statusword

Bit	Function	Description
0	Ready to Switch On	0: Not ready to switch on 1: Ready to switch on
1	Switched On	0: No voltage present 1: The drive is in the *Switched On* state
2	Operation Enabled	0: Operation disabled 1: Operation enabled
3	Fault	0: No fault present 1: Fault present
4	Voltage Enabled [a)]	0: Power supply disabled 1: Power supply enabled
5	Quick Stop	0: Quick Stop disabled 1: Quick Stop enabled
6	Switch On Disabled	0: Switch On enabled 1: Switch On disabled
7	Warning	0: No raised temperatures 1: One of the monitored temperatures has exceeded the warning threshold or more.
8	0	Not used
9	Remote	Not used
10	Operation Mode Specific	See the respective operating mode
11	Internal Limit Active	0: Internal range limit not reached 1: Internal range limit e.g. limit switch reached
12	Operation Mode Specific	See the respective operating mode
13	Operation Mode Specific	See the respective operating mode
14	Configurable	The object 0x233A.01 allows configuration of which combination of states should be shown in the object 0x2324.01 (device statusword) (see chap. 7, p. 136)
15	Configurable	The object 0x233A.02 allows configuration of which combination of states should be shown in the object 0x2324.01 (device statusword) (see chap. 7, p. 136).

a) *FAULHABER Motion Controllers and Motion Control Systems are powered directly by a DC power supply. Bit 4 has thus no meaning.*

3.4 Drive stop at change of state

When the drive exits the *Operation Enabled* state it can be required that it is brought to a stop before the output stage is switched off. Possible causes of a change of state are:

- The drive is stopped by the ***Quick Stop*** command, the control system can however remain active.
- The drive is stopped by the ***Shutdown***, ***Disable Voltage*** or ***Disable Operation*** command.
- The drive switches into the *Fault* state as a consequence of detecting a fault.
- The stop options can be set via the object 0x605D in accordance with Tab. 7

Tab. 7: Options for stopping the drive at changes of state

	Braking procedure	Quick Stop (0x605A)	Shut Down (0x605B)	Disable Operation (0x605C)	Fault (0x605E)
0	Deactivate directly	x	x	x	x
1	Brake ramp + switch off	x	x	x	x
2	Quick stop ramp + switch off	x	–	–	x
3	Stop with maximum braking current	x	–	–	x
4	Stop with U = 0 + shut off	x	–	–	x
5	Brake ramp + maintain state	x	–	–	–
6	Quick stop ramp + maintain state	x	–	–	–
7	Stop with I* = 0 + maintain state	x	–	–	–
8	Stop with U = 0 + maintain state	x	–	–	–

The drive stops if the Halt bit of the controlword is set during operation.

If a holding brake is configured, it is activated before the controller is deactivated.

Meaning of the Halt bit (0x605D):

- 1: Brake ramp + maintain state
- 2: Quick stop ramp + maintain state
- 3: Stop with maximum braking current
- 4: Stop with U = 0 + maintain state

When the commands ***Quick Stop***, ***Shut Down***, ***Disable Voltage*** and ***Disable Operation*** are issued, and also during fault handling, any outstanding movement commands are cancelled. When the drive is subsequently reactivated, the drive resumes movement only once a new setpoint value has been input.

If the drive was stopped by a Halt bit, the drive immediately resumes the movement it was previously performing.

3.5 Behaviour at the limits of the movement range

3.5.1 Limit switch

The digital inputs to the Motion Controller can be configured for evaluation of the limit switches (see chap. 4.6.1, p. 54).

The drive will be stopped if it reaches a limit switch during operation. The options for stopping the drive are the same as those available when the drive is stopped due to a change of state (see chap. 3.4, p. 21). The configuration is performed via the object 0x2310.03.

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2310	0x03	Limit Switch Option Code	I8	rw	1	0: - 1: Brake ramp 2: Quick stop 3: Stop at max. current 4: Stop at max. current

The drive is stopped and the velocity is then controlled to = 0.

3.5.2 Software Position Limits

Via the object 0x606D the limits of the movement range can be configured irrespective of the limit switches.

Target positions via the object 0x607A are always limited to this value range, even in cases of relative positioning in the Profile Position (PP) operating mode, no target values can be set outside the movement range thereby determined.

Endless positioning, for instance for cyclical conveying device is available in the Profile Position (PP) operating mode. For this purpose the Position Range Limit object (0x606B) must be used to specify the numeric range of the position values more precisely then the range given by the software position limits.

In velocity-controlled mode, software position limits can be treated like limit switches. If the upper or lower position limit is violated, the drive is brought to a stop via the ramp defined in the object 0x2310.03.

The reaction to reaching a software position limit can be set via the object 0x233F

- Bit 1 = 0: Software position limits have no effect except in positioning mode.
- Bit 1 = 1: When not in positioning mode, software position limits are evaluated like limit switches.

4 Configuring the drive and starting it

NOTICE!

Disregarding the basic settings can cause damage to components.

- Comply with the description of the basic settings.

The steps described below are applicable when the Motion Manager is being used.

4.1 Establishing the connection

1. Connect the drive to the Motion Controller.
2. Connect the power supply to the Motion Controller.
3. Set the communications interface as specified in the Motion Manager manual and the respective communications manual.

Depending on the selected communications interface, it may be necessary to set the Baud rate and the node number. If other configuration tools are used, the following settings must be performed:

- CANopen: The node number and Baud rate are set via the LSS protocol. This can be done using the Motion Manager or any CANopen configuration program (see the CANopen communications manual).
- RS232: The node number and Baud rate are set via the objects 0x2400.02 (Baud rate) and 0x2400.03 (node number). The object 0x2400.05 allows a network mode for the RS232 interface to be activated, so that several Motion Controllers can be operated using a single RS232 interface (see the RS232/USB communications manual).
- USB: The node number of the Motion Controller can be set via the object 0x2400.03 (see the RS232/USB communications manual).

4.2 Setting the motor type

✓ A connection to the Motion Controller must be established.

1. Use the Motion Manager motor wizard to set the connected motor.
2. Use the Motion Manager motor wizard to set the connected sensor system.

i If you are using your own programming tool or other configuration tool, the following objects must be set:

- Motor data (0x2329 and 0x232A)
- Type of commutation or voltage output (0x2340.01)
- Position encoder used:
  - Incremental encoders (0x2315 or 0x2316, see chap. 4.6.4, p. 58)
  - Analogue Hall sensors (0x2318, see chap. 4.6.4, p. 58)
  - Analogue actual value encoder (0x2313, see chap. 4.6.4, p. 58)
- The sensor system used for control (0x2330, see chap. 4.8, p. 76)
- Basic parametrisation of the controller used (0x2342, 0x2343, 0x2344 and 0x2348)

4.3 Configuration of the control parameters and current limitation values

The motor controller ensures that the required setpoint values are maintained. This is done by comparing the setpoints and actual values, and adjusting the operation accordingly. The factor group is used to convert internal position values or velocities into user-defined scaling.

Actual values may be generated by:

- Analogue Hall sensors
- Digital Hall sensors
- Incremental encoders
- Absolute encoders
- Tacho generators

Setpoints may be taken from:

- Setpoint value objects in the object dictionary
- Analogue inputs
- PWM input
- Target position as quadrature or pulsed/directional signal

4.3.1 Control cascade

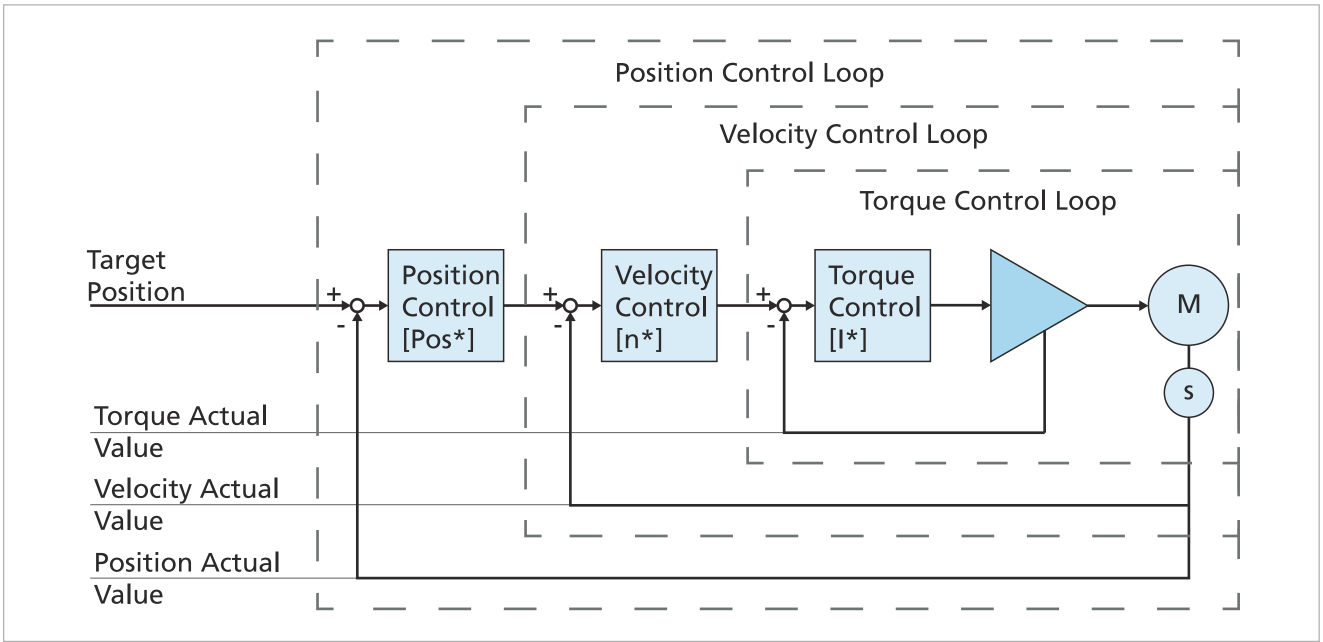


Fig. 4: Control cascade

The following control loops are installed as a cascade structure in the Motion Controller (see Fig. 4):

- Control loop for torque control

  The innermost control loop controls the torque by means of the motor current (torque controller).

- Control loop for velocity control

  The velocity controller is the middle control loop and, depending on the control deviation of the velocity, calls for a torque target, which the subordinate torque controller sets.

- Control loop for position control

  The position controller is the outermost control loop and, depending on the control deviation of the position, calls for a velocity target, which the subordinate velocity controller sets.

The advantage of the cascade structure is the separate commissioning of each stage. Target value limitations can be set directly within each stage.

i For optimisation of the controller, the control loops must be set up, starting with the innermost (torque controller) and proceeding to the outermost (position controller). Various different optimisations are available, depending on the target of the controller.

Targets of the control process

- Constant torque or constant force
- High consistency (constant motor speed)
- Smooth running of the motor (low-noise)
- High dynamic response when the setpoint value changes
- High dynamic response to interference
- High positioning accuracy
- Achievement of the target position without overswing

Not all the aspects of target control parameters can be achieved with any given group of control settings. Instructions for optimising the control parameters can be found in the chapters below for the respective controllers.

Behaviour of the controlled drive in the various operating modes

In operating modes CSP, CSV and CST the target values for position, velocity and torque are output cyclically by a supervisory controller and are loaded cyclically directly to the local controller. The supervisory controller determines the necessary intermediate values (interpolation) and coordinates the movement with the other drives of the system.

In operating modes PP and PV, the profile generator in the Motion Controller uses the target values for the position and velocity and the limit values for the acceleration and velocity to autonomously calculate a movement profile. This directly ensures compliance with the following values in the drive:

- Limits of the acceleration or deceleration under braking
- The maximum permissible velocity

In operating modes APC, AVC and ATC the setpoints for the controller are set by means of discrete inputs such as an analogue input.

4.3.2 Supported motor range

The control systems implemented in the FAULHABER Motion Controllers are optimised for operation of FAULHABER DC, BL-servo and linear motors.

All the supported motors can be called up directly in the Motion Manager motor selection wizard.

In the following cases, motors from other companies can also be operated using the Motion Controller:

- The motor must have a suitable speed sensor and/or position sensor system.
- The value range for the electric motor characteristics must be comparable to the motors from the FAULHABER portfolio.

When operating a motor from another company using the Motion Manager, the motor must be added to the motor selection wizard (see the Motion Manager manual).

NOTICE!
Operating a motor with incorrectly set control systems can damage the motor or the Motion Controller.

- Make sure that the motor control settings are correct.

4.3.3 Torque controller

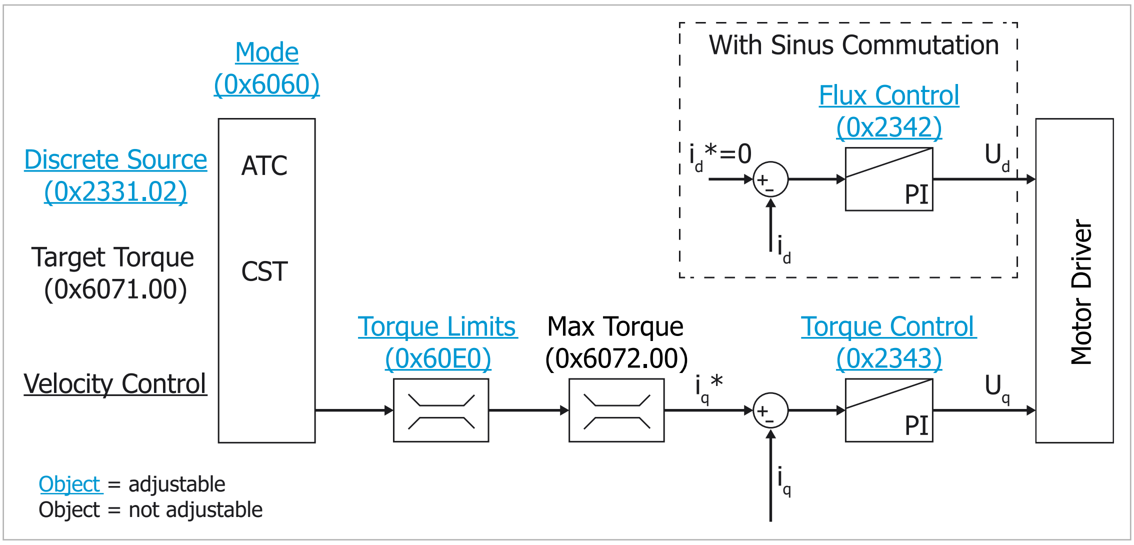


Fig. 5: Motion Manager view of the torque controller

For DC motors, the torque controller controls the motor current. For BL motors with sinusoidal commutation the torque-generating part of the current I_q is in phase with the EMF of the motor, and the field-generating part of the current I_d is controlled separately, in phase with the magnetic field of the rotor. For BL motors with block commutation, the amplitude of the motor current is controlled.

For DC motors and for BL motors with block commutation, the output value of the current control is the value of the motor voltage. For BL motors with sinusoidal commutation, the output value of the current control is the value and phase of the motor voltage.

4.3.3.1 Configuration

Torque controller

The torque controller is implemented as a PI controller for the motor current or for the torque-generating motor current component I_q.

The relevant parameters are the controller reset time $T_{N,I}$ in the object 0x2343.02 and the controller gain $K_{P,I}$ in the object 0x2342.01.

Tab. 8: Torque control parameter set

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2342	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Gain $K_{P,I}$	U32	rw	a)	Controller gain [mOhm]
	0x02	Integral Time $T_{N,I}$	U16	rw	a)	Controller reset time [µs], Range: 150–600 µs

a) *Motor-specific, is set by the motor selection wizard.*

The Motion Manager motor selection wizard sets the parameters of the torque controller to values optimised for the electrical characteristics of the connected motor.

Field controller

For BL motors with sinusoidal commutation the part of the current I_d that is in phase with the magnetic field of the rotor is controlled separately. The settings of the controller can be found in the object 0x2343 and are generally similar to those for the torque controller.

Tab. 9: Flux control parameter set

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2343	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Gain $K_{P,I}$	U32	rw	a)	Controller gain [mOhm]
	0x02	Integral Time $T_{N,I}$	U16	rw	a)	Controller reset time [µs], Range: 150–600 µs

a) *Motor-specific, is set by the motor selection wizard.*

Setpoint values

When setting the setpoint values, take into account the following points:

- In CST operating mode the setpoint values for the torque controller are determined directly via the communications system. In ATC operating mode the setpoint values are input via a discrete source, such as an analogue input (see chap. 4.6, p. 53 or chap. 4.8, p. 76).
- In operating modes with an active velocity controller, the target torque is determined by the velocity controller.
- The control is performed by the related variables. A setpoint value of 1000 corresponds to the rated torque of the connected motor.
- The setpoint value of the field-generating part of the current is generally 0, since for small motors with air-gap windings no field-weakening is available.
- A setpoint ≠ 0 is required for the field controller if the motor power supply voltage exceeds the set limit. Short-term peak currents can be dissipated by this means without affecting the dynamics of the motor.

Actual value

The torque controller controls the motor current by comparing the actual value to the set-point value. The actual value is measured in the device as the motor current.

The best motor control results are achieved when the motor rated current is greater than 30% of the device rated current (see Tab. 10).

Tab. 10: Example of operation of a 3564K024 B motor with 2.5 A rated current

Motion Controller	Device continu-ous current	Suitability
MC 5010	10 A	Possible
MC 5005	5 A	Recommended
MC 5004	4 A	Recommended

Limits

The setpoint values of the torque controller can be limited using the objects 0x60E0 (Positive Torque Limit Value) and 0x60E1 (Negative Torque Limit Value). In addition the setpoint value is initially limited to the set peak current. At higher loads on the motor and consequently higher winding temperatures, the setpoint value is limited to the set continuous current.

The continuous current and peak current of the motors are set by the Motion Manager during commissioning on the basis of the motor data sheet values. Depending on the application these values can or must be configured (see chap. 6.1, p. 133).

Tab. 11: Positive torque limit value

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60E0	0x00	Positive torque limit value	U16	rw	6000	Upper limit value in relative scaling [a]

a) 1000 = motor rated torque

Tab. 12: Negative torque limit value

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60E1	0x00	Negative torque limit value	U16	rw	6000	Lower limit value in relative scaling [a]

a) 1000 = motor rated torque

Optimising the control system

The Motion Manager commissioning wizard has already pre-set the current controller for the ongoing tasks. The tools available in the Motion Manager permit manual optimisation.

For manual optimisation of the current controller, apply setpoint value jumps to the current controller with the motor stopped and braked, and adjust the controller gain accordingly (see Fig. 6).

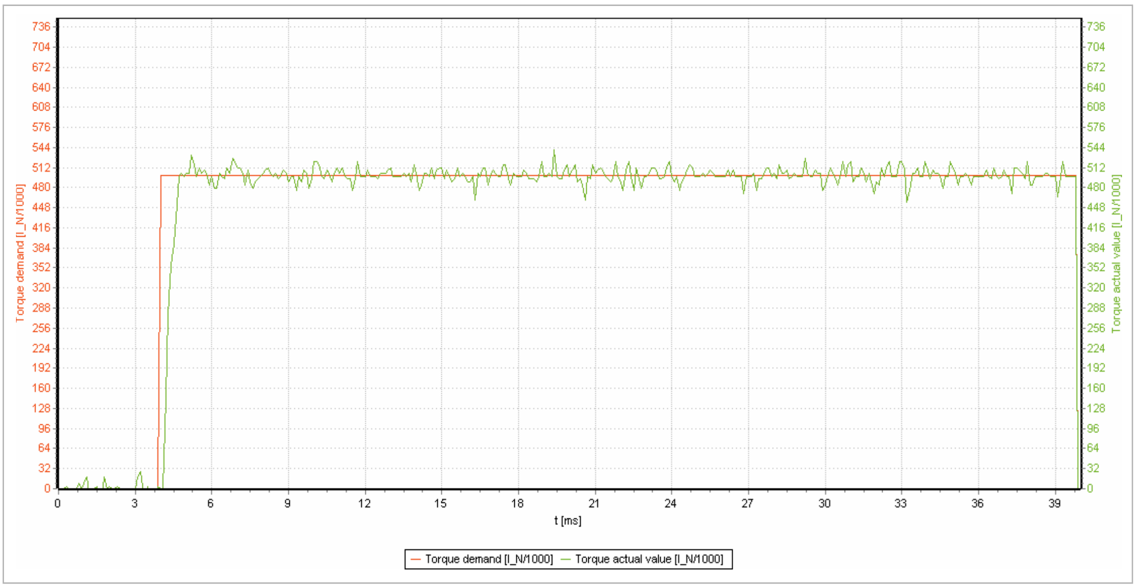


Fig. 6: Setpoint jump at the torque controller

4.3.4 Velocity controller

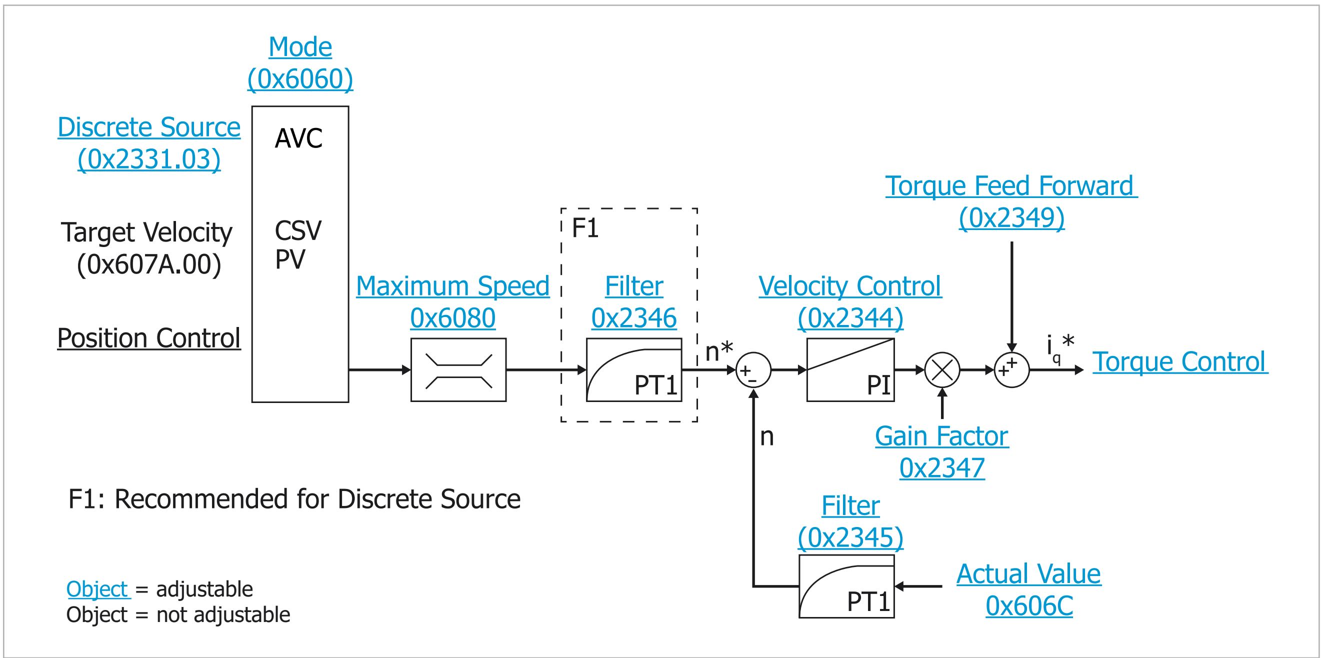


Fig. 7: Motion Manager view of the velocity controller

The velocity controller uses the torque controller which has already been set and optimised as necessary. The variation of the control deviation over time is used to determine the torque required for the balancing of target values and actual values. The subsidiary torque controller provides the required torque if no limitations are active.

The parameters of the velocity controller depend on the load which the motor has to drive:

- Mass inertia or the mass of the load that is moved
- Mass inertia of the motor
- Resilience of the coupling between the motor and the driven load

4.3.4.1 Configuration

Velocity controller

The velocity controller is implemented within the Motion Controller as a PI controller. The parameters are the controller reset time $T_{N,n}$ in the object 0x2344.02 and the controller gain $K_{P,n}$ in the object 0x2344.01.

Tab. 13: Velocity Control Parameter Set

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2344	0x00	Number of entries	U8	ro	4	Number of object entries
	0x01	Gain K_P	U32	rw	a)	Controller Gain [As 1e^{-6}]
	0x02	Integral time TN	U16	rw	a)	Controller reset time [100 µs]
	0x03	Velocity Deviation Threshold	U16	rw	65535	Maximum permissible control deviation
	0x04	Velocity Deviation Time	U16	rw	100	Maximal permissible duration of a control deviation outside the corridor
	0x05	Velocity Warning Threshold	U32	rw	30000	Warning threshold for the velocity, see 0x2324.01 bit 21

a) Motor-specific, is set by the motor selection wizard.

If a FAULHABER motor is selected in the Motion Manager motor selection wizard, the pre-set controller settings for no-load operation are loaded.

The controller configuration wizard also permits the controller to be adapted to a moving load.

Setpoint values

When setting the setpoint values, take into account the following points:

- In operating modes CSV and PV the setpoint values for the velocity controller are input directly via the communications system. In operating mode AVC the setpoints are set via a discrete source, such as an analogue input (see chap. 4.6, p. 53 and chap. 4.8, p. 76).
- In operating modes with an active position controller, the target velocity is determined by the position controller.
- The setpoint value and actual value of the control system are expressed with 32 bit resolution. Internally, the control system operates with revolutions per minute (min^{-1}). The lowest value 8 bits are used as values after the decimal point. This allows precise control even at very low velocities.

Actual value

The velocity actual value can be determined by various different sensors (see chap. 4.8, p. 76). If Hall sensors or an encoder are used, the velocity actual value is determined internally. If the actual velocity is determined via a freely configurable input, such as an analogue input, the conversion of the input value into a velocity must be set up manually.

If a drive system has not only a motor-mounted sensor but also a load-mounted sensor (such as an incremental encoder) on the gearbox driving shaft, the velocity actual value must be determined using the motor-mounted sensor. Control of the position can be based on the additional load-mounted sensor.

Limits

The target velocities in the controller are limited by the maximum motor velocity set in the object 0x6080. Where the operating mode has an active profile generator, the setpoint value are additionally limited by the maximum profile velocity (see chap. 4.4, p. 42).

Tab. 14: Maximum Motor Speed

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6080	0x00	Maximum Motor Speed	U32	rw	32767	Maximum velocity of the motor in user-defined scaling

Optimising the control system

The Motion Manager control configuration wizard allows the parameters of the controller to be adapted to the control task. In order to achieve this, either the inertia factor K_J can be entered manually or the parameters for the range can be determined via an identification procedure.

$$K_J = \frac{J_M + J_L}{J_M}$$

The stated inertia factor K_J allows the control gain and the filter time for the velocity actual value to be determined. A rigid coupling to the load is assumed. If a resilient coupling is used or play is present (for instance if a drive belt or a gearbox is used) the control gain must be reduced if necessary.

Dynamically configured controllers can be set up to an inertia factor of about 4. If the inertia factor $K_j > 4$ a highly dynamic controller is affected by the setting limits.

If the standard control parameters are used for inertia factors > 10 the drive will be noticeably noisier, since even minor displacements of the actual velocity value will lead to a significant control intervention.

CAUTION!

Hazards due to hot surfaces.

Because of the higher demands on the controller at inertia factors > 10 the heat generated by the drive will increase.

- Ensure that the drive is adequately cooled.
- Do not touch the drive without protective clothing.

At inertia factors > 10 it is possible that the rated torque of the drive can no longer be achieved. Because of the rise in temperature, the thermal protection mechanisms come into effect (see chap. 6, p. 133).

For the motor to run very quietly, especially at high values of the inertia factor K_j, it may be necessary to increase the time constant of the actual value filter (0x2345). The control reset time (0x2344.02) must be raised proportionately and the control gain (0x2344.01) reduced.

At high values of the inertia factor K_j the filter time (0x2346) of the velocity actual value must be increased in proportion to the square root of K_J compared to the no-load motor.

The Motion Manager commissioning wizard has already pre-set the velocity controller for commonly-set tasks. The controller tuning tool is available in the Motion Manager for optimisation of the control parameters.

For manual optimisation of the velocity controller, apply setpoint value jumps to the controller and adjust the control gain accordingly (see Fig. 8 or Fig. 9).

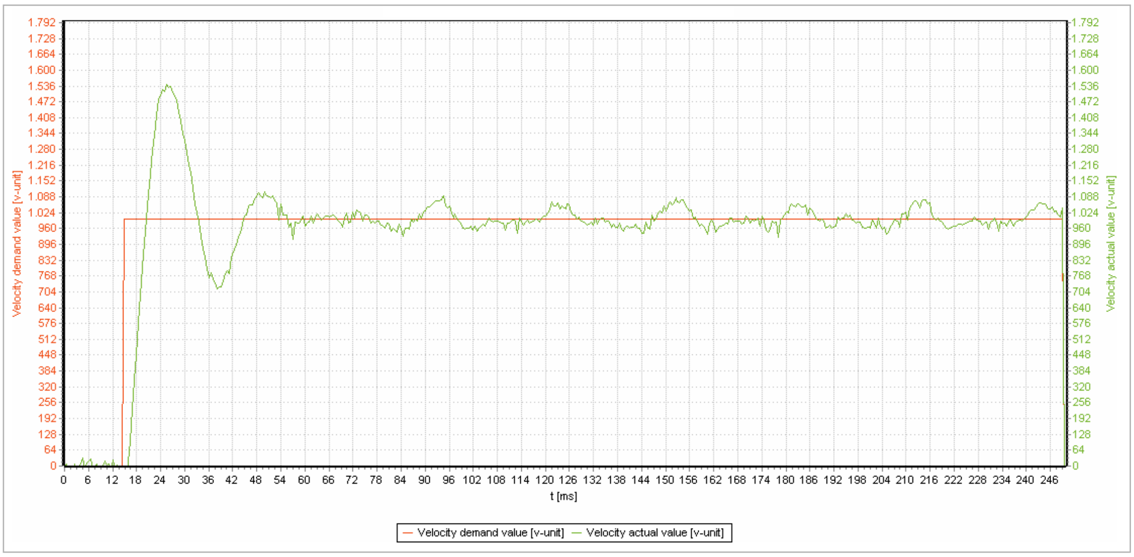


Fig. 8: Setpoint value jump during speed control

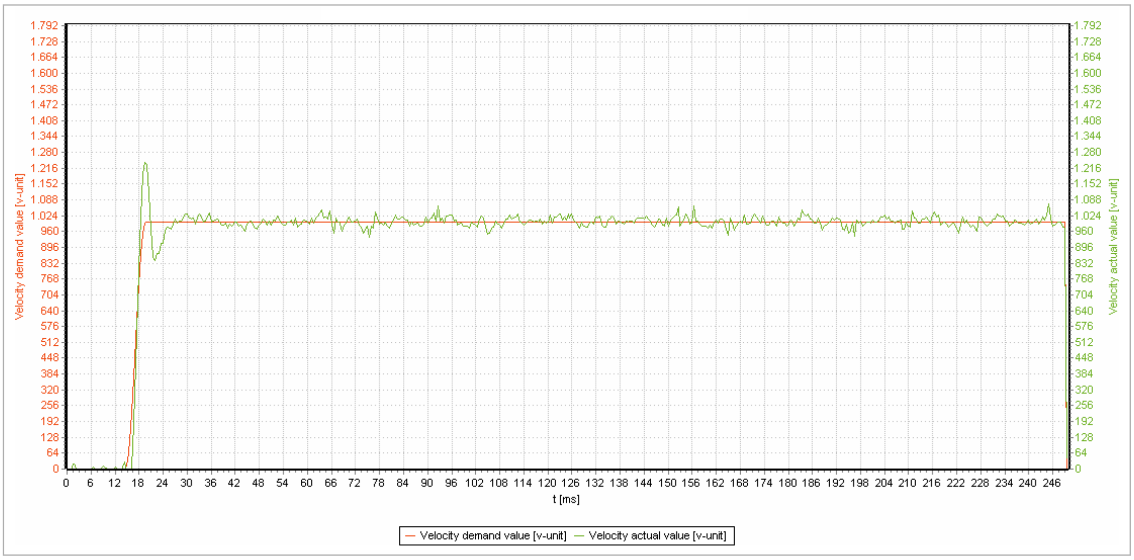


Fig. 9: Setpoint value jump during optimised speed control

4.3.4.2 Monitoring

Two monitors for the velocity are active in the velocity controller. These monitor whether the drive is stationary (n=0) and whether in profile velocity mode the drive has reached the target velocity.

The parameters set for monitoring are the velocity corridor and the minimum residence time in the corridor.

Velocity Window (0x606D)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606D	0x00	Velocity Window	U16	rw	32	Corridor around the setpoint value velocity [in user-defined scaling]

Velocity Window Time(0x606E)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606E	0x00	Velocity Window Time	U16	rw	48	Minimum residence time within the corridor in ms

Velocity Threshold (0x606F)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606F	0x00	Velocity Threshold	U16	rw	32	Corridor at n = 0

Velocity Threshold Time (0x6070)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6070	0x00	Velocity Threshold Time	U16	rw	48	Monitoring time [ms]. If the velocity lies outside the corridor for longer than is listed here, the velocity is reported as not equal to 0.

The control deviation of the velocity controller is also monitored, in addition to the monitoring of the velocity corridor and the minimum residence time within the corridor (objects 0x2344.03 and 0x2344.04).

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2344	0x00	Number of entries	U8	ro	4	Number of object entries
	0x01	Gain K_P	U32	rw	a)	Controller Gain [As $1e^{-6}$]
	0x02	Integral time TN	U16	rw	a)	Controller reset time [100 µs]
	0x03	Velocity Deviation Threshold	U16	rw	65535	Maximum permissible control deviation
	0x04	Velocity Deviation Time	U16	rw	100	Maximal permissible duration of a control deviation outside the corridor
	0x05	Velocity Warning Threshold	U32	rw	30000	Warning threshold for the velocity, see 0x2324.01 bit 21

a) *Motor-specific, is set by the motor selection wizard.*

4.3.4.3 Filter settings

Actual value filter (0x2345)

The speed controller uses a configurable actual value filter for the actual speed. The filter time can be configured to the application:

- It the quality and resolution of the sensor system is high, the filter time can be reduced.
- If only a rough resolution of the speed information is available (for instance when using digital Hall sensors or incremental sensors of low resolution), the filter time must be increased.
- The filter time should be increased if large masses or high moments of inertia have to be controlled, since otherwise small changes in the actual speed of the motor can lead to large control variations at the motor.

Tab. 15: Velocity Filter Parameter Set

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2345	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Actual Velocity Filter T_F	U16	rw	a)	Filter time T_F [100 µs]
	0x02	Display Velocity Filter	U16	rw	20	Filter time for displaying the actual velocity [100 µs]

a) *Motor-specific, is set by the motor selection wizard.*

Setpoint value filter (0x2346)

The setpoint value filter damps abrupt changes of the target velocities. This reduces the overshoot amplitude of the speed controller. To do this set the filter time of the setpoint value filter to a value identical to the reset time of the velocity controller.

Tab. 16: Set Point Velocity Filter Parameter Set

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2346	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Setpoint Velocity Filter T_F	U16	rw	a)	Filter time T_F [100 µs]
	0x02	Setpoint Filter Enable	U8	rw	0	0: inactive 1: active

a) *Motor-specific, is set by the motor selection wizard.*

4.3.5 Position controller

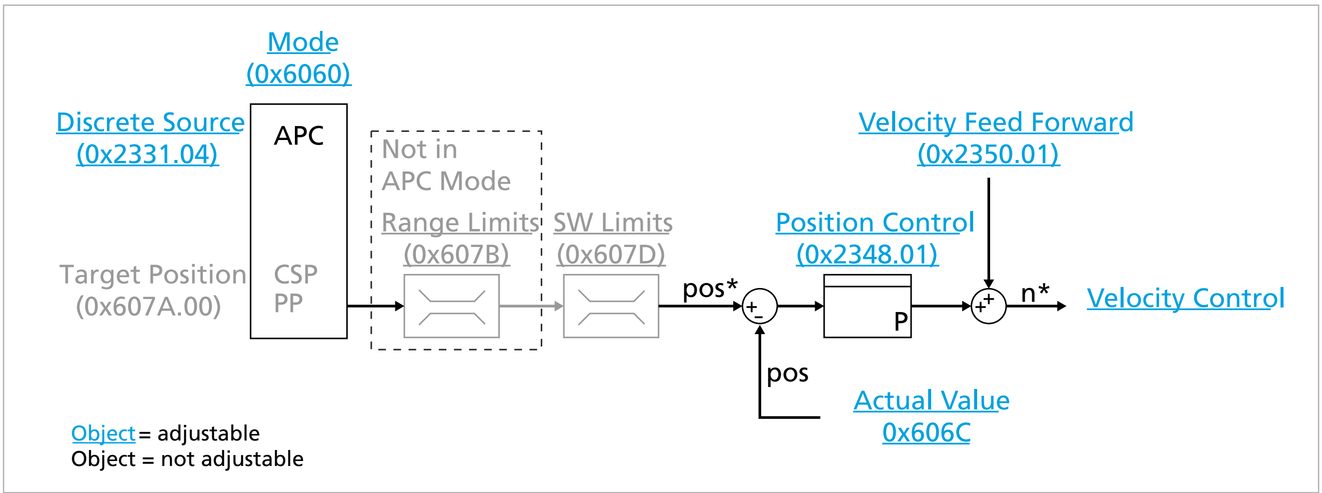


Fig. 10: Motion Manager view of the position controller

The position controller represents the outermost control loop in the Motion Controller. The path for the movement is calculated from the comparison of the position setpoint values and actual values, and the velocity is specified from this:

$$\text{Target Velocity} = \text{Controller Gain} * (\text{Target Position} - \text{Position Actual Value})$$

4.3.5.1 Configuration

The position controller is implemented as a P-controller. The sole direct parameter is the control gain K_v in object 0x2348.

Position control parameter set (0x2348)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2348	0x00	Number of entries	U8	ro	1	Number of object entries
	0x01	Gain K_v	U8	rw	a)	Controller gain [1/s], range: 1–255

a) *Motor-specific, is set by the motor selection wizard.*

If a FAULHABER motor is selected in the Motion Manager motor selection wizard, the pre-set controller settings for no-load operation are loaded.

The controller configuration wizard also permits the controller to be adapted to a moving load.

4.3.5.2 Setpoint values

In the Profile Position (PP) operating mode and Cyclic Synchronous Position (CSP) operating mode the setpoint values can be specified using the object 0x607A.00 of the communications system.

In the Analogue Position Control (APC) operating mode the target position is determined directly via a discrete source (see chap. 4.6, p. 53 and chap. 4.8, p. 76).

4.3.5.3 Actual value

The position actual value can be determined using various different sensors (see chap. 4.8, p. 76). The following sensor systems are often employed:

- Analogue Hall signals
- Incremental sensors or protocol-based absolute encoders (AES or SSI) for BL motors
- Incremental sensors for DC motors
- Analogue Hall signals or incremental sensors for linear BL servomotors

In addition the actual value can also be determined via an analogue voltage such as via a potentiometer coupled to the motor shaft or via a PWM signal.

Internally the position is calculated in increments directly in the resolution of the position sensor that is used. For analogue Hall signals the position resolution is 4096 increments per revolution of the shaft.

The factor group (see chap. 4.7, p. 67) allows the internal representation to be converted into application-specific physical scaling, such as into ° or mm.

Limits

The target position in the object 0x607A is limited in advance by the position range limits and the software position limits.

Position range limits limit the value range for the actual and target position values. Values outside the defined range are wrapped to the opposite end of the value range.

Example

The position range limits are set as follows:

- Negative limit (0x607B.01) = –2048
- Positive limit (0x607B.02) = 2047

During positive movement in velocity mode the actual position first reaches the value 2047 and in the next step wraps to the position –2048.

Thus no absolute setpoint values that are outside the defined range can be specified. Relative setpoint values can also be specified in the PP operating mode. Thus any desired positioning in a direction can be achieved.

For instance a belt may be driven by a shaft to achieve a direction of conveying. For this the shaft is rotated by one revolution each. In the Profile Position (PP) operating mode the target value for this is specified relative to one revolution each.

Software position limits set limits to the range of positions. No setpoint values outside this range will be accepted, provided the setpoint values are specified using the object 0x607A.00. Software Position Limits cannot be violated, even with relative positioning.

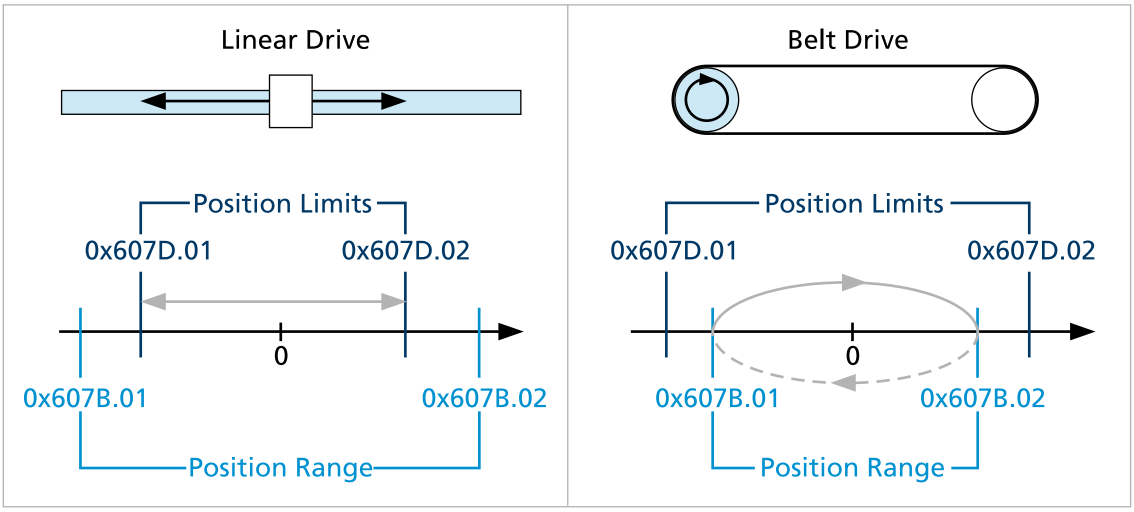


Fig. 11: Software Position Limits and Position Range Limits for linear drive systems and belt drive systems

Optimising the control system

The dynamics of position control depend on the dynamics of the subordinate velocity control loop. High gains in position control are generally available only if the subordinate motor control changes rapidly.

In addition the behaviour of the position controller can be influenced as follows:

- Via the setpoint velocity filter (0x2346). The filter time of the setpoint velocity filter primarily determines the amount of position overshoot beyond the specified target position. If the filter is deactivated, the run of the actual position into the target corridor is heavily damped if necessary.
- The behaviour of the velocity controller within the target corridor for the position can be modified appropriately by use of the gain scheduling (0x2347) parameter of the velocity controller.

Gain scheduling

The factor K_{rel} allows a relative change to the set gain of the velocity controller to be configured (see Fig. 12).

The control gain of the velocity controller can be adjusted by this means by a maximum of +-100%.

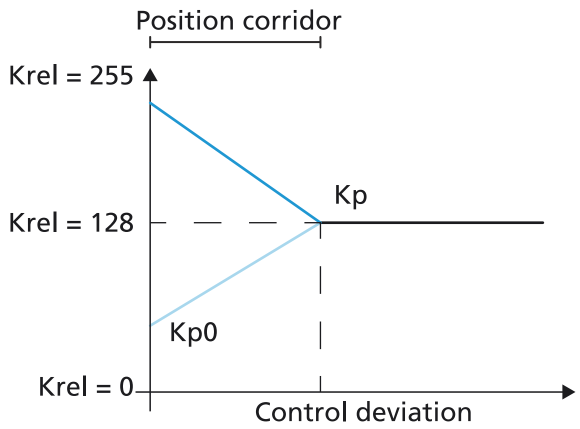


Fig. 12: Adaptive gain of the velocity controller

- Values of K_{rel} < 128 lead to a reduced gain for the velocity controller within the position corridor.
- For K_{rel} = 0 the gain in the corridor is reduced to 0.
- For K_{rel} = 128 the gain in the corridor remains unchanged.
- For K_{rel} > 128 the gain of the velocity control in the corridor is increased, up to factor of 2 at K_{rel} = 255.

Gain Scheduling (0x2347)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2347	0x00	Number of entries	U8	ro	1	Number of object entries
	0x01	Gain factor	U8	rw	128	Gain factor (used by the velocity controller in PP mode on the K_P) 0: Reduction of the gain to 0 in the target 128: no variable gain 255: Doubling the gain in the target

K_{rel} defines a factor by which the gain of the velocity controller within the position corridor is increased or decreased. The factor is effective in proportion to the position deviation.

Example

The velocity controller is to be configured to a smoother setting within the position corridor, so as to maintain the target position as smoothly as possible. For this purpose the target corridor is specified via object position window (0x6067) in increments or user-specific scalings. The factor for the maximum degree to which the gain of the velocity controller can be reduced is specified by the object 0x2347.01.

The Motion Manager commissioning wizard has already pre-set the position controller for commonly-set tasks.

The controller tuning tool is available in the Motion Manager for further optimisation of the control parameters.

For manual optimisation of the position controller, apply setpoint value jumps to the position controller and adjust the motor control gain to suit (see Fig. 13 or Fig. 14).

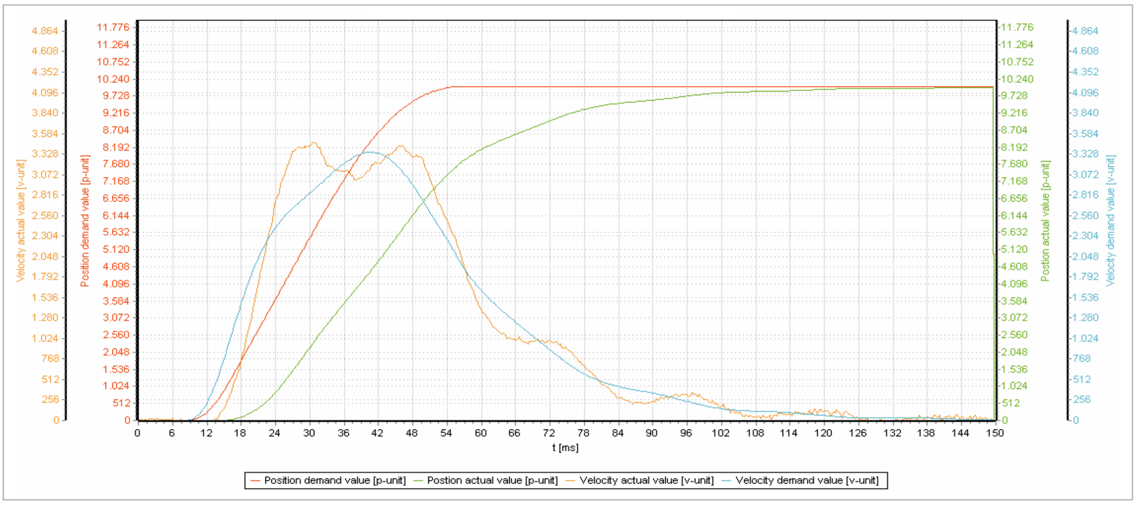


Fig. 13: Setpoint jump for the position controller

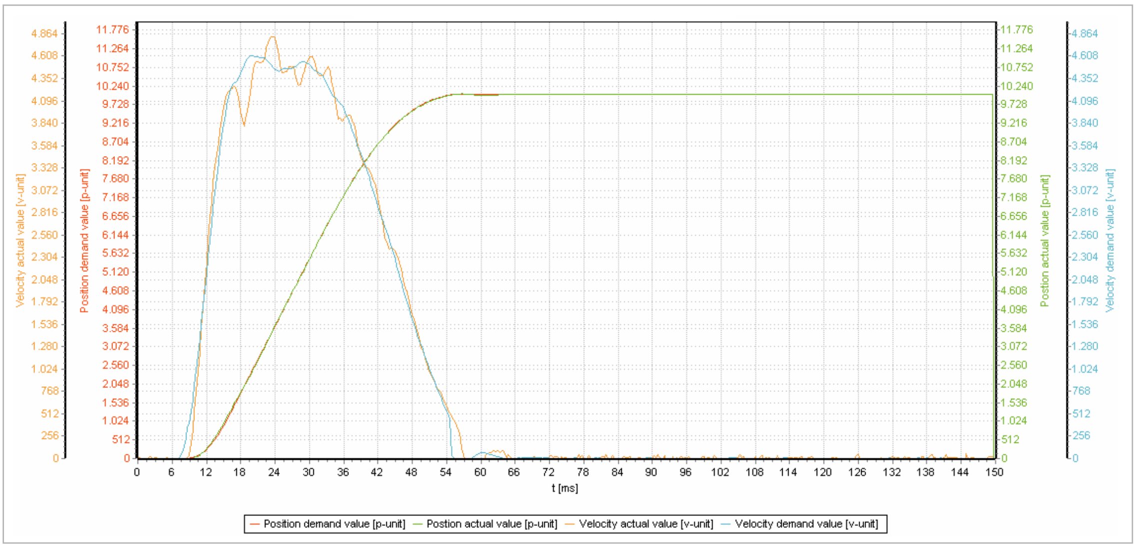


Fig. 14: Setpoint jump for the optimised position controller

Further settings

Two control monitoring functions access the position controller. In Profile Position mode the question of whether the drive has reached the target position is monitored. In addition the motor control deviation of the position controller is monitored as a following error.

The position corridor and the minimum residence time in the corridor are set as parameters for monitoring the actual position. In Profile Position mode the position is signalled as reached when the actual position has remained within the target corridor for at least the position window time.

Position Window (0x6067)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6067	0x00	Position Window	U32	rw	32	Corridor around the setpoint value position in user-defined scaling

Position Window Time (0x6068)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6068	0x00	Position Window Time	U16	rw	48	Minimum residence time within the corridor in PP operating mode, until the setpoint value position is reported as achieved.

Following Error Window (0x6065)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6065	0x00	Following Error Window	U32	rw	32	Corridor for the control deviation of the position controller

Following Error Time Out (0x6066)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6066	0x00	Following Error Time Out	U16	rw	48	Minimum time for which a following error must lie outside the defined corridor before the error is reported

4.4 Configuration of the profile generator

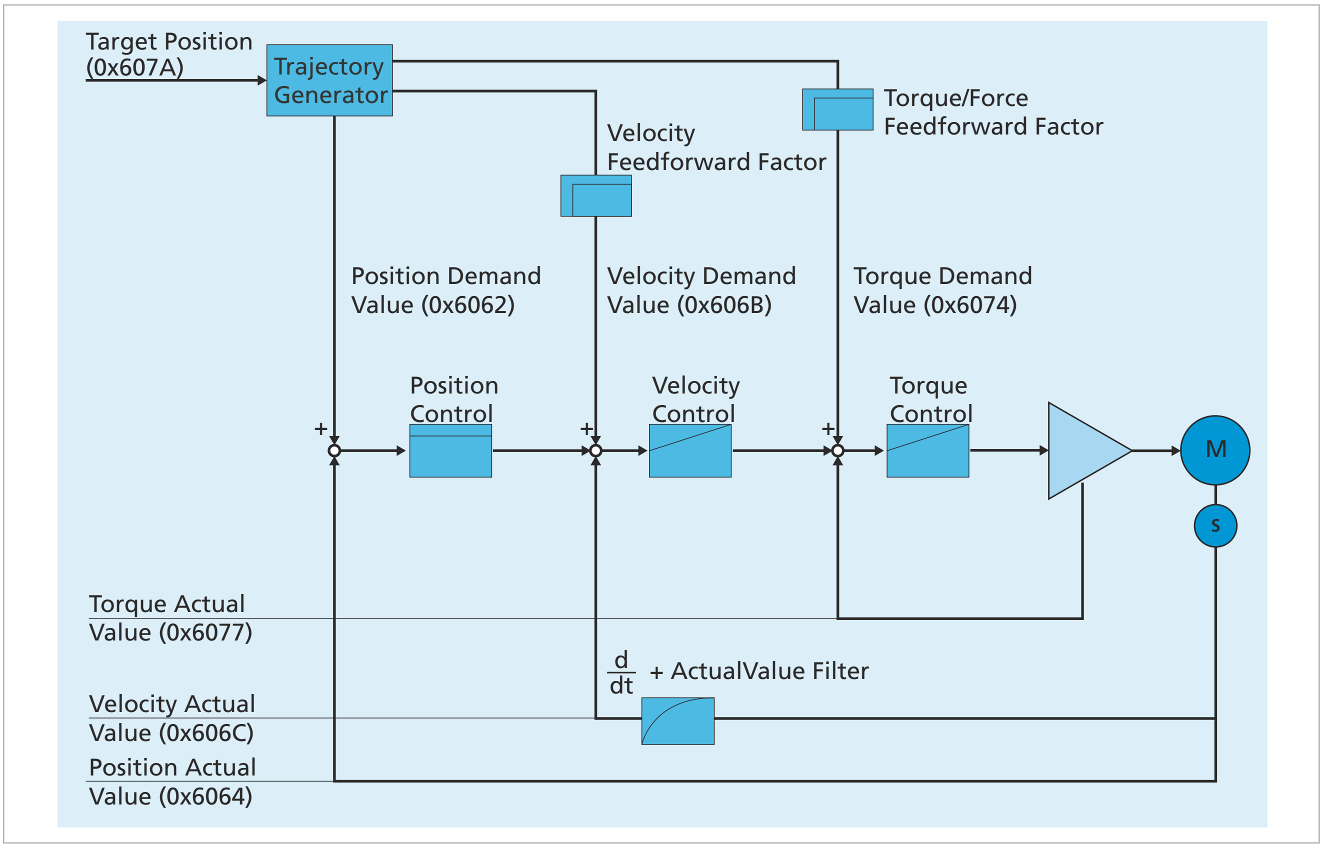


Fig. 15: Control loop with upstream profile generator

In Profile Position (PP) operating mode and Profile Velocity (PV) operating mode the profile generator calculates from the following values a movement profile for the position **Pos(t),** velocity **v(t)** and acceleration **a(t)**:

- Target position (0x607A)
- Target velocity (0x60FF)
- Acceleration (0x6083)
- Braking rate (0x6084)
- Maximum profile velocity (0x6081)

The profile type can be selected using the object 0x6086.

The following are supported:

- Linear profiles: Accelerations are activated directly. The movements correspond to a trapezoidal profile for the velocity. This type of profile is limited in relation to the activated acceleration and unlimited in relation to the resulting jerk.
- Sin^2-velocity: The accelerations are activated so that sin² velocity profiles result. This type of profile is limited both in relation to the activated acceleration and also in relation to the resulting jerk.

Linear profiles are suitable for stiff mechanisms. Linear profiles are the quickest way to achieve the target position or the target velocity. Sin² profiles are suitable for resiliently coupled mechanisms. In theory, the target position is achieved later. Since Sin² profiles generate less oscillation, the setting time can nevertheless be lower than when using linear profiles.

The setpoint values for the control system are always specified by the profile generator, the additional pre-control values can be activated by the control system. The pre-control values for velocity and torque or force can be fully or only partly activated by a parameter. In addition the pre-control values can be delayed by means of a filter.

Tab. 17: Target value for PP or PV

Operating mode	Target position	Setpoint velocity	Setpoint torque
PP	From the profile generator	Can be activated as a pre-control value	Can be activated as a pre-control value
PV	-	From the profile generator	Can be activated as a pre-control value

Velocity Feedforward Parameters (0x234A)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x234A	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Velocity feed forward factor	U8	rw	0	Factor for the torque or force control 0: 0% pre-control 128: 100% pre-control
	0x02	Velocity feed forward delay	U16	rw	0	Setpoint delay: 0: undelayed activation 1: Activation delayed by one sampling

Torque/Force Feedforward Parameters (0x2349)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2349	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Current FeedForward Factor	U8	rw	0	Factor for the torque or force control 0: 0% activation of the pre-control value 128: 100% pre-control
	0x02	Current FeedForward Delay	U16	rw	0	Setpoint delay: 0: undelayed activation 1: Activation delayed by one sampling

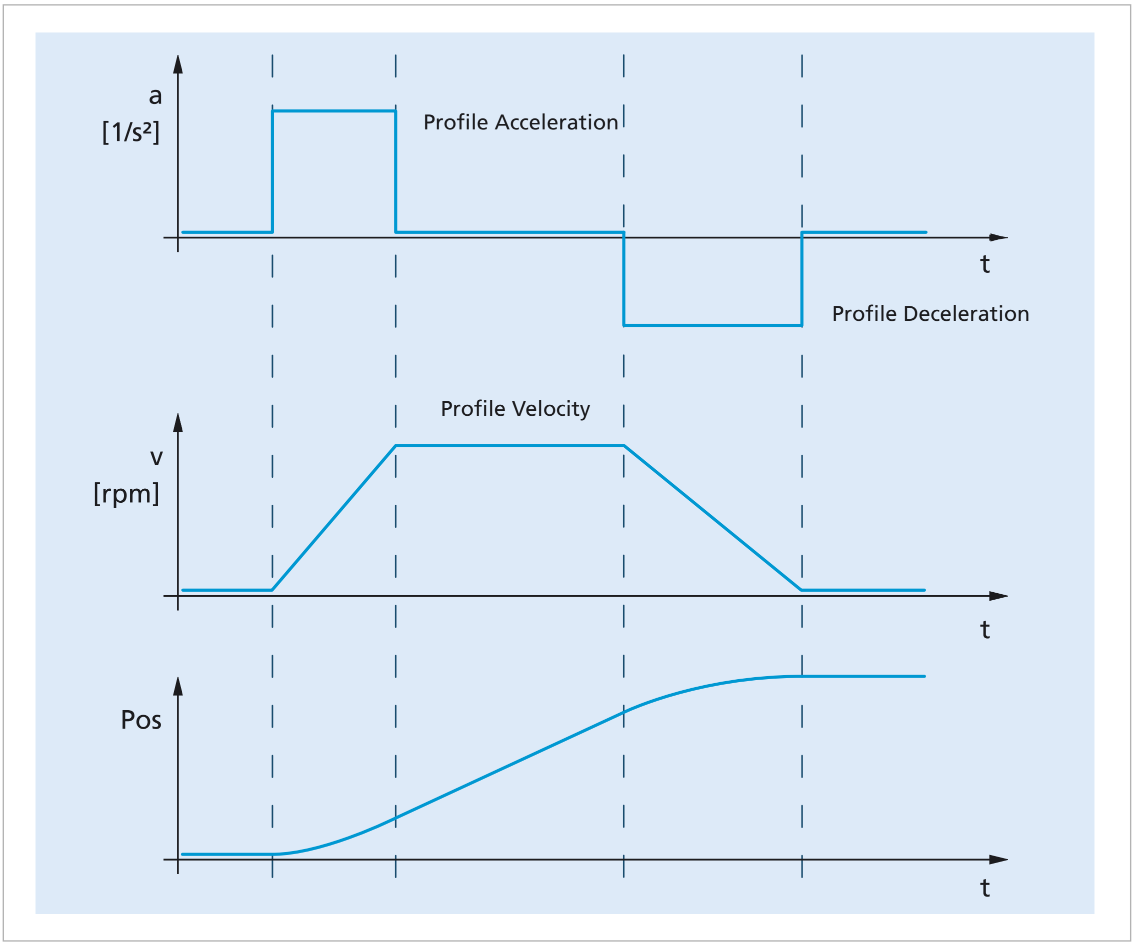


Fig. 16: Target position and velocity values when using linear profiles

In order to use the profile planning, the required profile parameters for the drive must be physically capable of being implemented.

Typical acceleration values for DC- and BLDC servomotors are in the range up to 7500 1/s.^{2}. Linear motors even reach accelerations up to 30000 1/s^2 and more.

4.4.1 Combined movement profiles

In Profile Position operating mode, profile segments can be combined with each other. This allows multiple target values with different profile parameters to be loaded successively to the Motion Controller. The following options can be selected, using the operating mode-specific bits in the controlword 0x6040:

- Send target values one after another as individual movement jobs.
- Immediately activate a new setpoint value with new profile parameters.
- Activate a new target value with new profile parameters when the preceding target value has been reached, without stopping the movement.

Tab. 18: Coding of the controlword (0x6040) and behaviour of the drive when a new target position is loaded

Bit	Function	Meaning
Bit 4	New setpoint value	The setpoint value including the profile parameters is loaded at the rising flank of bit 4.
Bit 5	Change set immediately	Load the setpoint value immediately or after the end of a current movement task: Bit 5 = 1: Movement towards the position starts immediately. Bit 5 = 0: Movement towards the new position does not start until the preceding positioning task has been completed.
Bit 6	Abs/Rel	Positions can be specified as absolute or relative setpoint values. If bit 6 is set at the rising flank of bit 4, the setpoint value is interpreted as relative.
Bit 9	Change on setpoint value	Change on the following movement task after a previous standstill or during operation. Bit 9 = 0: The new movement task is not loaded until the drive has reached its previous target position. Bit 9 = 1: On reaching the preceding target position the drive does not brake to a stop. The new movement task is activated as the drive continues through the preceding target.

The controlword (0x6040) is set by the supervisory controller. The statusword (0x6041) is the response of the drive and is set by the drive. The relevant bits in the statusword are:

- Bit 10: Target Reached
- Bit 12: Setpoint acknowledged

4.4.2 Specifying a single setpoint value

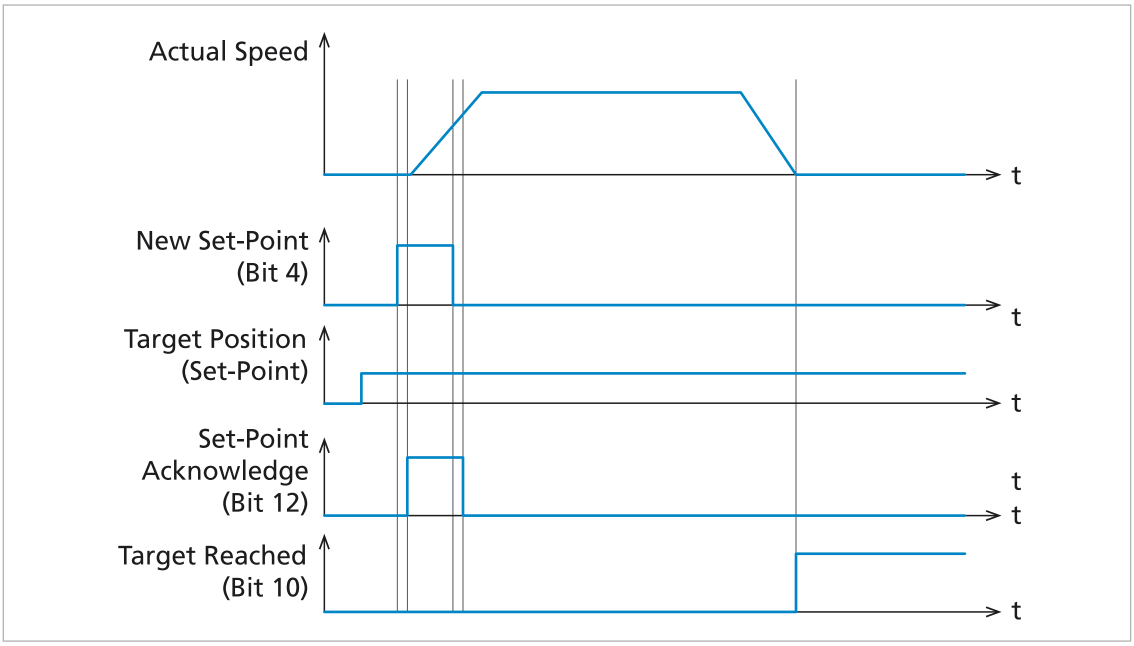


Fig. 17: Behaviour of the drive when setting a setpoint value

- In PP operating mode a new target position is not loaded until the rising flank of bit 4 in the controlword (new setpoint). To achieve this, the supervisory controller sets bit 4, after which the new setpoint is written to the object 0x607A.

If during operation via CANopen the controlword and the target position are loaded together in PDO, firstly the new setpoint value is written to the object 0x607A and then the controlword is evaluated.

- The drive checks whether the new setpoint value can be loaded. The drive signals the loading of a setpoint value via bit 12 (Setpoint Acknowledge) = 1 in the statusword. Only then the supervisory controller may reset the New Set Point bit in the controlword. If the drive can make an advance note of further setpoint values, bit 12 of the statusword (Setpoint Acknowledge) is reset to 0.
- If after the setpoint value has been reached no further setpoint values have been communicated to the drive, bit 10 of the statusword (Target Reached) is set to 1.

4.4.3 Specifying multiple setpoint values in succession (Set of Setpoints)

Whilst the first setpoint value is being processed (with the drive still running, i.e. Target Reached has not yet been signalled), further setpoint values can still be loaded. Bit 4 (New Setpoint) in the controlword and bit 12 (Setpoint Acknowledge) in the statusword are set to allow further setpoint values to be loaded to the drive.

If the value 0 is set in bit 5 of the controlword (Change Set Immediately), movement towards the new setpoint value does not start until the preceding setpoint value has been reached.

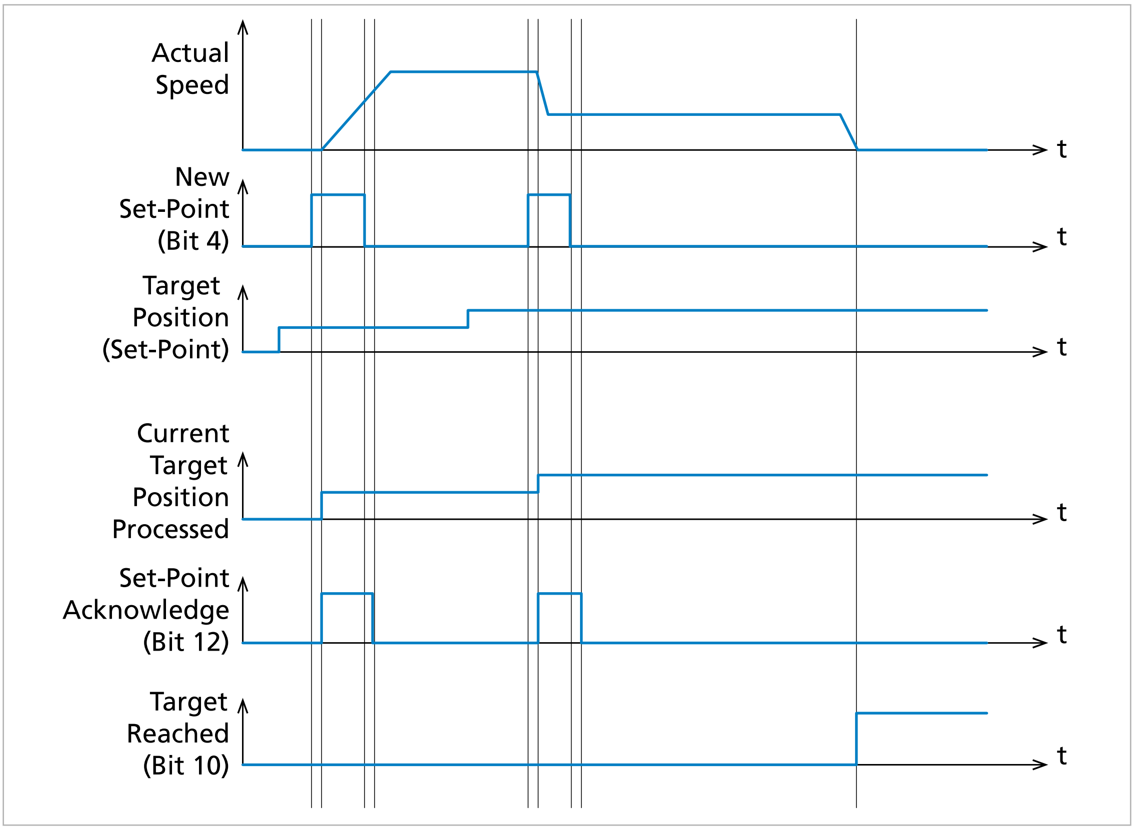


Fig. 18: Handshaking procedure for a succession of setpoint values, each setpoint value is loaded immediately

- In PP operating mode a new target position is not loaded until the rising flank of bit 4 in the controlword (new setpoint). To achieve this, the supervisory controller sets bit 4, after which the new setpoint is written to the object 0x607A.

If during operation via CANopen the controlword and the target position are loaded together in PDO, firstly the new setpoint value is written to the object 0x607A and then the controlword is evaluated.

- The drive checks whether the new setpoint value can be loaded. The drive signals the loading of a setpoint value via bit 12 (Setpoint Acknowledge) = 1 in the statusword. Only then the supervisory controller may reset the New Set Point bit in the controlword. If the drive can make an advance note of further setpoint values, bit 12 of the status-word (Setpoint Acknowledge) is reset to 0.

- If a setpoint value is set and a further setpoint value is to be transferred to the drive, two types of response behaviour are available depending on the setting of bit 5 of the controlword (Change Set Immediately):
  - Change set immediately = 1: Movement towards the new setpoint value starts immediately.
  - Change set immediately = 0: Movement towards the new setpoint value does not start until the preceding setpoint value has been reached. The drive stops briefly in between the two movements until it recognises that the first target has been reached (see chap. 4.4.4, p. 49).
- If after the setpoint value has been reached no further setpoint values have been communicated to the drive, bit 10 of the statusword (Target Reached) is set to the value 1.

In the example shown (see Fig. 18), new instructions for the acceleration or the velocity were communicated with the new setpoint value; this is evidenced by the change of the velocity profile (Actual Speed).

4.4.4 Specifying multiple position setpoint values with direct transition (Change on Setpoint)

If the value 0 is set in bit 5 of the controlword (Change Set Immediately), movement towards the new setpoint value does not start until the preceding setpoint value has been reached.

If bit 9 (Change on Setpoint) = 0, the drive stops between the two setpoints before the next movement instruction is being processed. If on the other hand the Change on Setpoint bit is set, the change to the new profile parameters proceeds without stopping as soon as the preceding setpoint value has been achieved. The specification of setpoints with a change on setpoint bit setting is thus particularly suitable for continuous positioning movements with continuously changing profile parameters such as for 2-axis operation when milling or for a 3D printer.

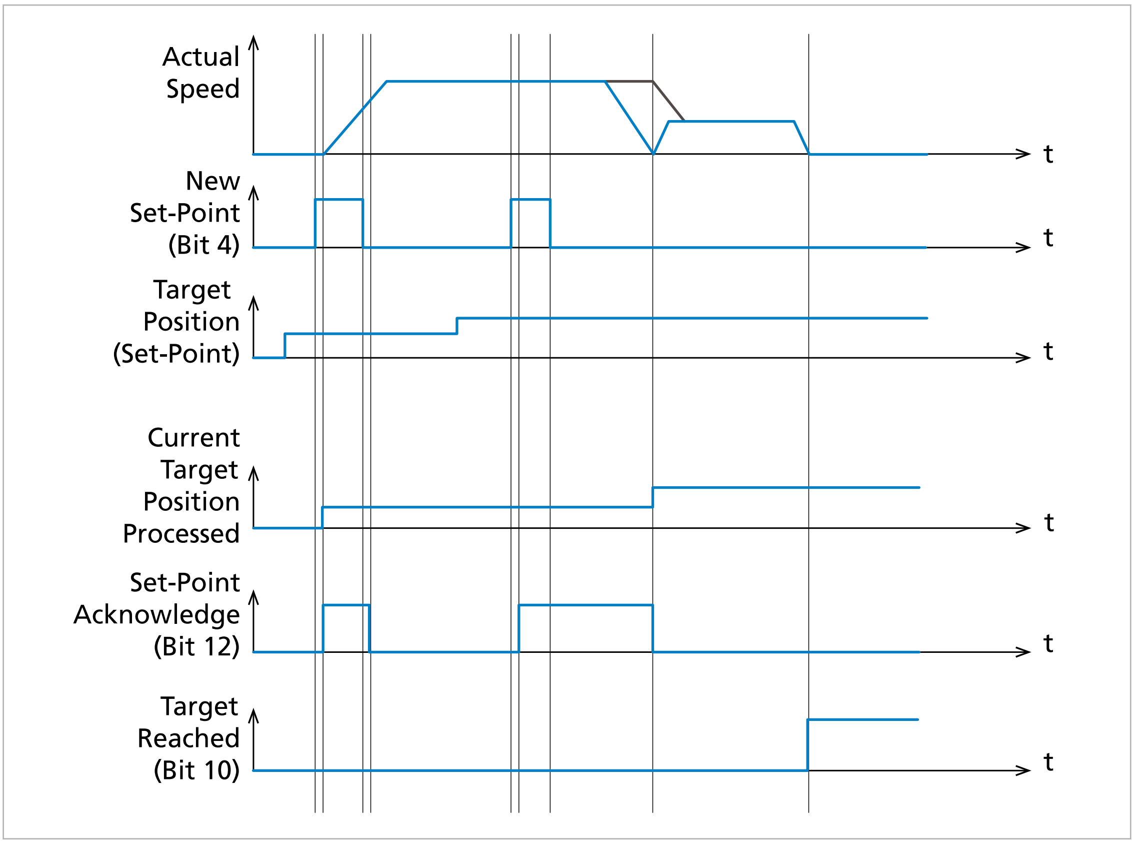


Fig. 19: Handshaking procedure for the set of setpoint value methods

- In PP operating mode a new target position is not loaded until the rising flank of bit 4 in the controlword (new setpoint). To achieve this, the supervisory controller sets bit 4, after which the new setpoint is written to the object 0x607A.

> **i** If during operation via CANopen the controlword and the target position are loaded together in PDO, firstly the new setpoint value is written to the object 0x607A and then the controlword is evaluated.

- The drive checks whether the new setpoint value can be loaded. The drive signals the loading of a setpoint value via bit 12 (Setpoint Acknowledge) = 1 in the statusword. Only then the supervisory controller may reset the New Set Point bit in the controlword. If the drive can make an advance note of further setpoint values, bit 12 of the status-word (Setpoint Acknowledge) is reset to 0.

- If during movement towards the first setpoint value a further setpoint value is sent to the drive (Immediate Bit = 0), this will be noted in advance. The drive will move towards this setpoint value only once the first setpoint value has been reached.
- Bit 9 of the controlword (Change on Setpoint) controls the dynamic behaviour during the transition from one setpoint value to another setpoint value (see Fig. 19):
  - Change on Setpoint = 0: The drive stops at the setpoint value. Then it proceeds towards the new setpoint value.
  - Change on setpoint value = 1: The drive brakes or accelerates to the velocity profile of the next setpoint value. Movement towards the new setpoint value starts without any preceding stop.
- If after the setpoint value has been reached no further setpoint values have been communicated to the drive, bit 10 of the statusword (Target Reached) is set to the value 1.

4.4.5 Example of a specification of multiple position setpoint values

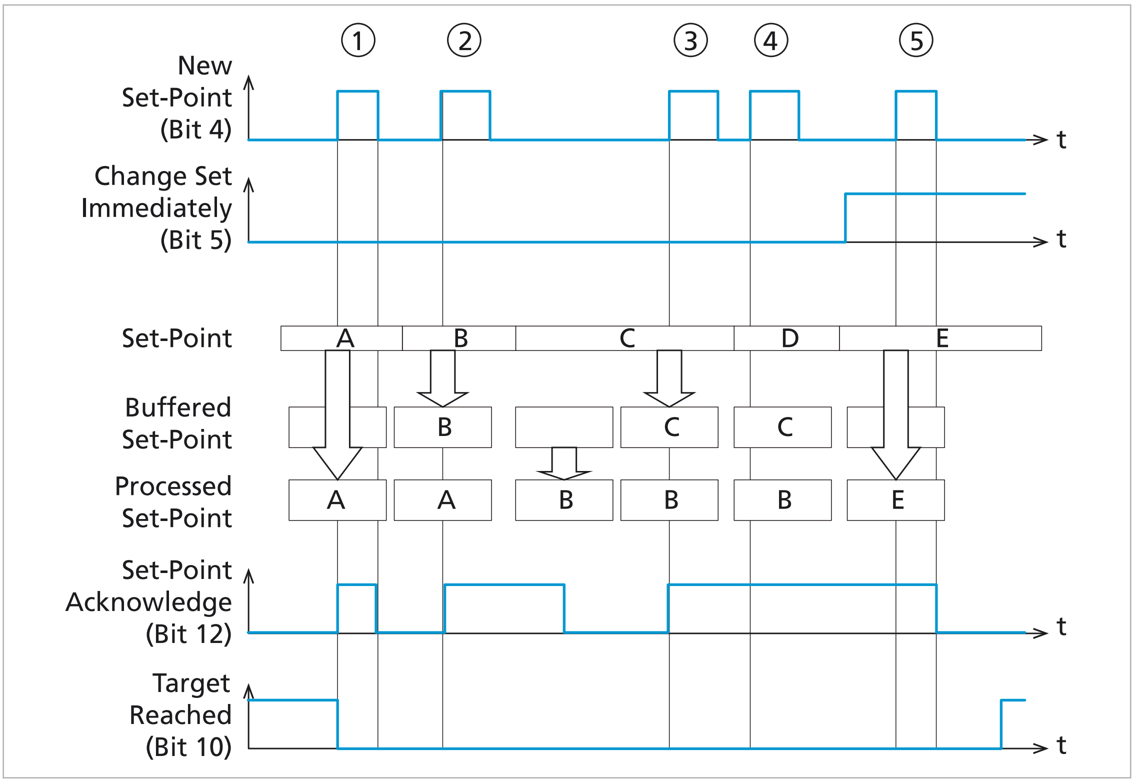


Fig. 20: Transfer of multiple setpoint values for a drive with a maximum of 2 setpoint value memory slots

- Setpoint A is communicated to the drive. Bit 12 in the statusword (Setpoint Acknowledge) is initially set from 0 to 1. Since the drive can still accept further setpoint values, bit 12 in the statusword (Setpoint Acknowledge) is then reset again from 1 to 0 as soon as the “New Setpoint” bit in the controlword is reset.
- Setpoint value B is communicated to the drive. Bit 12 in the statusword (Setpoint Acknowledge) is initially set from 0 to 1. Since the drive cannot accept any further setpoint values, bit 12 in the statusword (Setpoint Acknowledge) is not reset from 1 to 0 until the setpoint value A has been reached.

- Setpoint value C is communicated to the drive. Bit 12 in the statusword (Setpoint Acknowledge) is set from 0 to 1. Since the drive cannot accept any further setpoint values, bit 12 in the statusword (Setpoint Acknowledge) is not reset from 1 to 0.
- Setpoint value D is communicated to the drive. Since setpoint value C is still flagged in the buffer, the drive cannot accept any further setpoint values. Bit 12 in the statusword (Setpoint Acknowledge) remains unchanged at 1.
- Setpoint value E is communicated to the drive. Bit 5 of the controlword (Change Set Immediately) was previously set from 0 to 1. The drive immediately moves towards the setpoint value E. All previous setpoint values are discarded.

In addition to the target position being executed, FAULHABER Motion Controllers have the capacity to store a maximum of two further setpoint values.

4.5 Voltage output

Object 0x2340.01 allows the type of voltage output to be specified. In addition, the voltages that are output can be read here. The setting variants for the voltage output are as follows:

- 0: not active
- 1: DC motor
- 2: BL motor with block commutation
- 3: BL or linear motor with sinusoidal commutation

The further sub-indexes of the object 0x2340 allow the voltages at the motor to be read back. The scaling is 10 mV per digit.

The selection of the motor variant is available only in the *Switch On Disabled* state.

BL motors with block commutation are supported only for BL motors with digital Hall signals.

BL motors or linear motors with sinusoidal commutation require analogue Hall signals, an AES or SSI absolute encoder or digital Hall signals in combination with an incremental encoder connected to the encoder input.

General parameters (0x2340)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2340	0x00	Number of entries	U8	ro	8	Number of object entries
	0x01	Commutation Type	U8	rw	3	Commutation type 0: switched off 1: DC motor 2: BL motor with block commutation 3: BL motor with sinusoidal commutation
	0x02	Motor Output Voltage DC	S16	rw	-	Motor output voltage DC [a)]
	0x03	Motor Output Voltage BL Block	S16	rw	-	Motor output voltage, BL block [a)]
	0x04	Motor Output Voltage X_d	S16	rw	-	Motor output voltage X_d [a)]
	0x05	Motor Output Voltage X_q	S16	rw	-	Motor output voltage X_q [a)]
	0x06	Sinus Output Voltage U_a	U16	ro	-	Phase voltage U_a [a)]
	0x07	Sinus Output Voltage U_b	U16	ro	-	Phase voltage U_b [a)]
	0x08	Sinus Output Voltage U_c	U16	ro	-	Phase voltage U_c [a)]

a) *All voltages are in multiples of 10 mV*

If a FAULHABER motor is selected in the Motion Manager motor selection wizard, the commutation type is already set to match the motor type

4.6 Configuration of the digital inputs and outputs

The digital inputs and outputs of the FAULHABER Motion Controller can be used flexibly.

The following functions of the digital inputs are supported:

- Connection of limit switches
- Direct referencing of the drives with a reference switch
- Connection of a setpoint value or actual value to a PWM signal at DigIn1 or DigIn2
- Connection of an additional 2-channel or 3-channel quadrature encoder to DigIn1 - DigIn3
- Setpoint specification for the position controller using a pulsed directional signal at DigIn1 and DigIn2
- Recording the actual position in response to a flank at the input (touch probe)
- Free inputs for procedures that are programmable at the controller
- Default value of the polarity for an analogue input such as the direction of rotation)

The following functions of the digital outputs are supported:

- Output of an error signal
- Direct activation of a holding brake
- Output of a freely configurable diagnostic signal such as for the following applications:
  - Controller limitations
  - Temperature warnings
  - Display at standstill (n = 0)
  - Achievement of the target position
- Free outputs for procedures that are programmable at the controller

The following functions of the sensor connections are supported:

- Sensor connection:
  - Connection of three analogue Hall sensors as motor position sensors for position and velocity control and also for commutation
  - Connection of three digital Hall sensors as motor position sensors for velocity control and for commutation
  - Connection of two analogue Hall sensors (sin/cos) as motor position sensors for position and velocity control and also for commutation
- Encoder connection:
  - Connection of an incremental encoder with two or three channels
  - Connection of a 12-bit AES encoder
  - Connection of a 12-bit SSI encoder with binary coding as actual value.

    As actual value of the position and velocity of the controlled motor or as target position and target velocity.
- Analogue inputs (AnIn1/AnIn2):
  - Connection of analogue sources with a voltage range of ±10 V as target position,

target velocity or target torque

- Connection of analogue sources with a voltage range of ±10 V as actual values for position or velocity

- PWM input (DigIn1 or DigIn2):
  - Connection of a PWM signal as a target position, target velocity or target torque
  - Connection of a PWM signal as an actual value for position or velocity
- Additional encoders (DigIn1 - DigIn3)
  - Connection of an incremental encoder two or three channels.
  - Connection of a pulsed/directional signal at DigIn1 and DigIn2 as a setpoint or actual value for the position of the drive.

The sensors are already incorporated in Motion Controllers of the MCS product range. Motion Controllers of the MCS product range therefore haven no sensor connections.

4.6.1 Configuring the digital inputs

4.6.1.1 Configuring limit switches and reference switches

- Configure the digital input for the lower limit switch via a bit mask in the object 0x2310.01.

Tab. 19: Bit mask of the object 0x2310.01 (lower limit switch)

0x2310.01	In8	In7	In6	In5	In4	In3	In2	In1

- Configure the digital input for the upper limit switch via a bit mask in the object 0x2310.02.

Tab. 20: Bit mask of the object 0x2310.02 (upper limit switch)

0x2310.02	In8	In7	In6	In5	In4	In3	In2	In1

- Configure the behaviour on reaching the limit switch in the object 0x2310.03.

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2310	0x03	Limit Switch Option Code	I8	rw	1	0: – 1: Brake ramp 2: Quick stop 3: Stop at max. current 4: Stop at max. current

- Configure the digital input for the reference switch by entering the input number into the object 0x2310.04.

↳ Limit switches and reference switches are now configured.

If multiple inputs are set at the same time for the lower or upper limit switch, the function becomes active when one of the switches trips.

The number of available digital inputs depends on the Motion Controller used.

Only one reference switch may be selected.

4.6.1.2 General settings of the digital inputs

Setting the active level

- Use the flags in the object 0x2310.10 to set whether a high level or a low level at an input should be evaluated as an active level.
  - Input mask bit = 0: Input is not inverted (high = active)
  - Input mask bit = 1: Input is inverted (low = active)

Tab. 21: Bit mask of the object 0x2310.10

0x2310.10	In8	In7	In6	In5	In4	In3	In2	In1

The active level is now set.

Setting the trigger thresholds

- In object 0x2310.11, determine whether the trigger thresholds of the digital input should be 5 V TTL-compatible or 24 V SPS/PLC-compatible.
  - 0x2310.11 = 0: TTL level for all digital inputs
  - 0x2310.11 = 1: PLC level for all digital inputs

The setting of the trigger thresholds is also effective if the digital inputs are used as a connection for a reference encoder.

The trigger threshold is now set.

Configuring the filters at digital inputs

- Configure filters for the digital input to be filtered using object 0x2310.12.
  - Input mask bit = 0: Filter deactivated (default)
  - Input mask bit = 1: Filter activated

Tab. 22: Bit mask of the object 0x2310.12

0x2310.12	In8	In7	In6	In5	In4	In3	In2	In1

When a filter is activated, as change of level must be present for at least 4 ms before it is recognised as valid.

4.6.1.3 Configuring digital inputs DigIn1 - DigIn3 as connections for an additional encoder

If the additional encoder used is configured in the object 0x2316.01 as a pulsed/directional input or as a 2-channel or 3-channel incremental encoder, the inputs are configured automatically.

Functions set at digital inputs are also evaluated if an encoder is activated via 0x2310.

So as to avoid malfunctions, the digital inputs used for the encoder may not be used as limit switches or reference switches.

If an additional encoder is used, the setting of the voltage level for the digital inputs in the object 0x2310.08 is active and the settings in the object 2310.04 for the logical level are ineffective.

- Use the object 0x2316.01 to configure the additional encoder that is used (see chap. 4.6.4.2, p. 59).

4.6.2 Directly read the level of the digital inputs and outputs, or directly write the digital outputs

- In object 0x2311.01, read the logical status of the digital inputs having regard to the polarity from object 0x2310.10.

Tab. 23: Bit mask of the object 0x2311.01

0x2311.01	In8	In7	In6	In5	In4	In3	In2	In1

- In object 0x2311.02 directly read the physical status of the digital inputs. A high level at the input leads to a set bit in the mask.

Tab. 24: Bit mask of the object 0x2311.02

0x2311.02	In8	In7	In6	In5	In4	In3	In2	In1

- Directly read the logical status of the digital outputs in object 0x2311.03. A set output leads to a set bit in the mask.
- Directly set, toggle or delete a digital output in object 0x2311.04.

Example: Toggling digital output 3

- Set the value **0x00EF (bit pattern 11 10 11 11)** in object 0x2311.04.
  - ↳ Digital output 3 is now toggled.

    The other digital outputs remain unchanged.

Tab. 25: Meaning of the bit pattern

0x2311.04	DigOut4		DigOut3		DigOut2		DigOut3	
	B1	B0	B1	B0	B1	B0	B1	B0

B1|B0 = 00: Deleting a digital output
B1|B0 = 01: Setting a digital output
B1|B0 = 10: Toggling a digital output
B1|B0 = 11: Leave a digital output unchanged

4.6.3 Configuring the digital outputs

4.6.3.1 Configuring the fault output

- Set the digital output to be used for the fault output in object 0x2312.01.
- Use the mask in object 0x2321.03 to configure which internal faults should trigger the fault signal (see chap. 7, p. 136).
- The fault output is now configured.

4.6.3.2 Configuring the digital output as a brake activation

If a holding brake is used, it may be necessary to impose a waiting time before the output stage and controller can be activated or deactivated. For this purpose it may for instance be ensured that the brake has been reliably applied before the motor controller is switched off.

- Set the digital output to be used for the brake activation in the object 0x2312.02.
- Configure the waiting time in the object 0x2312.03.
- The digital output is now configured as a brake activation.

4.6.3.3 Configure a digital output as a diagnostic output

To configure a digital output as a diagnostic output, a bit mask must be defined in relation to the device statusword 0x2324.01 for each digital output (see chap. 7, p. 136).

If the bit-wise AND link of the bit mask with the statusword delivers a result > 0, the configured digital output is activated.

Example: The standstill of the drive is to be signalled by digital output 2:

1. Set the value **0x02** in the object 0x2312.08.
   - Digital output 2 will be used.
2. Set the bit mask **0x00 00 00 01** in the object 0x2312.09.
   - Only at a standstill (n = 0) is an output performed.

- At a standstill, firstly bit 0 is set (n=0) in the object 0x2324.01. Output 2 is then set using the mask in the object 0x2312.09.

4.6.3.4 Configuring the polarity of the digital output

- Set the polarity of a digital output in the object 0x2312.10.
  - Input mask bit = 0: A set digital output switches the output to ground. A low level is measured at the DigOut pin.
  - Input mask bit = 1: When the digital output is set it switches the output into a high-resistance state. A high level is measured at the DigOut pin when the pin is switched by an external pull-up resistance relative to the supply.

Tab. 26: Bit mask of the object 0x2312.10

0x2312.10	Out8	Out7	Out6	Out5	Out4	Out3	Out2	Out1

4.6.4 Configuring the sensor inputs

4.6.4.1 Configuring the motor encoder

Either an incremental encoder or a protocol-based AES or SSI encoder can be connected at the encoder connection. The trigger threshold is always at the TTL level. The motor encoder is configured using the inputs in the object 0x2315.

Tab. 27: Motor encoder settings

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2315	0x00	Number of entries	U8	ro	3	Number of object entries
	0x01	Operation Mode, Index Polarity	U16	rw	0	Selection of the encoder type
	0x02	Resolution	U16	rw	0x0800	Encoder resolution in increments/revolutions (4-flank evaluation)
	0x03	Motor Encoder Position	S32	ro		Current position value

Tab. 28: Encoder types available for selection in the object 0x2315.01

Bit	Meaning
0	Incremental encoders
1	Index signal present
2	Index signal evaluation at the rising flank
3–7	Reserved
8	AES encoder with 12-bit angular resolution (4096 increments)
9	SSI encoder with 12-bit angular resolution (4096 increments)
10–15	Reserved

Example: Configuring a 3-channel incremental encoder with positive index pulse

- Set the value **0x00 07** in object 0x2315.01.
  - ↳ The 3–channel incremental encoder with positive index pulse has now been configured.

Example: Configuring a 12-Bit AES encoder

- Set the value **0x01 00** in object 0x2315.01.
  - ↳ The 12–bit AES encoder has now been configured.

Example: Configuring an incremental encoder with 512 impulses per revolution

ℹ For incremental encoders the encoder resolution in increments per revolution must be stated explicitly. By means of the quadrature method, the resolution corresponds to the 4-way line number of the encoder.

- Calculate the resolution of the incremental encoder as a quadrature signal:
  - Resolution of the encoder = 4 x 512 = 2048
- Set the value **0x00 07** for the encoder type in object 0x2315.01.

- Set the encoder resolution to the value **2048** in the object 0x2315.02.
  - ↳ The incremental encoder has been configured with 512 lines per revolution.

4.6.4.2 Configuring an additional encoder

Either an incremental encoder with or without an index signal or a pulsed/directional signal can be connected at the digital outputs. The trigger thresholds set for the digital input in the object 0x2310.08 are effective. The additional encoder is now configured by the entries in the object 0x2316.

Tab. 29: Reference Encoder Settings

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2316	0x00	Number of entries	U8	ro	5	Number of object entries
	0x01	Operation Mode, Index Polarity	U16	rw	0	Selection of the additional encoder
	0x02	Resolution	U16	rw	2048	Encoder resolution
	0x03	Reference Encoder Position	S32	ro	0	Encoder position
	0x04	Gain (Numerator / Divisor)	S32	rw	0x40004000	Conversion of the raw value of the external encoder into the position value used internally
	0x05	Offset	S32	wo	0	Position offset for the reference encoder.

The object 0x2316.04 allows the number of steps on the reference sensor to be converted into scaling suitable for the internal position resolution.

Tab. 30: Available selection of the encoder types in the object 0x2316.01

Bit	Meaning
0	Incremental encoders
1	Index signal present
2	Index signal evaluation at the rising flank
3–7	Reserved
8	Position specification by the pulsed/directional signal
9–15	Reserved

Example: Configuring a 3-channel incremental encoder with positive index pulse

- Set the value **0x00 07** in object 0x2316.01.
  - ↳ The 3–channel incremental encoder with positive index pulse has now been configured.

If a reference encoder is connected to the digital inputs, the following assignment is applicable:

- DigIn 1: Encoder track A
- DigIn 2: Encoder track B
- DigIn 3: Encoder index

Example: Position specification by the pulsed/directional signal

- Set the value **0x0100** in object 0x2316.01.
  - ↳ Position specification is set by the pulsed/directional signal.

The following assignment is applicable to the position specification by the pulsed/directional signal:

- DigIn 1: Impulse
- DigIn 2: Direction
  - 0: Negative direction of movement
  - 1: Positive direction of movement

Example: Specifying a target position via an external reference encoder

A BL motor with analogue Hall signals should perform one revolution for each revolution of the reference signal. The reference signal has a resolution of 16384 increments per revolution.

There is no need to set a resolution for the additional encoder. It is not used as an velocity actual value encoder. Instead of this a conversion factor from the external resolution to an internal resolution must be stated in the object 0x2316.04.

- Calculate the target position:
  - internal setpoint value = reference value x (4096 / 16384)
- Set the scaling factor for the internal setpoint value:
  - Set the value **0x10 00 40 00** in the object 0x2316.04.
  - ↳ The scaling of the target position of an external reference encoder is set to 1/4.

4.6.4.3 Adjusting the Hall sensors

When the Motion Manager is used, the Hall sensors can be adjusted using the motor selection wizard.

The selection of the sensor type and the compensation procedure are performed using the object 0x2318.

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2318	0x00	Number of entries	U8	ro	3	Number of object entries
	0x01	Hall Sensor Type	U8	rw	0	Bit-coded selection of the sensor type (see chap. 4.6.4.3, p. 60)
	0x02	Enable Adaption	U8	rw	0	0: static adjustment switched off 1: static adjustment active
	0x03	Adaption Threshold Speed	U32	rw	1000	Minimum velocity in [min^{-1}], from which the Hall signals are compensated.

The Hall sensors can be adjusted either statically by a reference run or at least by an electrical pole width or dynamically during operation.

For motors with 2 pole pairs, perform a one-off adjustment of the sensor signals in both magnetic poles.

For the shortest movement distance during the static adjustment of the Hall sensors, activate the adjustment before starting the motor.

For dynamic adjustment or the special adjustment of BX4 and BP4 motors, run the motor initially at a low constant speed, and then start the compensation.

For dynamic adjustment, object 0x2318.03 allows the setting of a minimum threshold for the speed. The Hall sensors are adjusted when the value of the speed is above the specified value.

1. Set the sensor type and adjustment procedure in the object 0x2318.01:
   - Bit 0: Adjustment procedure

     0: Dynamic adjustment: The amplitudes of the Hall signals are progressively compensated as the motor is running. This should not be used for linear motors.

     1: Static adjustment: The amplitudes of the Hall sensors are scaled only after an explicitly started compensation run at a suitable level.
   - Bit 1: Sensor type

     0: Three sensor signals 120° apart are evaluated

     1: Two sensor signals 90° apart are evaluated.
   - 2-7: Reserved
2. Perform the adjustment (see following examples).

Example: Dynamic Hall sensor adjustment
The dynamic adjustment adjusts the Hall signals during operation.

✓ The sensor type and adjustment procedure are set.

1. Run the motor at a constant low velocity.
2. At the start of the adjustment, set a value > **0** in the object 0x2318.02.
3. Allow the motor to run for a few seconds.
4. At the end of the adjustment, set the value **0** in the object 0x2318.02.
5. Run the motor at a velocity significantly higher than the limit velocity set in 0x2318.03.
6. Allow the motor to run for a few seconds.

↳ The adjustment is now complete. The motor can be stopped and the parameters saved.

Example: Static Hall sensor adjustment
Static adjustment is particularly suitable for motors that do not have to run continuously over long distances, such as linear DC servomotors.

✓ The sensor type and adjustment procedure are set.

1. At the start of the adjustment, set a value > **0** in the object 0x2318.02.
2. Run the drive for the maximum available length.
3. At the end of the adjustment, set the value **0** in the object 0x2318.02.

The adjustment will be successful if at least one complete magnetic period of the motor (e. g. the magnetic pole width of the linear DC servomotor) has been travelled.

4.6.4.4 Configuring analogue inputs

The analogue inputs of the Motion Controller can process electrical signals in the level range ±10 V. Internally the signals are shown as numeric values ±10,000 equivalent to ±100 %. The scaling factor can be set by making entries in the object 0x2313. The values are updated every 1 ms.

If the analogue inputs are used as setpoint or actual value encoders, the values from the analogue input must be converted into a suitable physical variable.

The raw value of the analogue input can be subjected to a low pass 1st order filter before further processing.

The filtered raw values can be read in the following objects:

- AnIn1: 0x2314.07
- AnIn2: 0x2314.08

The scaled end value can be read using the object 0x2313.04 or 0x2313.14.

If a polarity input is used, the following rule applies:

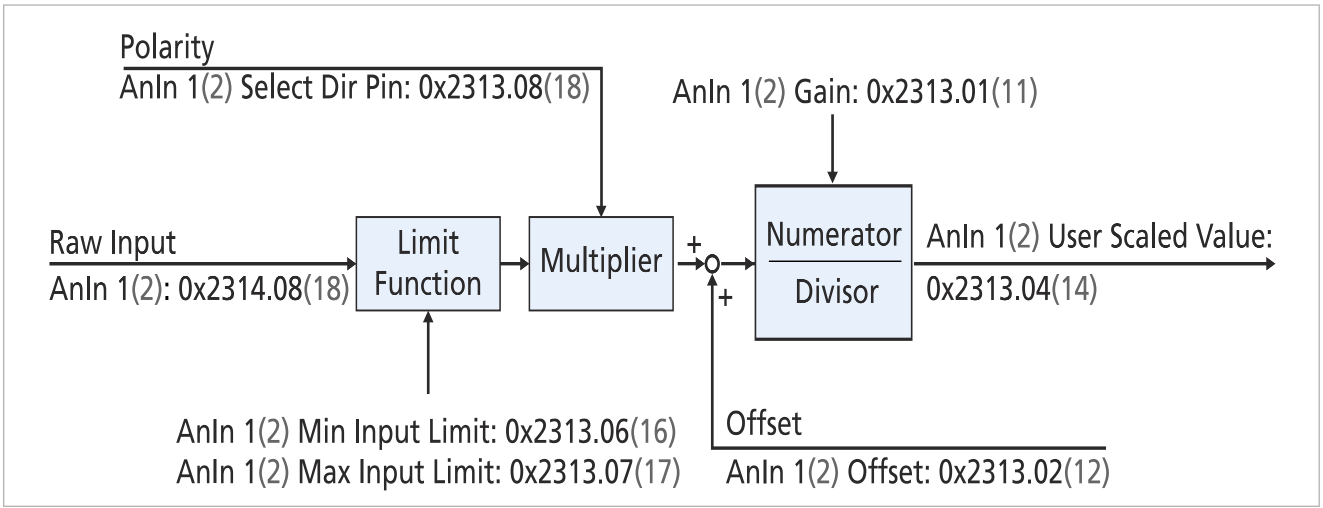
- Logic level = High, the limited raw value is multiplied by +1
- Logic level = Low, the limited raw value is multiplied by –1

Tab. 31: User-defined scalings

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2313	0x00	Number of entries	U8	ro	21	Number of object entries
	0x01	AnIn 1 Gain (Numerator/ Divisor)	S32	rw	0x7FFF8000	AnIn 1 Gain (Numerator/Denominator) ▪ Bit 1...16: Denominator ▪ Bit 17...32: Numerator
	0x02	AnIn 1 Offset	S16	rw	0	AnIn 1 Offset
	0x03	AnIn 1 Filter Time	U16	rw	0	AnIn 1 Filter time in 100 µs
	0x04	AnIn 1 User Scaled Value	S32	ro	-	Scaled AnIn 1 value
	0x05	AnIn 1 Resolu-tion as Encoder	U16	rw	1000	AnIn 1 Resolution of the encoder
	0x06	AnIn 1 Min Input Limit	S16	rw	–32768	AnIn 1 Lower limit for the input value
	0x07	AnIn 1 Max Input Limit	S16	rw	32767	AnIn 1 Upper limit for the input value
	0x08	AnIn 1 Select Dir Pin	U8	rw	0	AnIn 1 Polarity input: 0: No polarity input used 1...8: Digital input used as polarity input
	0x11	AnIn 2 Gain (Numerator/ Divisor	S32	rw	0x7FFF8000	AnIn 2 Gain (Numerator/Denominator) ▪ Bit 1...16: Denominator ▪ Bit 17...32: Numerator
	0x12	AnIn 2 Offset	S16	rw	0	AnIn 2 Offset
	0x13	AnIn 2 Filter Time	U16	rw	0	AnIn 2 Filter time in 100 µs

Index	Subindex	Name	Type	Attr.	Default value	Meaning
	0x14	AnIn 2 User Scaled Value	S32	ro	-	Scaled AnIn 2 value
	0x15	AnIn 2 Resolu-tion as Encoder	U16	rw	1000	AnIn 2 Resolution of the encoder
	0x16	AnIn 2 Min Input Limit	S16	rw	–32768	AnIn 2 Lower limit for the input value
	0x17	AnIn 2 Max Input Limit	S16	rw	32767	AnIn 2 Upper limit for the input value
	0x18	AnIn 2 Select Dir Pin	U8	rw	0	AnIn 2 Polarity input: 0: No polarity input used 1...8: Digital input used as polarity input

Example: Configuring the scaling of the ADC end value



Depending on the selected scaling, even output values >S32 (±2147483647) can be achieved. The output value is then wrapped around to the opposite end of the value range.

To prevent this, the input limits must be configured to suit.

The objects 0x2313.01 and 0x2313.11 (AnIn gain) allow the raw values of the analogue inputs to be converted into internal units.

The objects 0x2313.02 and 0x2313.12 (AnIn offset) can be used additionally to specify offset displacements.

- Intermediate values and end values are signed 32-bit variables.
- Raw values, offsets and the denominator are signed 16-bit variables.
- The numerator is an unsigned 16-bit variable.

- Calculating intermediate value 1:
  - Intermediate value 1 = raw value + offset

- Calculating intermediate value 2:
  - Intermediate value 2 = Intermediate value 1* denominator
- Calculating the end value:
  - End value = Intermediate value 2 / numerator

  ↳ Scaling of the ADC end values is configured.

Example: Filtering the ADC raw value of the AnIn1 with a filter time of 2.5 ms

- Set the value **25** in the object 0x2313.03 (unit 100 µs).

  ↳ The ADC raw value of the AnIn1 will now be filtered with a filter time of 2.5 ms.

4.6.4.5 Configuring the PWM input

DigIn1 or DigIn2 can be used to read in a PWM signal as a setpoint value or actual value for the control system. The settings of the digital inputs are performed in object 0x2317.

Tab. 32: PWM input

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2317	0x00	Number of entries	U8	ro	7	Number of object entries
	0x01	Digital Input Pin	U8	rw	0	PWM input: 1: DigIn1 = PWM input 2: DigIn2 = PWM input
	0x02	PWM Input Frequency	U32	ro		Frequency of the PWM signal
	0x03	Duty Cycle Raw Value	S16	ro		Duty cycle of the PWM signal (unscaled)
	0x04	Duty Cycle Gain (Numerator / Divisor)	U32	rw	0x7FFF8000	PWM In Gain (numerator / denominator)
	0x05	Duty Cycle Offset	S16	rw	0	PWM In Offset
	0x06	Duty Cycle Scaled Value	S32	ro		Scaled pulse width
	0x07	Resolution As Encoder	S16	rw	1000	Resolution of the encoder

The raw value of the Duty Cycle can assume values from 0 = 0% to 32767 = 100%. The measured duty cycle can be converted into an internal variable using the parameters for gain and offset.

4.6.5 Configuring the digital input as a touch probe

The current position of a drive or a reference encoder can be recorded in response to a flank at a digital output configured as a touch probe. In addition the number of flanks can be counted.

Tab. 33: Overview of the objects used

Index	Description	Attr.	Type
0x60B8	Configuration of the touch probe function	rw	U16
0x60B9	Status of the touch probe function	ro	U16
0x60BA	Position of the positive flank at input 1	ro	S32
0x60BB	Position of the negative flank at input 1	ro	S32
0x60BC	Position of the positive flank at input 2	ro	S32
0x60BD	Position of the negative flank at input 2	ro	S32
0x60D5	Counter of the positive flanks at input 1	ro	U16
0x60D6	Position of the negative flanks at input 1	ro	U16
0x60D7	Counter of the positive flanks at input 2	ro	U16
0x60D8	Position of the negative flanks at input 2	ro	U16

In total up to two inputs can be configured as a touch probe function. The configuration is performed using the object 0x60B8.

At touch probe 1, DigIn 2 is evaluated as the digital input, at touch probe 2 it is DigIn 3.

Touch probe input 2	Touch probe input 1
U8 (bit 15 - bit 8)	U8 (bit 7 - bit 0)

Tab. 34: Meaning of the bits in the object 0x60B8 (touch probe function)

Bit	Meaning
0 or 8	Enable 0: Touch probe function switched off 1: Touch probe function activated
1 or 9	Trigger Mode 0: Only the first flank is recorded 1: Flanks are continuously recorded and counted
2+3 or 10+11	Trigger source 00: The digital input is evaluated as a trigger 01: The index of the position encoder selected using 0x2330.03 is evaluated as a trigger. 10: Not used 11: Not used
4 or 12	Positive flank active 0: No evaluation of the positive flank 1: Recording of the positive flank activated

Bit	Meaning
5 or 13	Negative flank active 0: No evaluation of the negative flank 1: Recording of the negative flank activated
6+7 or 14+15	Position source 00: Not selected 01: The actual motor position is saved as the position. The position is updated every 100 µs. 10: The actual position of the reference encoder is saved as the position. The position is thus updated directly at the flank of the input. 11: Not used

As for the configuration of the touch probe, the status of both the available channels are combined within a single object.

Touch probe input 2	Touch probe input 1
U8 (bit 15 - bit 8)	U8 (bit 7 - bit 0)

Tab. 35: Meaning of the bits in the object 0x60B9

Bit	Meaning
0 or 8	Enable 0: Switched off 1: Activated
1 or 9	Positive flank recorded 0: No positive flank has yet been recorded 1: At least one positive flank has been recorded
2 or 10	Negative flank recorded 0: No negative flank has yet been recorded 1: At least one negative flank has been recorded
3–5 or 11–13	Reserved 000
6+7 or 14+15	Not used 00

Not all combinations lead to usable results.

Restrictions

- Touch input 1 evaluates DigIn2 as a trigger input. Combination with an external reference encoder (DigIn1 and DigIn2) is thus not possible.
- The same trigger cannot be used for both touch inputs. Permissible combinations are:

Input 1	Input 2
Digital input (DigIn 2)	Digital input (DigIn 3)
Index	Digital input (DigIn 3)
Digital input (DigIn 2)	Index

4.7 Factor group

The factor group is used to convert internal position values or velocities via the objects into user-defined scaling. Internal position values are stated in increments and are dependent on the resolution of the encoder that is used. Internal velocities are stated in min^{-1}. User-defined scaling such as for the position can be stated for instance in 0.1°, irrespective of the encoder resolution. User-defined scaling for the velocity on the other hand can be selected so that the movement is stated in direct relationship to a linear feed.

The relationship between the user-defined scaling and the internal position values is given by the following formula:

$$\text{Position Value} = \text{Position Internal Value} * \frac{\text{Feed Constant}}{\text{Position Encoder Resolution} * \text{Gear Ratio}}$$

The relationship between the user-defined scaling and the internal velocity values is given by the following formula:

$$\text{Velocity Value} = \text{Velocity Internal Value} * \frac{\text{Feed Constant}}{\text{Gear Ratio}} * \text{Velocity Factor}$$

The meanings of the parameters are as follows:

- Velocity internal value: The velocity of the motor in min^{-1}.
- Position encoder resolution: The resolution of the encoder used for the position control system, in increments per motor revolution.
- Gear ratio: The transmission ratio of a gearbox attached to the motor.
- Feed constant: The feed of the axis in user-defined scaling per revolution of the gearbox driving shaft.
- Velocity factor: A scaling factor for the velocity which permits scaling of the velocity independently of the position representation.

If no gearbox is fitted, a ratio of 1:1 must be set (default value).

If a position resolution set by the factor group is different from the internal resolution, the position limits must be configured as required, since these may no longer be achievable (see example below).

Example: Position resolution different from the internal resolution
The internal position can reach a maximum of S32 values (±2147483647).

With a gearbox ratio of 14:1, the motor must perform 14 revolutions for one revolution of the output shaft.

- The encoder resolution must be set (e. g. 2048 increments per revolution).
- The gearbox ratio 14:1 is set in the factor group.
- A new setpoint of 1000 increments is specified.
- Internally the motor will rotate by 14,000 increments, since the position setpoint is always interpreted as an instruction in scaling at the output shaft.

Consequently: The internal position values are greater than the external values by the gearbox ratio.

Action: The limits for the position (in increments at the output shaft) must be reduced so that no numerical overrun occurs.

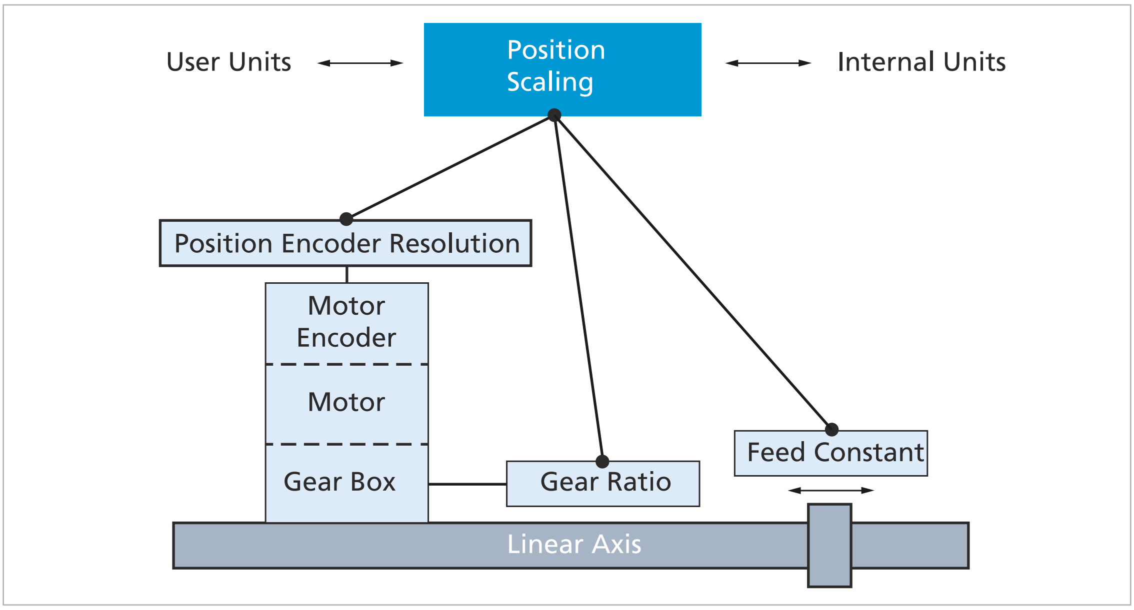


Fig. 21: Calculation of the factor group

4.7.1 Position encoder resolution

$$\textit{Position Encoder Resolution} = \frac{\textit{Encoder Increments}}{\textit{Motor Revolutions}}$$

All units are dimensionless.

Position encoder resolution

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x608F	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Encoder Incre-ments	U32	ro	4096	Encoder increments
	0x02	Motor Revolutions	U32	ro	1	Motor revolutions

The resolution of the encoder is set in the objects for configuration of the connected position encoder (see chap. 4.6.4, p. 58). The encoder used for the position control system is set via the object 0x2330.03 (see chap. 4.8, p. 76).

4.7.2 Velocity encoder resolution

The velocity encoder resolution object (0x6090) states the relationship between the encoder increments and the number of motor revolutions.

$$\text{Velocity Encoder Resolution} = \frac{\text{Encoder} * \dfrac{\text{Increments}}{\text{Sec}}}{\text{Motor} * \dfrac{\text{Revolutions}}{\text{Sec}}}$$

All units are dimensionless.

Velocity encoder resolution

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6090	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Encoder Incre-ments	U8	ro	4096	Position resolution of the sensor that is configured
	0x02	Motor Revolutions	U8	ro	1	Number of motor revolutions for the impulse number specified in subindex 1

The resolution of the encoder is set in the objects for configuration of the connected position encoder (see chap. 4.6.4, p. 58). The encoder used for the velocity control sys-tem is set via the object 0x2330.02 (see chap. 4.8, p. 76).

4.7.3 Velocity factor

The velocity factor is used to adapt the internal scaling to the user-defined scaling. The velocity factor is calculated as follows:

$$\text{Velocity Factor} = \frac{\text{Feed Velocity Units}}{\text{Feed Pos. Units}} * \frac{\text{Minutes}}{\text{User Time Units}}$$

The velocity factor consists of two parts:

$$\frac{\text{Feed Speed Units}}{\text{Feed Pos. Units}}$$

Converts the different reference ranges for the position and the velocity (see chap. 4.7.7, p. 71).

$$\frac{\text{Minutes}}{\text{User Time Units}}$$

Converts the velocity shown internally in min^{-1} into the desired time-based resolution.

If the motor is being configured using the motor wizard of the Motion Manager, the velocity factor and feed constants are pre-set:

$$\text{Feed Constant} = \frac{\text{Position Encoder Resolution}}{1}$$

$$\text{Velocity Factor} = \frac{1}{\text{Position Encoder Resolution}}$$

Thus the position is available initially in increments of the motor encoder, and the velocity in min^{-1}.

Velocity factor

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6096	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Numerator	U32	rw	1	Denominator
	0x02	Denominator	U32	rw	4096	Numerator

Tab. 36: Typical assignment of the velocity factor

Drive type	Feed	Feed speed units/ Feed position units	Minutes/User time units	Velocity factor
Rotational	Encoder Resolution (z. B. 4096)	1/4096	1	1/4096
Linear motor (e.g. LM 1247)	Magnetic pole width in µm (e. g. 18000)	1/1000	1/60	1/(1000*60)
Spindle (e.g. BS 22 1. 5)	Pitch in µm (e. g. 1500)	1/1000	1/60	1/(1000*60)

4.7.4 Gear ratio

The gear ratio object (0x6091) states the relationship of the revolutions of the motors to the number of revolutions of the driving shaft:

$$\text{gear ratio} = \frac{\text{motor shaft revolutions}}{\text{driving shaft revolutions}}$$

All units are dimensionless.

Gear ratio

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6091	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Motor Shaft Revolutions	U32	ro	1	Revolutions of the gearbox input shaft
	0x02	Driving Shaft Revolutions	U32	rw	1	Revolutions of the gearbox driving shaft

4.7.5 Feed constant

The feed constant object (0x6092) states the relationship of the feed to the number of revolutions of the driving shaft:

$$\text{feed constant} = \frac{\text{feed}}{\text{driving shaft revolutions}}$$

The feed is stated in user-defined scaling. The revolutions of the driving shaft are dimensionless.

Feed constant

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6092	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Feed	U32	rw	4096	Feed
	0x02	Shaft Revolutions	U32	rw	1	Revolutions

4.7.6 Polarity

The polarity object (0x607E) multiplies the setpoint value by 1 or –1 and is bit-coded. 0x80 inverts the position values, 0x40 inverts the velocity values.

The polarity object has no effect on the Homing mode.

Polarity (0x607E)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x607E	0x00	Polarity	U8	rw	0	Bit-coded

4.7.7 Examples of the Factor Group

4.7.7.1 General - conversion of a position

Internally the position of the drive is held in increments of the position encoder that is used. The number of revolutions of the motor shaft is counted in encoder increments per revolution of the motor.

However in systems with a gearbox and in some cases a connected linear unit, it may be desired to express in µm the orientation position of an attachment to the drive.

In rotational systems the factor group can be used to express the position uniformly across an entire group of drives, even irrespective of the resolution of the position encoder, e. g. in 0.1° rotation of the driving shaft, even with an additional attached gearbox. The following calculation instructions allow conversion of an internal position into an application-oriented position.

$$Position_{User} = Polarity * \frac{Feed}{Shaft\ Rev.} * \frac{Gear\ Shaft\ Rev.}{Motor\ Rev.} * \frac{Motor\ Rev.}{Encoder\ Increments} * Pos_{int}$$

The parameters are:

- The feed per revolution of the gearbox driving shaft:

$$Feed = \frac{Feed}{Shaft\ Rev.}$$

- The gearbox ratio:

$$Gear\ Ratio = \frac{Gear\ Shaft\ Rev.}{Motor\ Rev.}$$

- The resolution of the position encoder:

$$Position\ Encoder\ Resolution = \frac{Encoder\ Increments}{Motor\ Revolutions}$$

Thus the internal position in increments is initially converted into motor revolutions. The gearbox ratio is then used to determine the number of revolutions at the gearbox driving shaft. The feed constant is then used to determine the distance moved.

4.7.7.2 General - conversion of a velocity

FAULHABER Motion Controllers calculate the velocity of the motor internally in min^{-1}.

In addition the velocity can be converted into a representation independently of the drive. The conversion uses information that has already been used for the conversion of the position (see chap. 4.7.7.1, p. 71):

$$Velocity_{User} = Polarity * \frac{Feed}{Shaft\ Rev.} * \frac{Gear\ Shaft\ Rev.}{Motor\ Rev.} * Velocity\ Factor * Velocity_{int}$$

The velocity factor is also used here (see chap. 4.7.3, p. 69).

4.7.7.3 Configuring a DC motor with incremental encoder without a gearbox within a spindle system

The following system is available:

- In a spindle system, the position should be specified in µm. The velocity is specified in mm/s.
- A DC motor with incremental encoder is used.
- The incremental encoder has a resolution of 512 lines.
- No gearbox is connected.
- The spindle has pitch of 1.5 mm per revolution.

✓ The motor type is specified in the object 0x2329 or in the Motion Manager.

✓ The incremental encoder is configured in the objects 0x608F.01 and 0x6090.01 with a resolution of 2048 increments per revolution (see chap. 4.6.4, p. 58)

✓ The incremental encoder is configured as a position and velocity sensor in the object 0x2330.

The resolution of the sensor can be read from the objects in the factor group.

- Position encoder:
  - 0x608 F.01=2048
  - 0x608F.02 = 1
- Velocity encoder:
  - 0x6090.01 = 2048
  - 0x6090.02 = 1

By means of the quadrature signal, the resolution of the encoder corresponds to the 4-way line number of the encoder.

- Set the feed in the object 0x6092:
  - Set the value **0x05DC (1500)** for the pitch of the spindle in the object 0x6092.01.
  - Set the value **0x0001 (1)** for the spindle revolutions in the object 0x6092.02.
- Set the velocity factor in the object 0x6096:
  - Feed position units = 1000 (µm)
  - Feed velocity units = 1 (mm)
  - User time units = 60 (s/min)

The factor group is set to suit the application.

4.7.7.4 Configuring a DC motor with incremental encoder with a gearbox within a spindle system

The following system is available:

- In a spindle system, the position should be specified in µm. The velocity is specified in mm/s.
- A DC motor with incremental encoder is used.
- The incremental encoder has a resolution of 512 lines.
- A gearbox with a ratio of 14:1 is available.
- The spindle has pitch of 1.5 mm per revolution.

✓ The motor type is specified in the object 0x2329 or in the Motion Manager.

✓ The incremental encoder is configured in the object 0x608F.01 and 0x6090.0 with a resolution of 2048 increments per revolution (see chap. 4.6.4, p. 58)

✓ The incremental encoder is configured as a position and velocity sensor in the object 0x2330.

The resolution of the sensor can be read from the objects in the factor group.

- Position encoder:
  - 0x608 F.01=2048
  - 0x608F.02 = 1
- Velocity encoder:
  - 0x6090.01 = 2048
  - 0x6090.02 = 1

By means of the quadrature signal, the resolution of the encoder corresponds to the 4-way line number of the encoder.

- Set the gearbox ratio in the object 0x6091:
  - Set the value **0x000E (14)** for the number of motor revolutions per driving shaft revolution in the object 0x6091.01.
  - Set the value **0x0001 (1)** for the driving shaft revolutions in the object 0x6091.02.
- Set the feed in the object 0x6092:
  - Set the value **0x05DC (1500)** for the pitch of the spindle in the object 0x6092.01.
  - Set the value **0x0001 (1)** for the spindle revolutions in the object 0x6092.02.
- Set the velocity factor in the object 0x6096:
  - Feed position units = 1000 (µm)
  - Feed velocity units = 1 (mm)
  - User time units = 60 (s/min)

↳ The factor group is set to suit the application.

4.7.7.5 Configure the linear motor with analogue Hall sensors

The following system is available:

- In a linear drive system, the position should be specified in µm. The velocity is specified in mm/s.
- A linear motor LM1247 with a magnetic pole width of 18 mm is used.
- Hall sensors are used for the actual value of the following variables:
  - Commutation angle
  - Velocity
  - Position

✓ The motor type is specified in the object 0x2329 or in the Motion Manager.

✓ The analogue Hall sensors are configured as actual value encoders.

The resolution of the sensor can be read from the objects in the factor group.

- Position encoder:
  - 0x608 F.01= 4096
  - 0x608F.02 = 1
- Velocity encoder:
  - 0x6090.01 = 4096
  - 0x6090.02 = 1

- Set the feed in the object 0x6092:
  - Set the value **0x4650 (18000)** for the feed in the object 0x6092.01.
  - Set the value **0x0001 (1)** for the reference variable in the object 0x6092.02.
- Set the velocity factor in the object 0x6096:
  - Feed position units = 1000 (µm)
  - Feed velocity units = 1 (mm)
  - User time units = 60 (s/min)

↳ The factor group is set to suit the application.

4.8 Signal pathways

4.8.1 Selection of the actual values

The actual values for the motor position and the velocity can be taken from different sources. In addition, for BL motors the commutation angle can be taken from different sources.

If BL motors are used in combination with analogue Hall sensors or AES encoders, the motor position, velocity and commutation angle are reported by the same sensor.

If digital Hall sensors are used for the commutation, we recommend that the velocity and position information are recorded via and additional incremental encoder.

The commutation angle, the velocity and the position can be recorded by an encoder directly mounted on the motor; the position can be recorded by an encoder mounted on the load side.

For DC motors, incremental encoders are generally used for recording the position and velocity. The motor speed and position can be recorded by an encoder directly mounted on the motor. The position can optionally also be recorded by an encoder mounted on the load side.

The selection of the sensor systems to be used is set by means of entries in the object 0x2330.

Tab. 37: Switch settings of the actual value switch in object 0x2330

Value	Commutation angle (0x2330.01)[1]	Velocity (0x2330.02)	Position (0x2330.03)
00	Not selected		
01	Analogue Hall sensors [a]		
02	Digital Hall sensors [b]		
03	Digital Hall sensors + motor encoders[2]	Incremental encoder (sensor connection)	
04	AES/SSI encoder		
05	Not supported	Incremental encoder (sensor connection)	
06	Not supported		Incremental encoder (I/O connection)
07	Not supported	AnIn1 [c]	
08	Not supported	AnIn2 [c]	
09–12	Not supported	Reserved	
13	Not supported	PwmIn [c]	

a) Only for BL motors and linear motors.

b) If digital Hall sensors are used without incremental encoders, the only commutation type available is block commutation.

c) Analogue inputs and the PWM input must be converted into a suitable actual value by means of user-defined scaling.

The selection of the sensor systems to be used is set by means of entries in the object 0x2330. The settings for the sensor systems must be configured to suit in advance (see chap. 4.6.4, p. 58). If the motor and sensor system are configured using the Motion Manager motor selection wizard, all the settings will be correctly pre-set.

4.8.1.1 Examples of the selection of actual values

Set the AES encoder as the source for the commutation angle and actual value of the velocity:

- In the object 0x2330.01, set the value to **04**.
- In the object 0x2330.02, set the value to **04**.
- The AES encoder is set as the source for the commutation angle and actual value of the velocity.

Configuring a tacho sensor as the source for the actual velocity:

A tacho sensor is to be configured as the source for the actual velocity. At 5000 min^{-1} the tacho sensor delivers a voltage of 10 V. Under the standard setting the 10 V voltage at AnIn is converted into a numeric value of 10,000.

The analogue value must be converted into a numerical value suitable for the speed controller. The internal scaling for the analogue speed setpoint is n [min^{-1}].

- ✓ The tacho sensor is connected to AnIn1 or AnIn2.
- Calculate the scaling factor for the analogue input:
  - The raw value of the analogue input / internal numeric value for the voltage at 10 V = 5000/ 10000 = 1/2
- Configure the scaling factor for the analogue input in the object 0x2313:
  - Depending on the analogue input used, set the value **0x00 01 00 02** for the scaling factor 1/2 in the object 0x2313.01 or 0x2313.11
  - Ensure that depending on the analogue input used, the value**0x00** is set for the offset in the object 0x2313.02 or 0x2313.12.
  - In the object 0x2330.02, configure the value **7 or 8** for the analogue input used as the source for the actual velocity.
- The tacho sensor is now set as the source for the actual velocity.

4.8.2 Selection of discrete setpoint values

In the ATC, AVC, APC operating modes and in Voltage mode, a discrete source such as an analogue input can be selected as the setpoint value. To do this, the source must be selected in the entries of the object 0x2331.

The source to be used must be selected in advance via the entries in the object 0x2331.

Tab. 38: Selection of discrete sources in object 0x2330 (discrete references)

Value	Voltage (0x2330.01)	Torque (0x2330.02)	Velocity (0x2330.03)	Position (0x2330.04)
00	No source selected			
01	AnIn1 a)			
02	AnIn1 a)			
03-06	Reserved			
07	PwmIn a)			
08	Not supported	Not supported	Motor encoder	
09	Not supported	Not supported	Not supported	Additional encoder b)
10	Reserved			

a) *Analogue inputs and the PWM input must be converted via the user-defined scaling into a suitable setpoint.*

b) *The position of the additional encoder can be converted via the user-defined scaling within the object 0x2316.04 into a value suitable for the actual position value.*

4.8.2.1 Examples for selection of discrete setpoint values

Setting an additional encoder with incremental encoder as the target position source (gearing mode)

Analogue Hall sensors with a resolution of 4096 increments per motor revolution are used as actual value encoders. The target position should be specified via a quadrature signal of a master encoder. The connected motor should perform one revolution for 1000 increments of the external encoder. This corresponds to 4096 increments of the analogue Hall sensor.

✓ The additional encoder is configured as an incremental encoder via the object 0x2316 (see chap. 4.6.4, p. 58).

- Configure the scaling of the setpoint value to suit (denominator = 4096, numerator = 1000):
  - Set the value **0x10 00 03 E8** in the object 0x2316.04.
- Configure the APC operating mode:
  - Set the value **–2** in the object 0x6060.00.
- Select the additional encoder as the target position source:
  - Set the value **09** in the object 0x2331.04.

↳ The additional encoder with incremental encoder is now configured as the target position source.

Configuring the additional encoder connected to the pulse generator as the target position source (stepper mode)
Analogue Hall sensors with a resolution of 4096 increments per motor revolution are used as actual value encoders. The connected motor should perform one revolution for 1000 increments of the external encoder. This corresponds to 4096 increments of the analogue Hall sensor.

✓ The additional encoder is configured as a pulse counter via the object 0x2316 (see chap. 4.6.4, p. 58).

- Configure the scaling of the setpoint value to suit (denominator = 4096, numerator = 1000):
  - Set the value **0x10 00 03 E8** in the object 0x2316.04.
- Configure the APC operating mode:
  - Set the value **–2** in the object 0x6060.00.
- Select the additional encoder as the target position source:
  - Set the value **09** in the object 0x2331.04.

↳ The additional encoder with connected pulse generator is now configured as the target position source

Example: Analogue setpoint value

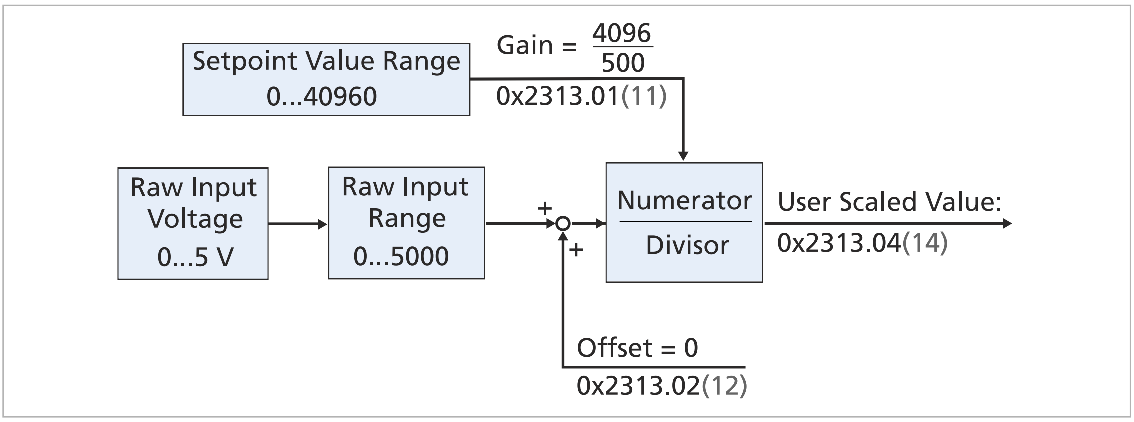


Fig. 22: Setting an analogue input for discrete setpoints

A voltage set by a potentiometer is to be used as the target position. The minimum voltage is 0 V, the maximum voltage is 5 V. With no application of user-defined scaling this corresponds to a numeric range of 0 ... 5,000. The motor position is resolved via the motor encoder with 4096 increments per revolution. This should be able to cover a setting range of 10 revolutions. The setpoint value range must therefore be from 0 ... 40960.

- Configure the user-defined scaling.
  - Depending on the analogue input, set an offset of 0 in the object 0x2313.02 or 0x2313.12.
  - Set the gain in the object 0x2313.01 or 0x2313.11:

    Gain = Max target position / Max raw numeric value = 40960 / 5000 must be cut back: 4096 / 500
- Set the value **–2** in the object 0x6060.00.

  ↳ The APC operating mode is selected.

- Configure a discrete target position source:
  - For AnIn1, set the value 01 in the object 0x2331.04.
  - For AnIn2, set the value 02 in the object 0x2331.04.

↳ The analogue input is now set as the input for the discrete setpoint values.

4.9 Data record management

The configuration settings performed in the Motion Manager can be permanently saved in the controller, so that they are available when the drive is switched on again.

4.9.1 Saving and restoring parameters via the Motion Manager

Saving parameters:
The configuration of a drive can be saved as a file for backup or for configuration of other drives.

The Motion Manager allows ongoing drive configurations to be read via the object browser and saved as an XDC file (XML device configuration file).

Loading parameters to a drive:
The Motion Manager allows previously saved XDC files to the opened in the object browser, edited as necessary and loaded to the drive.

The save command allows the loaded parameter records to the permanently saved in the drive.

4.9.2 Saving the parameter set in the drive

All or part of the parameter set can be saved continuously in the parameter memory of the Motion Controller by a write access to the object 0x1010.xx. They are thus available directly after the start.

- Write the “Save” signature (0x65 0x76 0x61 0x73) to one of the following sub-indices of the object 0x1010 (see Communications Manual):
  - .01: Saves all parameters
  - .02: Saves only the communications parameters
  - .03: Saves only the application parameters

4.9.3 Restore factory settings

The factory settings can be restored by a write access to object 0x1011. After a restore has been performed, at the next start of the controller the parameters have the as-delivered values.

- Write the “Load” signature (0x64 0x61 0x6F 0x6C) to one of the following sub-indices of the object 0x1011 (see Communications Manual):
  - .01: Resets all parameters
  - .02: Resets only the communications parameters
  - .03: Resets only the application parameters
  - .04: Loads the most recently saved application parameters into the current job

4.9.4 Switching between different application parameter sets

Part of the control parameters can be saved as parameter sets App1 and App2. The Reload command from object 0x1011 allows these two parameter sets to be exchanged dynamically.

Example
Switching the control parameters of a gripper application the operation with gripped item (App1) and without gripped item (App2).

5 Selecting the operating mode

The Modes of Operation parameter allows selection of the active drive profile. The current operating mode can be viewed in the Modes of Operation display.

The active operating mode (0x6061) is not always the same as the configured operating mode (0x6060). The PP and PV operating modes are not started until the first setpoint value is entered after the choice of operating mode. The Homing operating mode is not started until the rising flank of bit 4 in the controlword after the operating mode has been set via 0x6060.00 = 6.

Modes of Operation

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6060	0x00	Modes of operation	S8	rw	0	Select the operating mode –4: ATC –3: AVC –2: APC –1: Voltage Mode 0: Controller not activated 1: PP 3: PV 6: Homing 8: CSP 9: CSV 10: CST

Modes of Operation Display

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6061	0x00	Modes of Operation Display	S8	ro	–	Display of the selected operating mode

FAULHABER Motion Controllers support the following operating modes:

- Operating modes for position control:
  - Profile Position mode: Position control where the target position is achieved via a movement profile.
  - Cyclic Synchronous Position mode: cyclic synchronous position control
  - Analogue Torque Control mode: cyclic torque control with analogue setpoint value specification
- Operating modes for velocity control:
  - Profile Velocity mode: Velocity control where the target velocity is achieved via a velocity profile.
  - Cyclic Synchronous Velocity mode: cyclic synchronous velocity control
  - Analogue Velocity Control mode: cyclic velocity control with analogue setpoint value specification
- Operating modes for torque:
  - Cyclic Synchronous Torque mode: cyclic synchronous torque control
  - Analogue Torque Control mode: cyclic torque control with analogue setpoint value

specification

- Operating modes for referencing:
  - Homing mode: Operating mode for referencing the drive position
- Operating mode with direct voltage output:
  - Voltage mode: Direct output of the voltage at the motor, specified either via the communications system or via an analogue input.

5.1 Switching operating modes

The behaviour at the start of an operating mode depends on which operating mode was active previously.

5.1.1 Initial start of an operating mode

Even if no operating mode is activated, the state machine of the drive can already be in the *Operation Enabled* state. Changing the operating mode is requested via the object 0x6060.

Initial start for the APC, AVC and ATC operating modes
If the state machine is in the *Operation Enabled* state:

- Control is started as soon as the operating mode is selected.
- The last setpoint value channel that was set is evaluated again.

If the state machine is not in the *Operation Enabled* state:

- The operating mode can be selected.
- Control is not started until the state machine has been set to the *Operation Enabled* state.

Initial start for the CSP, CSV and CST operating modes
If the state machine is already in the *Operation Enabled* state:

- Control is started as soon as the operating mode is selected.
- Either the actual position (CSP) or 0 (CSV, CST) is specified as the setpoint value.
- No setpoint is loaded until after the switch into the operating mode.

If the state machine is not in the *Operation Enabled* state:

- The operating mode can be selected.
- Control is not started until the state machine is set to the *Operation Enabled* state.
- Either the actual position (CSP) or 0 (CSV, CST) is specified as the setpoint value.
- No setpoint is loaded until after the switch into the operating mode.
- Movement towards the new setpoint value starts immediately, once at least one valid setpoint value has been written after the drive has switched to the *Operation Enabled* state.

Initial start for the PV operating mode:
If the state machine is already in the *Operation Enabled* state:

- Control does not start until a setpoint value has been written after the selection of the PV operating mode.
- The movement profile is then calculated and activated, having regard to the current acceleration and braking ramps.

If the state machine is not in the *Operation Enabled* state:

- The operating mode can be selected.
- Control is not started until after the operating mode has been selected and a setpoint value written the state machine has been set to the *Operation Enabled* state.

Initial start for the PP operating mode:
If the state machine is already in the *Operation Enabled* state:

- Control does not start until a setpoint value has been signalled as validly written via a rising flank in bit 4 in the controlword after the selection of the PP operating mode.
- The movement profile is then calculated and activated, having regard to the current acceleration and braking ramps.

If the state machine is not in the *Operation Enabled* state:

- The operating mode can be selected.
- Setpoint values can be signalled as validly written via a rising flank in bit 4.
- If multiple setpoint values are loaded, the feedback must be monitored via the setpoint value acknowledge bit in the statusword.
- Control starts with the first movement job as soon as the state machine is set to the *Operation Enabled* state.

i The *Set Point Reset on Change of Operation Mode* bit in the Special Settings (0x233F) allows a setting which causes the operating modes to start immediately with a standard setpoint. If this bit is set the operating modes PP and PV also start immediately with the standard setpoint.

5.1.2 Reactivating the operating mode

An active operating mode can be interrupted by switching out of the *Operation Enabled* state. To reactivate the operating mode, the state machine of the drive must be reset to the *Operation Enabled* state.

Reactivating the APC, AVC and ATC operating modes

- Once the state machine is set to the *Operation Enabled* state, control starts immediately.
- The last setpoint value channel that was set is evaluated again.

Reactivating the CSP, CSV and CST operating modes

- Once the state machine is set to the *Operation Enabled* state, control starts immediately.
- Either the actual position (CSP) or 0 (CSV, CST) is specified as the setpoint value.
- No setpoint is loaded until after the switch into the operating mode.

Reactivating the PV operating mode

- Initially the last movement job is discarded when the drive has exited the *Operation Enabled* state.
- Once the state machine is set to the *Operation Enabled* state and a new setpoint value is written, control starts immediately.
- The sequence of these actions is irrelevant. Therefore a new setpoint value can be written first and then the state switched over, or vice versa.

Reactivating the PP operating mode

- Initially the last movement job is discarded when the drive has exited the *Operation Enabled* state. Once the state machine is set to the *Operation Enabled* state and a new setpoint value is written, control starts immediately.
- The sequence of these actions is irrelevant. If multiple setpoint values are loaded, the feedback must be monitored via the setpoint value acknowledge bit in the statusword.

5.1.3 Switching the operating mode

To switch to an active controlled operating mode with an active output stage, a new operating mode must be requested using the object 0x6060.

Switch to operating modes APC, AVC and ATC

- The new operating mode is activated immediately. The currently set setpoint value source is evaluated.

Switch to operating modes CSP, CSV and CST

- The new operating mode is activated immediately. The last setpoint value to be loaded is used.

Switch to operating mode PV

- The operating mode is pre-booked. The operating mode is not switched until a new velocity setpoint value has been specified via the object 0x60FF.
- The movement profile is then calculated and applied to the controller.
- Until the new setpoint value has been loaded, the old operating mode continues to be displayed in the object 0x6061.

Switch to operating mode PP

- The operating mode is pre-booked. The operating mode is not switched until a new position setpoint value is flagged as valid by a rising flank at bit 4 in the controlword.
- The movement profile is then calculated and applied to the controller.
- Until the new setpoint value has been loaded, the old operating mode continues to be displayed in the object 0x6061.

The *Set Point Reset on Change of Operation Mode* bit in the Special Settings (0x233F) allows a setting which causes the operating modes to start immediately with a standard setpoint. If this bit is set the operating modes PP and PV also start immediately with the standard setpoint.

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x233F	0x00	OpMode Options	U16	rw	0x0001	Bit-coded

Bit 0 - setpoint value reset on change of operating mode:

- 0: Setpoint values are not reset on a change of operating mode. In particular in operating modes with cyclical setpoint value (CSx) the last setpoint value received will be used directly for control. In the operating modes PP and PV the change into the operating mode does not occur until after the change to the first new setpoint has been written.
- 1: Setpoint values are reset on a change of operating mode.
  - For CST: Torque setpoint value = 0
  - For CSV, PV: Velocity setpoint value = 0
  - For CSP, PP: Position setpoint value = actual position

Bit 1 - use position limits as limits in Velocity mode:

- 0: The position limits from 0x607D limit merely the target position in the object 0x607A.
- 1: In Velocity mode and Torque mode the position limits from 0x607D are treated like limit switches. If a limit is reached, the drive stops.

Bit 2 - auto enable power stage:

- 0: After the start of the controller, the CiA 402 state machine finds itself in the *Switch On Disabled* status.
- 1: After the start of the controller, the drive attempts to switch the controller directly into the status *Operation Enabled*. That status enables the operating modes APC, AVC and ATC to be activated, even without any direct intervention by a sequence program or supervisory control system.

Bit 3 - immediate references are relative to the actual position:

- 0: Relative position setpoint values (move commands) are added to the last setpoint value that was received, even if they carry the “immediate” flag.
- 1: If relative move commands carry the “immediate” flag, they are added to the actual position.

Bit 4 - No Autoadjust of Pos-Limits:

- 0: Position limits are automatically limited if a numeric overflow occurs when the Factor Group is in use.
- 1: Position limits are not automatically limited. In this case the user must select limits that are sufficiently low.

5.2 Profile Position mode

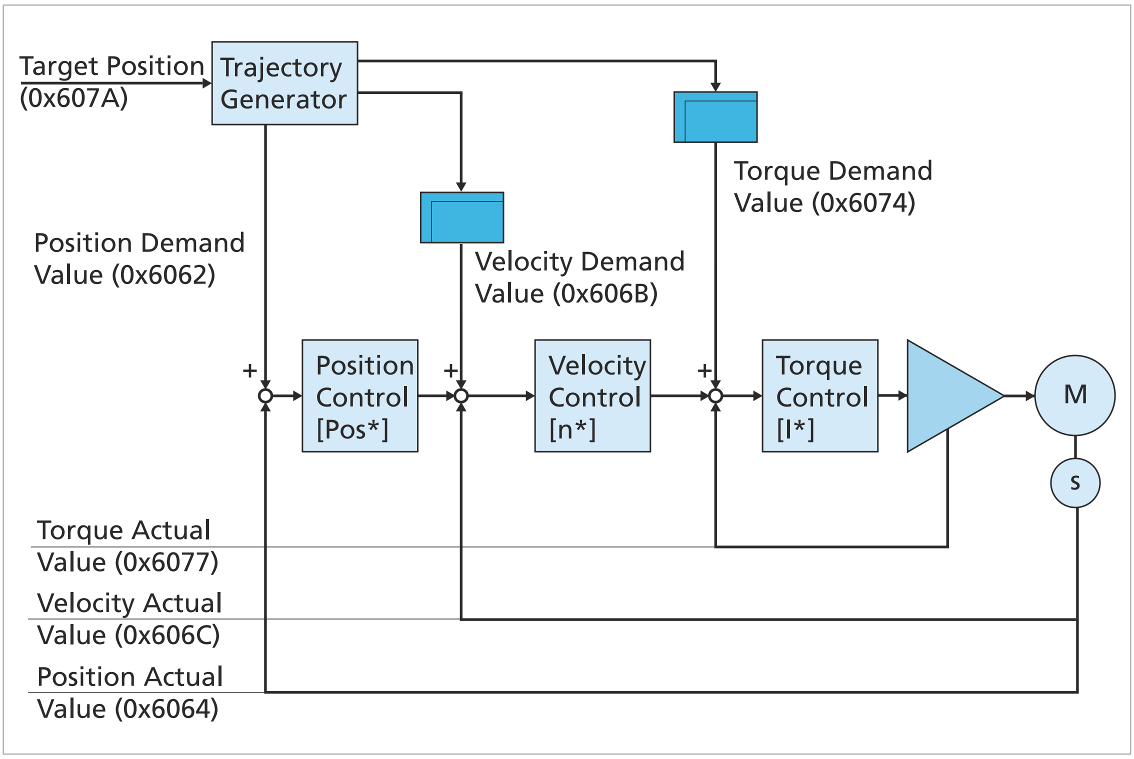


Fig. 23: Profile Position mode control loop

In the Profile Position operating mode the profile generator in the Motion Controller calculates a movement profile for position, velocity and acceleration from the target position and the maximum values for acceleration, delay and velocity.

Movement starts when the setpoints are loaded, triggered by the rising flank of bit 4 in the controlword. The target is reached when the drive has been at the target position for the period of time defined in the position window. If a set of setpoints is applicable, movement can be performed successively towards several setpoint values.

The drive performs torque control, velocity control and position control. New setpoint values or profile parameters are not processed until the previous setpoint value has been reached. By means of the Immediate bit in the controlword, new setpoint values can also be loaded immediately.

The setpoint value is specified via the object (0x607A). If the setpoint value is changed, the drive will behave in accordance with the specified profile.

The following actual values are passed to the drive by measurement:

- Position
- Velocity
- Torque

Profile Position mode contains the following sub-functions:

- Calculation of the movement profile
- Position recording
- Velocity recording

- Velocity control with corresponding input and output signals
- Torque recording
- Torque control with corresponding input and output signals
- Setpoint limitation of the torque
- Setpoint limitation of the velocity
- Monitoring the target position
- Monitoring the following error

5.2.1 Description of functions

From the point of view of the position controller the following values are inputs:

- Target Position (0x607A)
- Maximum Motor Speed (0x6080)
- Maximum Torque (0x6072)
- Profile Velocity (0x6081)
- Profile Acceleration (0x6083)
- Profile Deceleration (0x6084)
- Quick Stop Deceleration (0x6085)
- Motion Profile Type (0x6086)
- Controlword (0x6040)

The actual value of the position (object 0x6064) and the statusword (object 0x6041) are mandatory outputs of the controller.

The drive can have the following functions:

- Position controller when a movement profile is being used
- Quick Stop

5.2.2 Statusword/Controlword for profile position mode

Operating mode-specific bits are used in the controlword and statusword for the Profile Position operating mode.

If no positioning is being performed, setting bit 4 from 0 to 1 will start the positioning of the axes. If at that moment positioning is being performed, the drive behaves in accordance with Tab. 40.

Tab. 39: Operating mode-specific bits of the controlword (Profile Position mode)

Bit	Function	Description
4	New setpoint value	0: Do not start profile positioning. 1: See Tab. 40.
5	Change set immediately	0: Movement towards the new position does not start until the preceding positioning job has been completed. 1: Movement towards the new position value starts immediately. New setpoint values overwrite the old setpoint value.
6	Abs/Rel	0: Position setpoint value is an absolute value. 1: Position setpoint value is a relative value.
9	Change on set-point value	0: A new movement task is not loaded until the drive has reached its previous target position. 1: The drive brakes or accelerates to the velocity profile of the next setpoint value. Movement towards the new setpoint value starts without any preceding stop.

Tab. 40: Meaning of bits 4, 5, 9 in the controlword

Bit 9	Bit 5	Bit 4	Meaning
0	0	0 → 1	Movement towards the new position does not start until the preceding positioning job has been completed (Target Reached).
x	1	0 → 1	Move towards the new position immediately.
1	0	0 → 1	The current movement is maintained. If the previous target position had already been reached, the drive switches immediately to the next setpoint value as defined by the movement profile.

1 = bit set
0 = bit not set
X = bit not used (status irrelevant)
0 → 1 = rising flank of the bit

Tab. 41: Operating mode-specific bits of the statusword (profile position mode)

Bit	Function	Description
10	Target Reached	0: Stop (bit 8 in the controlword) = 0: Target position not reached. 0: Stop (bit 8 in the controlword) = 1: The drive brakes to a stop.
		1: Stop (bit 8 in the controlword) = 0: Target position reached. 1: Stop (bit 8 in the controlword) = 1: Drive at a standstill.
12	Setpoint acknowledge	0: Previous setpoint value being changed or already reached. 1: New setpoint value has been loaded.
13	FollowingError	0: The actual position follows the instructions without a following error. 1: Permissible range for the following error exceeded.

Information on the compiled movement profile can be found in chap. 4.4.1, p. 45.

5.2.3 Settings

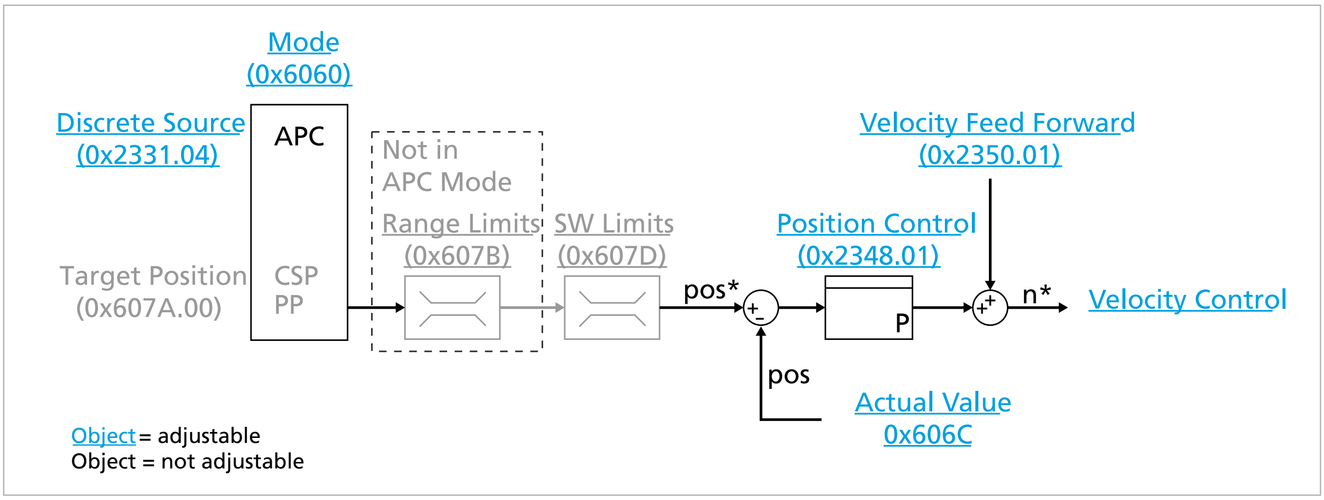


Fig. 24: Motion Manager view of Profile Position mode

The following objects must be set when using this operating mode:

- Operating mode (0x6060 = 1)
- Target Position (0x607A)
- Profile Velocity (0x6081)
- Maximum Motor Speed (0x6080)
- Profile Acceleration (0x6083)
- Profile Deceleration (6084)

Fig. 25 shows all the objects that are effective in this operating mode. The objects shown additionally permit optional settings within this operating mode.

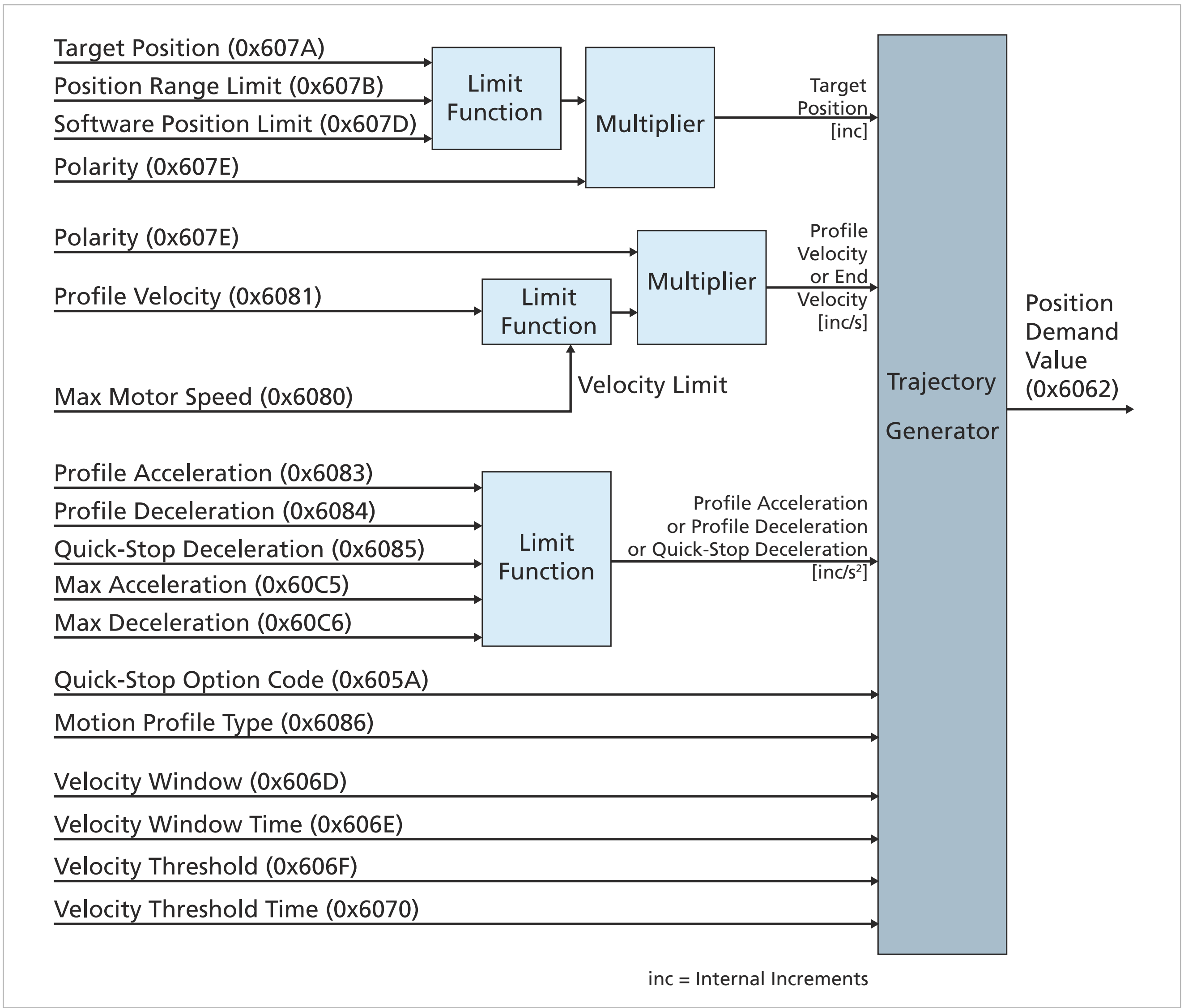


Fig. 25: Overview of all the objects that are effective in PP operating mode

5.2.4 Examples

5.2.4.1 Example 1 (positioning with absolute setpoint values, followed by reversing)

The drive shall be moved by 12,000 increments. After a brief wait, it is to return back to position 0.

✓ The state machine must be in the *Operation Enabled* state.

✓ The actual position must be zeroed by means of a reference run.

✓ The software position ranges and software range limits must lie outside the range 0 ... 12,000.

1. Select the PP operating mode:
   - In the object 0x6060.01, set the value to **01**.
2. Set the setpoint value and profile parameters:
   - In the object 0x607A.00, set the value to **12000**.
   - In the object 0x6083.00, set the value to **1000**.
   - In the object 0x6084.00, set the value to **1000**.
3. Mark the setpoint value as an absolute setpoint value, and start the move command:
   - In the object 0x6040, set the value to **0x00 1F**.

   The drive will acknowledge the setpoint value as accepted via bit 12 (0x6041 = 0x1027).
4. Reset the start bit again in the controlword.
   - In the object 0x6040, set the value to **0x00 0F**.

   The drive confirms its readiness to accept further setpoint values by resetting the Setpoint Acknowledge bits (0x6041 = 0x0027).

   The drive moves to the target position and, after the Position Window Time has elapsed, signals in bit 10 (0x6041.00 = 0x0427) that the target position has been reached.

   The drive is moved by 12,000 increments.
5. Reset the setpoint value for the return movement.
   - In the object 0x607A.00, set the value to **00 00 00 00**.
6. Repeat steps 3 and 4.

↳ Move the drive by 12,000 increments and then move it back again.

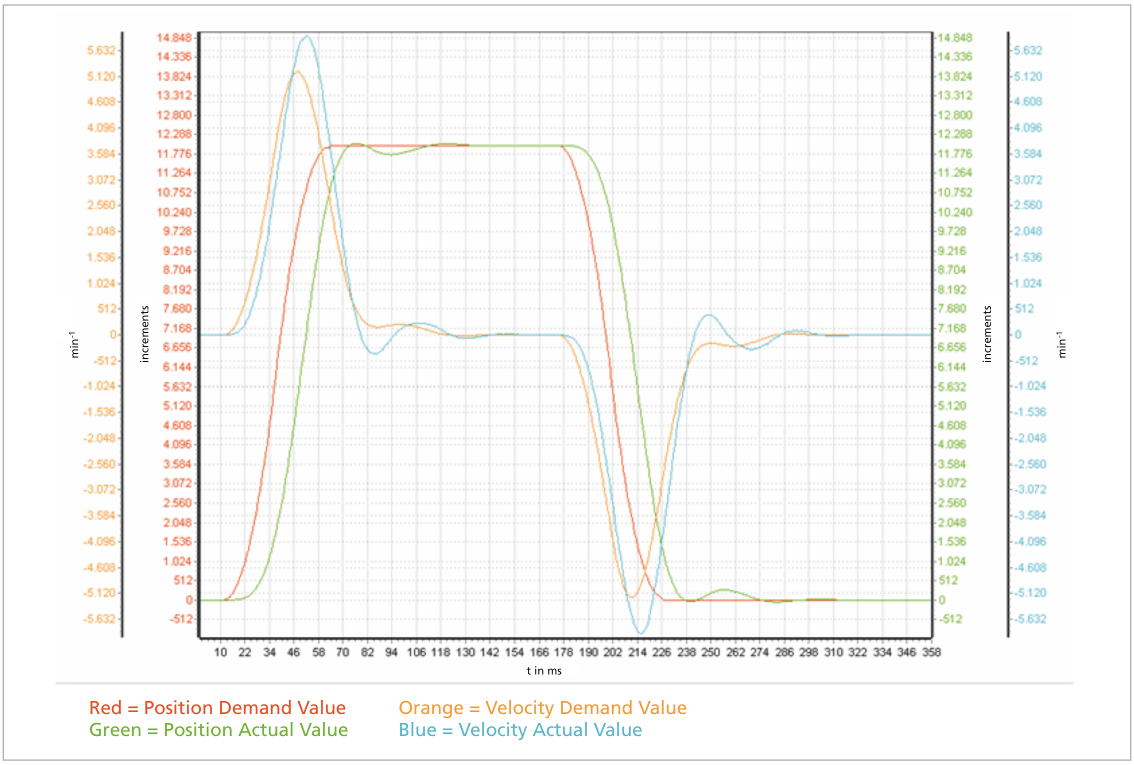


Fig. 26: Movement profile of a positioning operation with absolute setpoint values, followed by reversing

5.2.4.2 Example 2 (positioning with relative setpoint values, followed by reversing)

A drive shall be moved by 12,000 increments relative to the current position.

To avoid oscillations at the resiliently coupled mechanism, only gentle acceleration is employed and in addition the Sin² velocity profile is selected.

✓ The state machine must be in the *Operation Enabled* state.

✓ The actual position must be zeroed by means of a reference run.

✓ The software position ranges must lie around the current position and outside the range 0 ... 12,000.

1. Select the PP operating mode:
   - In the object 0x6060.01, set the value to **01**.
2. Set the setpoint value and profile parameters:
   - In the object 0x607A.00, set the value to **12000**.
   - In the object 0x6083.00, set the value to **1000**.
   - In the object 0x6084.00, set the value to **1000**.
   - In the object 0x6086.00, set the value to **1**.

3. Mark the setpoint value as a relative setpoint value and start the move command:
   - In the object 0x6040, set the value to **0x00 5F**.

   The drive will acknowledge the setpoint value as accepted via bit 12 (0x6041 = 0x1027).
4. Reset the start bit again in the controlword.
   - In the object 0x6040, set the value to **0x00 0F**.

   The drive confirms its readiness to accept further setpoint values by resetting the Setpoint Acknowledge bits (0x604 = 0x0027).

   The drive moves to the target position and, after the Position Window Time has elapsed, signals in bit 10 (0x6041.00 = 0x0427) that the target position has been reached.

   The drive is now in the position +12,000 increments relative to the start position.
5. Reset the setpoint value for the return movement.
   - In the object 0x607A.00, set the value to 12,000.
6. Repeat steps 3 and 4.

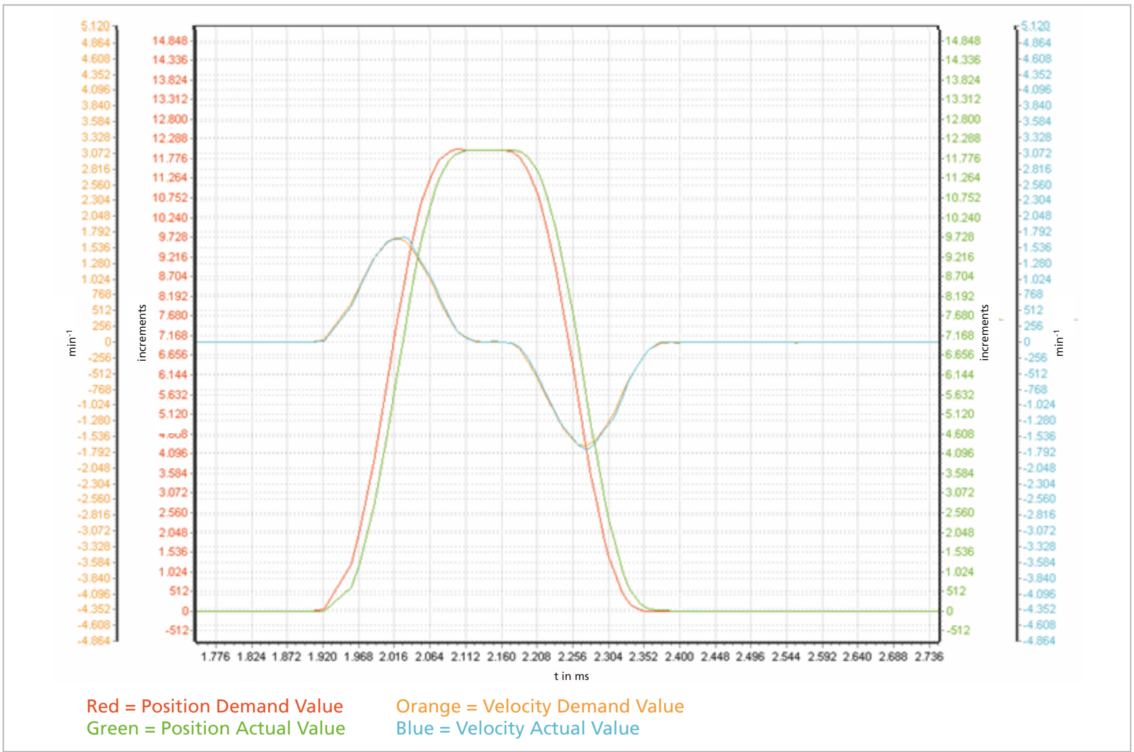


Fig. 27: Movement profile of a positioning operation with relative setpoint values, followed by reversing

5.2.4.3 Example 3

One axis is to move to the position 32,678 increments. After this it is to move immediately to the position 34,816. Finally it is to move back to the position 0.

To avoid oscillations at the resiliently coupled mechanism, only gentle acceleration is employed and in addition the Sin² velocity profile is selected.

✓ The state machine must be in the *Operation Enabled* state.

✓ The actual position must be zeroed by means of a reference run.

✓ The software position ranges and software range limits must lie outside the range 0 ... 34816.

1. Select the PP operating mode:
   - In the object 0x6060.01, set the value to **01**.
2. Set the setpoint value and profile parameters:
   - In the object 0x607A.00, set the value to **32678**.
   - In the object 0x6083.00, set the value to **100**.
   - In the object 0x6084.00, set the value to **100**.
   - In the object 0x6086.00, set the value to **1**.
3. Adapt the Position Window time to the application:
   - In the object 0x6068.00, set the value to **100**.
4. Mark the setpoint value as an absolute setpoint value, and start the move command:
   - In the object 0x6040, set the value to **0x00 1F**.

   The drive acknowledges the setpoint value as accepted via bit 12 (0x6041 = 0x1027)

   The drive starts with the first move command.
5. Reset the start bit again in the controlword.
   - In the object 0x6040, set the value to **0x00 0F**.

   The drive confirms its readiness to accept further setpoint values by resetting the Setpoint Acknowledge bits (0x6041 = 0x0027).
6. Write and activate the second setpoint value:
   - In the object 0x607A.00, set the value to **34816**.
   - In the object 0x6040.00, set the value to **0x00 1F**.

   The drive acknowledges the setpoint value as accepted via bit 12 (0x6041 = 0x1027)
7. Reset the start bit again in the controlword.
   - In the object 0x6040, set the value to **0x00 0F**.

   The drive confirms its readiness to accept further setpoint values by resetting the Setpoint Acknowledge bits (0x6041 = 0x0027).

8. Write and activate the third setpoint value.
   - In the object 0x607A.00, set the value to **34816**.
   - In the object 0x6040.00, set the value to **0x00 1F**.

   The drive acknowledges the setpoint value as accepted via bit 12 (0x6041 = 0x1027)
9. Reset the start bit again in the controlword.
   - In the object 0x6040, set the value to **0x00 0F**.

   By failing to reset the Setpoint Acknowledge bit (Bit 12), the drive signals that it cannot accept any further setpoint values.

   The first target is now reached. The drive starts to perform the 2nd move command. This releases a setpoint value buffer. The drive confirms its readiness to accept further setpoint values by resetting the Setpoint Acknowledge bits (0x6041 = 0x0027). Since there are still further setpoint values outstanding, the target is flagged as not yet reached.

   The second target is now reached. The drive starts to perform the 3rd move command. The drive moves to the target position and, after the Position Window Time has elapsed, signals in bit 10 (0x6041.00 = 0x0427) that the target position has been reached.

↳ The drive is now once again at the 0 increments position.

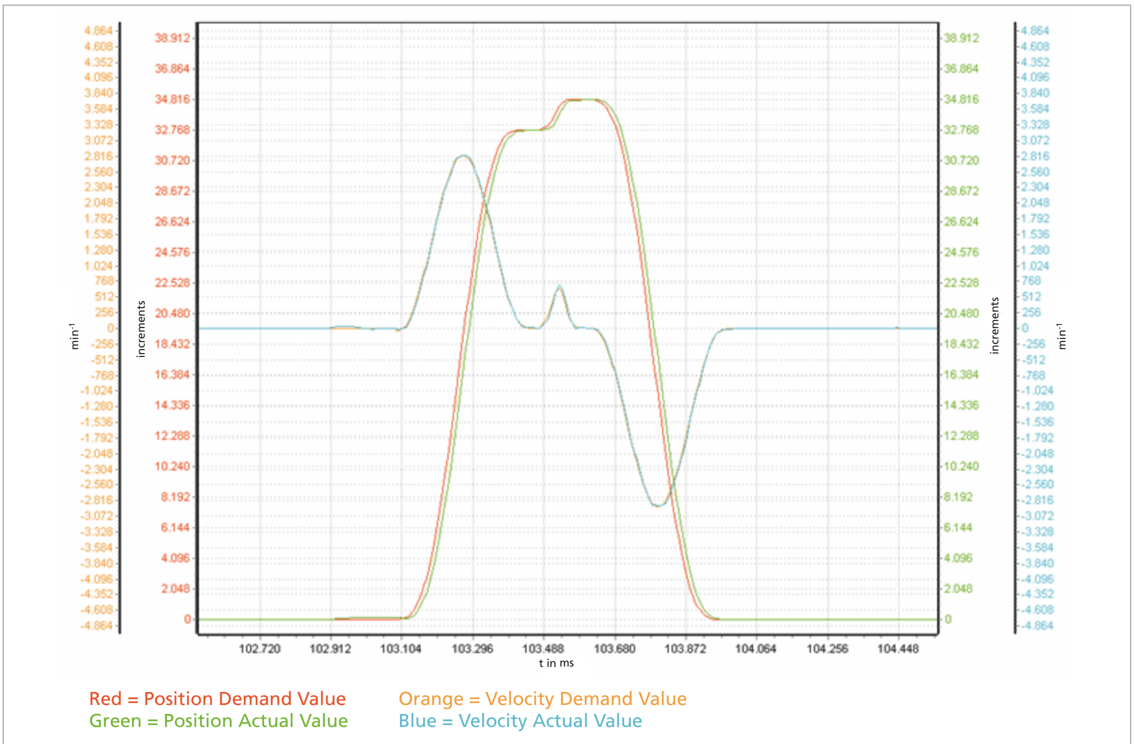


Fig. 28: Movement profile of a positioning operations with multiple absolute setpoint values, followed by reversing

5.3 Profile Velocity mode

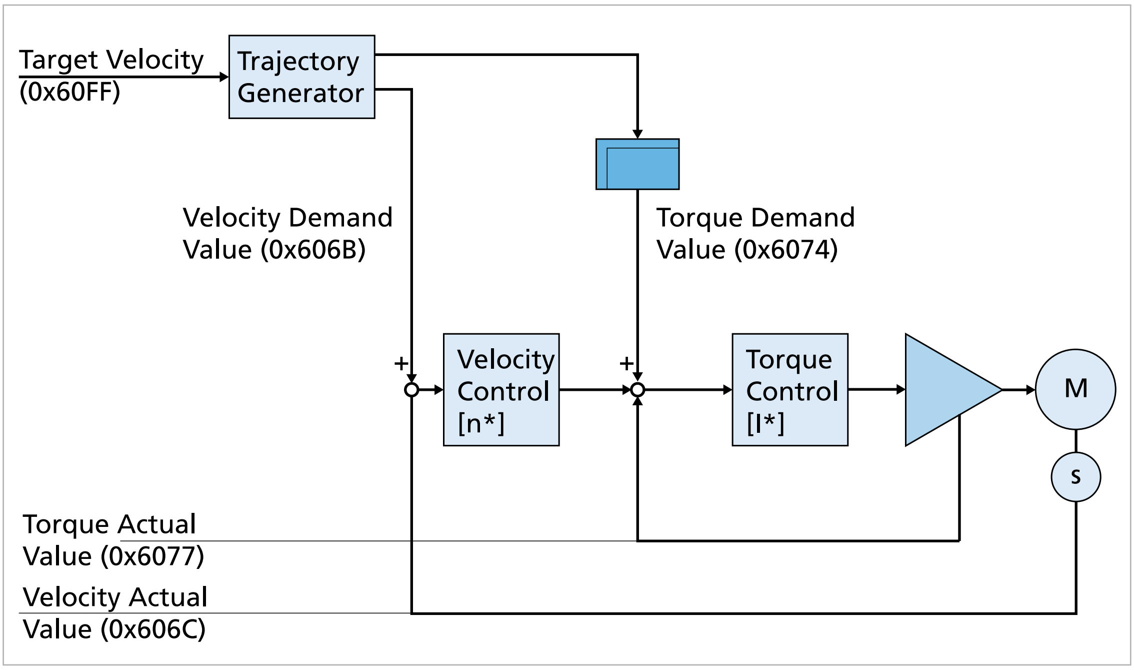


Fig. 29: Profile Velocity mode control loop

In Profile Velocity mode the velocity of the drive is controlled. The target value is specified by the object (0x60FF). When the target value is changed, the drive is accelerated or braked in accordance with the specified movement profile. The movement profile is calculated in the profile generator of the Motion Controller. This calculation ensures compliance with the specified values for the acceleration, deceleration and the maximum velocity.

The following actual values are passed to the drive by measurement:

- Position
- Velocity
- Torque

Profile Velocity mode contains the following sub-functions:

- Calculation of the movement profile
- Velocity recording
- Torque recording
- Torque control with corresponding input and output signals
- Setpoint limitation of the torque
- Setpoint limitation of the velocity
- Monitoring the velocity at standstill (n=0)
- Monitoring the velocity for deviations from the setpoint value

5.3.1 Description of functions

From the point of view of the velocity controller the following values are inputs:

- Target Velocity (0x60FF)
- Maximum Motor Speed (0x6080)
- Maximum Torque (0x6072)
- Profile Velocity (0x6081)
- Profile Acceleration (0x6083)
- Profile Deceleration (0x6084)
- Motion Profile Type (6086)

The actual value of the velocity (object 0x606C) and the statusword (6041) are mandatory outputs of the controller.

The drive can have the following functions:

- Velocity control, using a movement profile
- Quick Stop

5.3.2 Statusword/Controlword for profile velocity mode

Operating mode-specific bits are used in the statusword for the Profile Velocity operating mode.

Tab. 42: Operating mode-specific bits of the statusword (profile velocity mode)

Bit	Function	Description
10	Target Reached	0: Stop (bit 8 in the controlword) = 0: Target velocity not reached 0: Stop (bit 8 in the controlword) = 1: The drive brakes to a stop
		1: Stop (bit 8 in the controlword) = 0: Target velocity reached 1: Stop (bit 8 in the controlword) = 1: Drive at a standstill
12	Speed	0: Drive is in motion 1: Drive is at a standstill
13	Maximum Slippage Error	0: Maximum permitted velocity deviation not reached 1: Maximum permissible deviation between setpoint value and actual value reached.

In the controlword the Profile Velocity operating mode does not use any operating mode-specific bits.

5.3.3 Settings

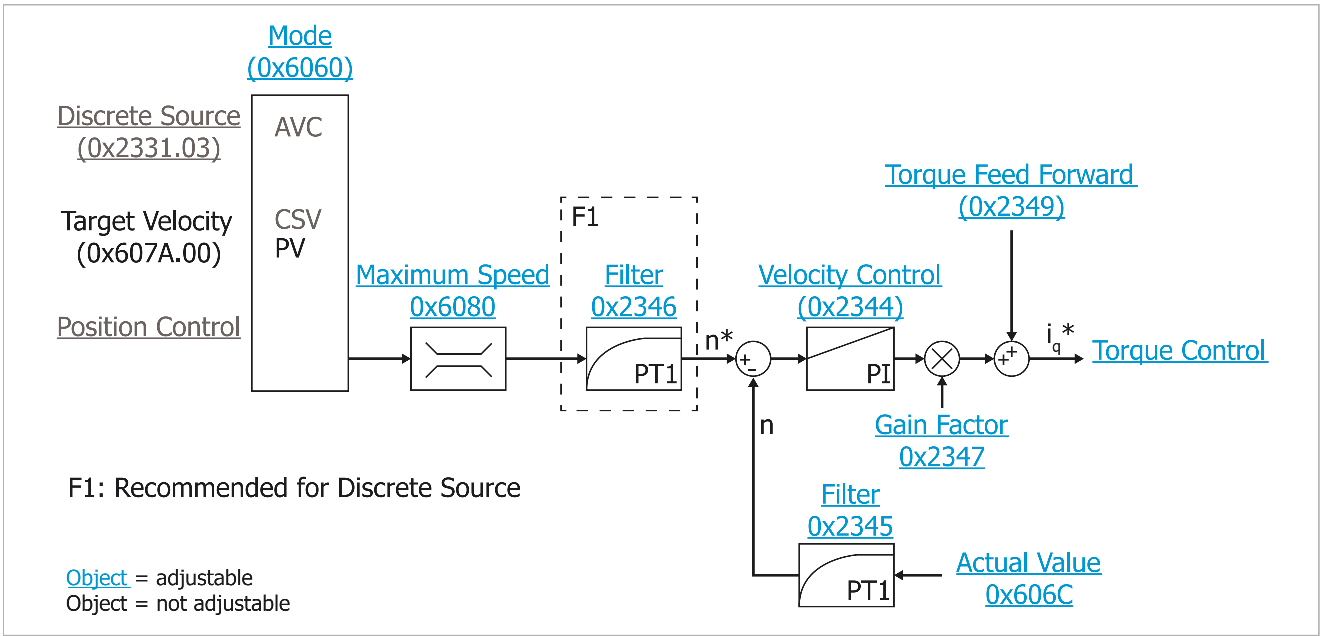


Fig. 30: Motion Manager view of the Profile Velocity mode

The following objects must be set when using this operating mode:

- Operating mode (0x6060 = 3)
- Target Velocity (0x607A)
- Profile Velocity (0x6081)
- Maximum Motor Speed (0x6080)
- Profile Acceleration (0x6083)
- Profile Deceleration (0x6084)

Fig. 31 shows all the objects that are effective in this operating mode. The objects shown additionally permit optional settings within this operating mode.

Selecting the operating mode

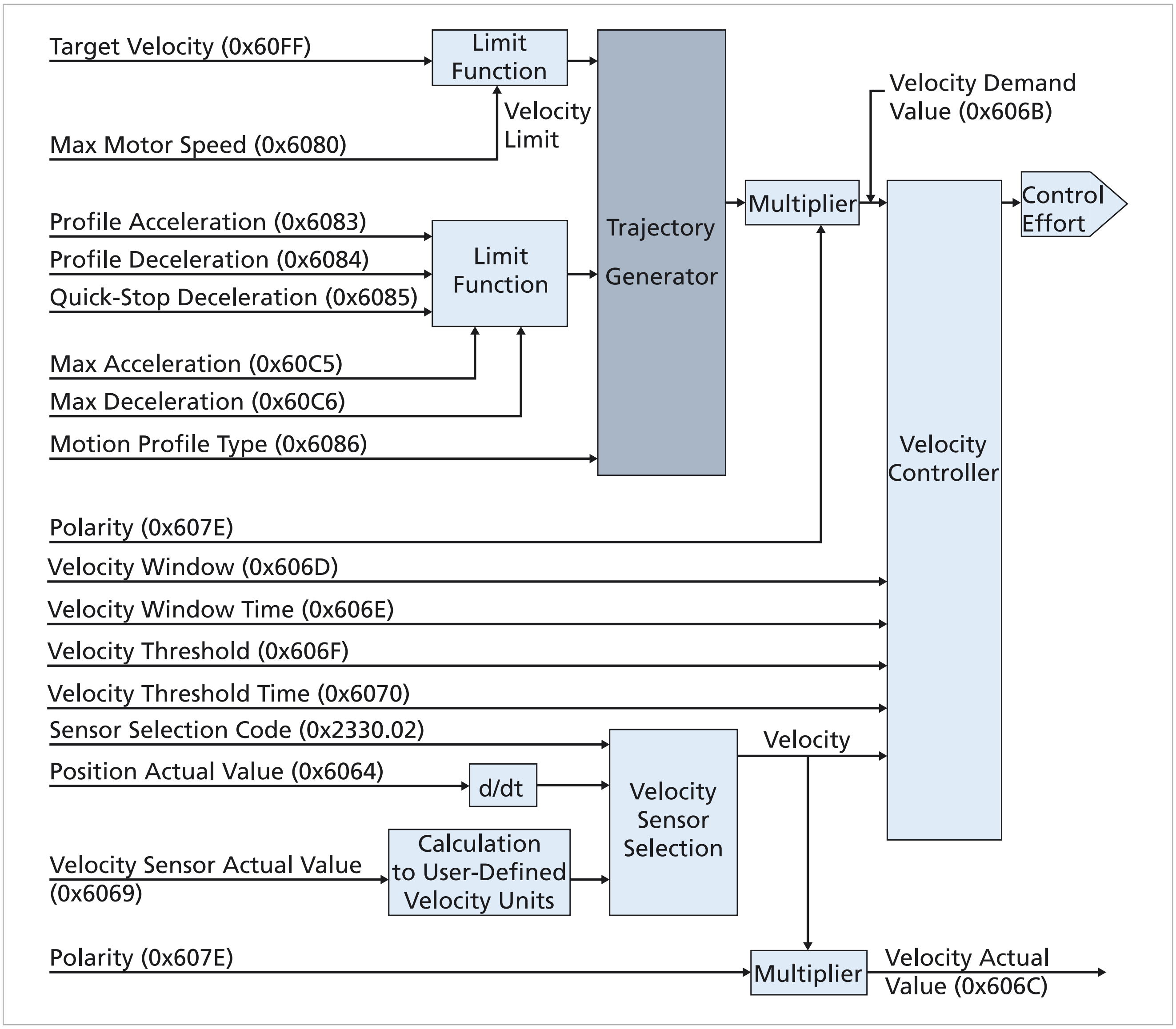


Fig. 31: Overview of all the objects that are effective in PV operating mode

5.3.4 Examples

5.3.4.1 Example 1 (reversing operation with a jerk-limited profile)

A resiliently coupled load is to be reversed from –4096 min^{-1} to +4096 min^{-1}. To avoid oscillations, braking and acceleration values are limited and a jerk-limited movement profile is selected.

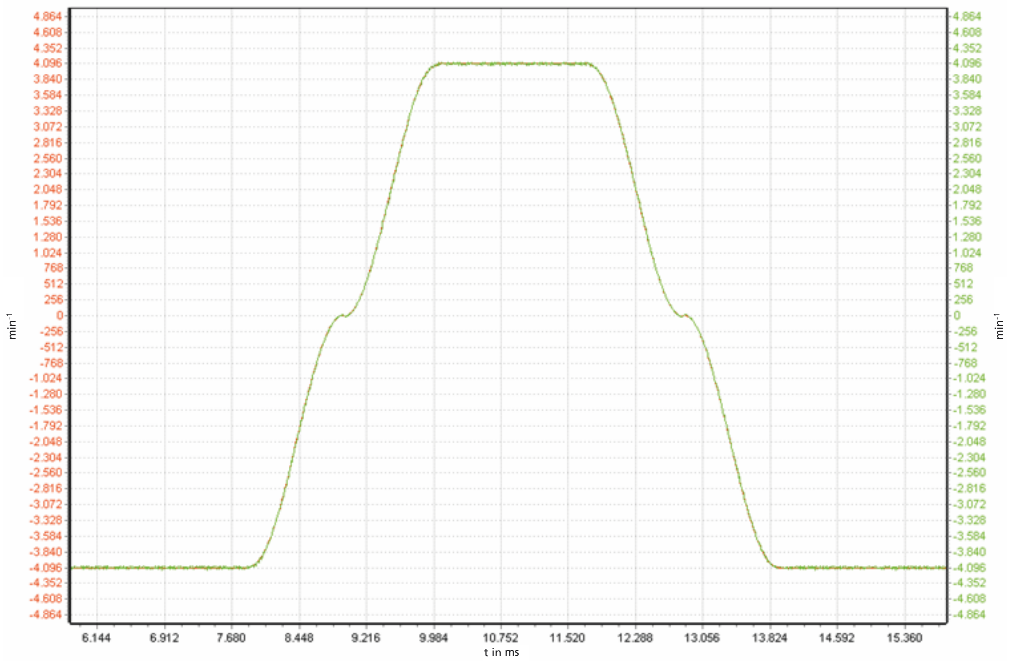
✓ The drive is switched on and is operated in a velocity-controlled mode (PV or CSV).

✓ The target velocity value is –4096 min^{-1}.

✓ To limit an initial overshoot of the controller, the setpoint velocity filter is active.

1. Configure the braking and acceleration ramp:
   - Set the value **100** in the object 0x6083.
   - Set the value **100** in the object 0x6084.
2. Select a jerk-limited profile:
   - Set the value **1** in the object 0x6086.
3. Select a new setpoint value:
   - Set the value **–4096** in the object 0x60FF.

↳ The drive stops and then starts to move again in the opposite direction.



Red = Velocity Demand Value
Green = Velocity Actual Value

Fig. 32: Movement profile of a reversing operation with a jerk-limited profile

5.3.4.2 Example 2 (acceleration from an existing movement with a limited acceleration rate)

A load has to be accelerated from 1000 min^{-1} to 5000 min^{-1}. The load is rigidly coupled.

✓ The drive is switched on and is running in velocity-controlled PV mode.

✓ The target velocity value is 1000 min^{-1}.

✓ To limit an initial overshoot of the controller, the setpoint velocity filter is active.

1. Configure the acceleration ramp:
   - Set the value **1000** in the object 0x6083.
2. Select a trapezoidal profile:
   - Set the value **0** in the object 0x6086.
3. Select a new setpoint value:
   - Set the value **5000** in the object 0x60FF.

↳ The drive will accelerate to 5000 min^{-1}.

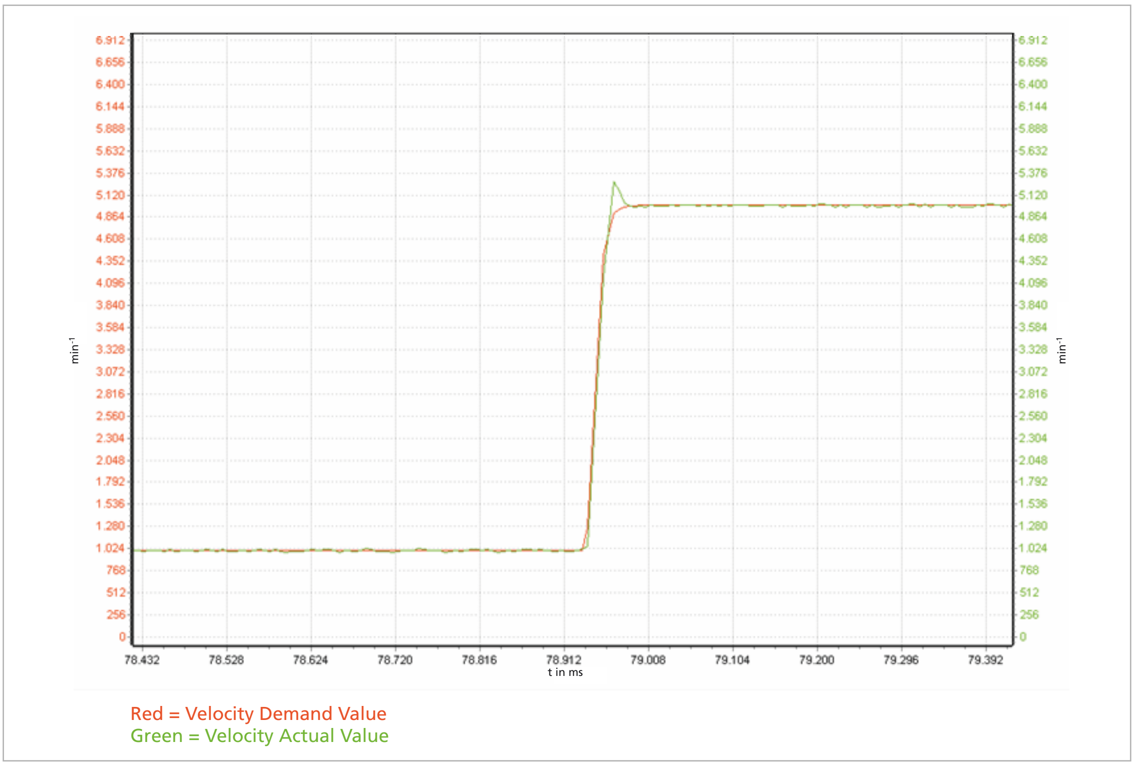


Fig. 33: Movement profile of an acceleration from an existing movement with a limited acceleration rate

5.4 Homing mode

After positioning mode has been switched on, the drive must perform a reference run (homing) to align the position value at the system. To achieve this, the position can be compensated as follows:

- With an explicitly provided reference switch
- With one of the limit switches
- At the current position
- By a run to a mechanical block (block run)

The exactness of the reference position can be refined using an index signal. The object 0x2310 (see chap. 4.6.1, p. 54) allows selection of which inputs should be used as a limit switch or reference switch.

The following homing methods are supported, based on CiA 402:

- 1 to 14 (homing with index impulse, if available)
- 17 to 30 (homing without index impulse)
- 33, 34 (homing at the index impulse, if available)
- 37 (homing at the current position)
- –1 to –4 (homing by block run with or without index impulse).

5.4.1 Description of functions

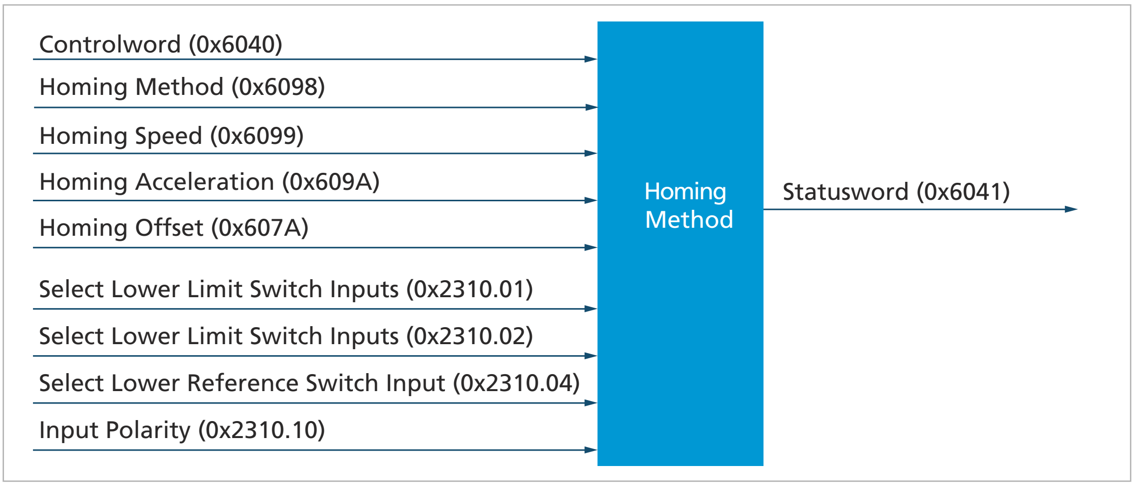


Fig. 34: Homing mode control function

Fig. 34 shows the inputs and outputs of the Homing mode. From the point of view of the controller the following objects are inputs:

- Controlword (0x6040)
- Homing Method (0x6098)
- Homing speed (0x6099)
- Homing Acceleration (0x609A)
- Homing offset (0x607A)

From the point of view of the controller the following objects are output values:

- Statusword (0x6041)

Limit switches limit the range of movement (negative/positive limit switch), but at the same time can also be used as reference switches for the zero position. A homing switch is a dedicated reference switch for the zero position.

The accuracy of the homing position can be increased if an index impulse is available in addition to the reference switch.

Methods 1 and 17

Homing to the lower limit switch (negative limit switch):

- If the limit switch is inactive, the drive moves first in the direction of the lower limit switch, until its positive flank is detected. Once the limit switch is active, the drive moves upwards away from the limit switch until the negative flank is detected. Under method 1, additionally the drive moves further to the next index impulse at which the home position is set.

Methods 2 and 18

Homing to the upper limit switch (positive limit switch):

- If the limit switch is inactive, the drive moves first in the direction of the upper limit switch, until its positive flank is detected. Once the limit switch is active, the drive moves downwards away from the limit switch until the negative flank is detected. Under method 2, additionally the drive moves further to the next index impulse at which the home position is set.

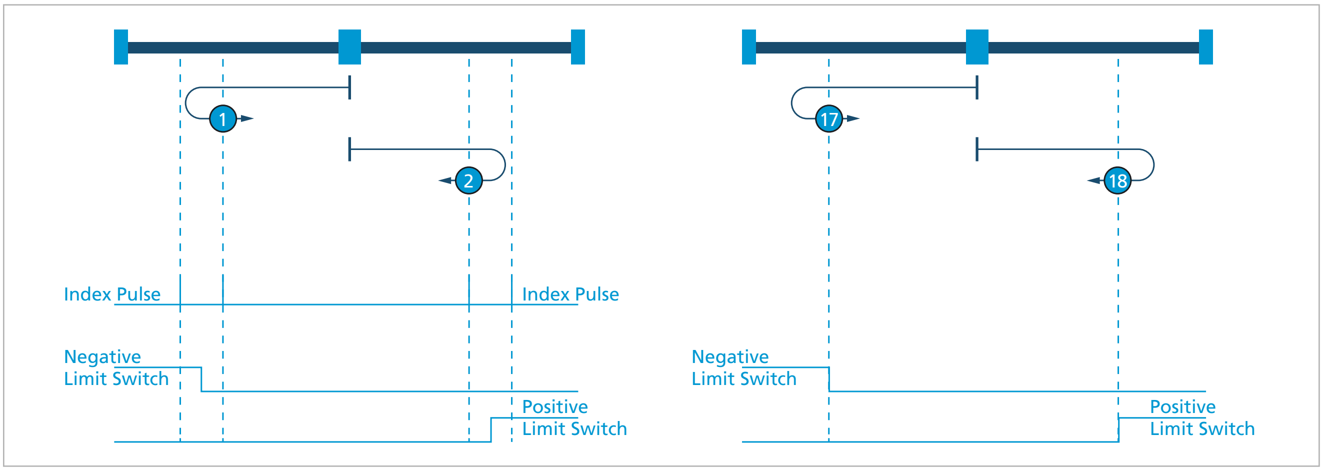


Fig. 35: Homing methods 1, 2, 17 and 18

Methods 3, 4 and 19, 20

Homing to a positive homing switch

Depending on the status of the homing switch, the drive moves in one direction until the falling (5, 21) or rising (6, 22) flank occurs. In the direction of the upper limit switch there is only one rising flank of the homing switch.

The homing position is at the point where the state of the home switch changes.

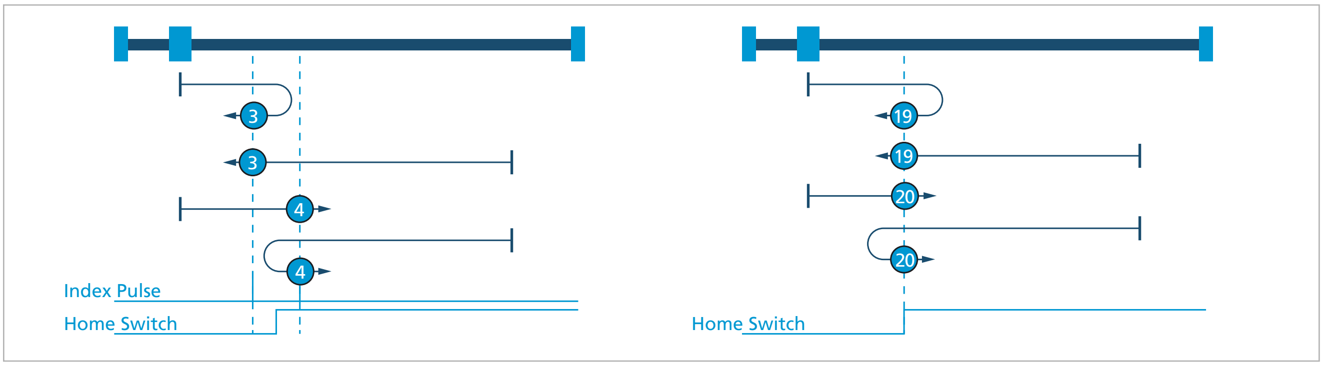


Fig. 36: Homing methods 3, 4, 19 and 20

Methods 5, 6 and 21, 22
Homing to a negative homing switch (negative home switch).

The initial direction of movement depends on the state of the home switch. The homing position is at the point where the state of the home switch changes. If during a reference run the direction of movement has to be reversed, this is always at the point where the state of the homing switch changes.

The homing position is at the point where the state of the home switch changes.

Under the methods 21 and 22 the homing position is set at a flank. Index impulses are not detected.

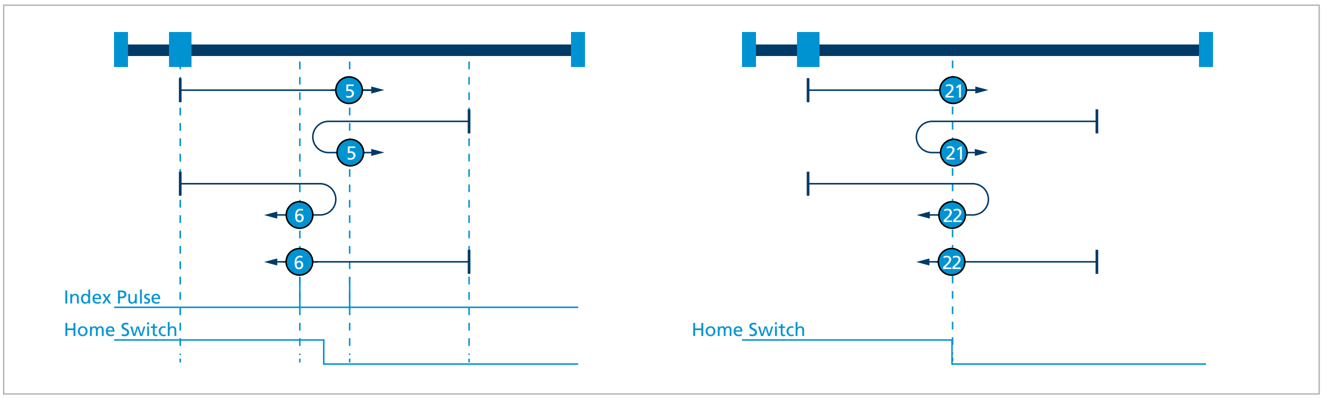


Fig. 37: Homing methods 5, 6, 21 and 22

Methods 7 to 14 and 23 to 30
Homing at the homing switch (home switch)

These methods use a homing switch which is active only in a particular range of movement. In this case the drive must respond differently to the two flanks of the home switch. Under methods 7 to 14, after detection of the flank the drive moves further to the index impulse, at which point the homing position is then set. Under the methods 23 to 30 the homing position is set at a flank. Index impulses are not detected.

- Methods 7 and 23:

  Homing at the falling flank at the bottom. Starts in a positive direction if the switch is inactive.

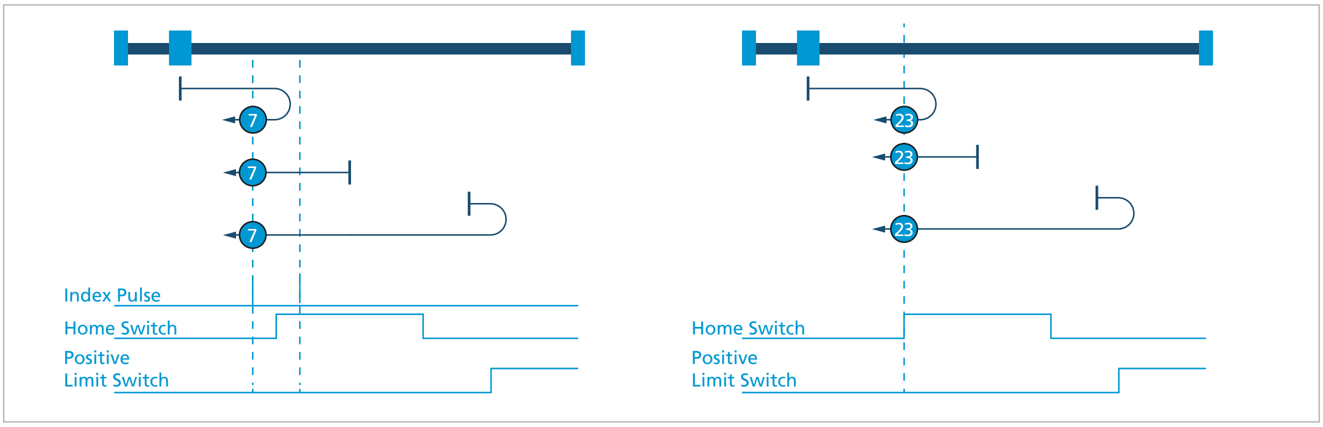


Fig. 38: Homing methods 7 and 23

- Methods 8 and 24:

  Homing at the rising flank at the bottom. Starts in a positive direction if the switch is inactive.

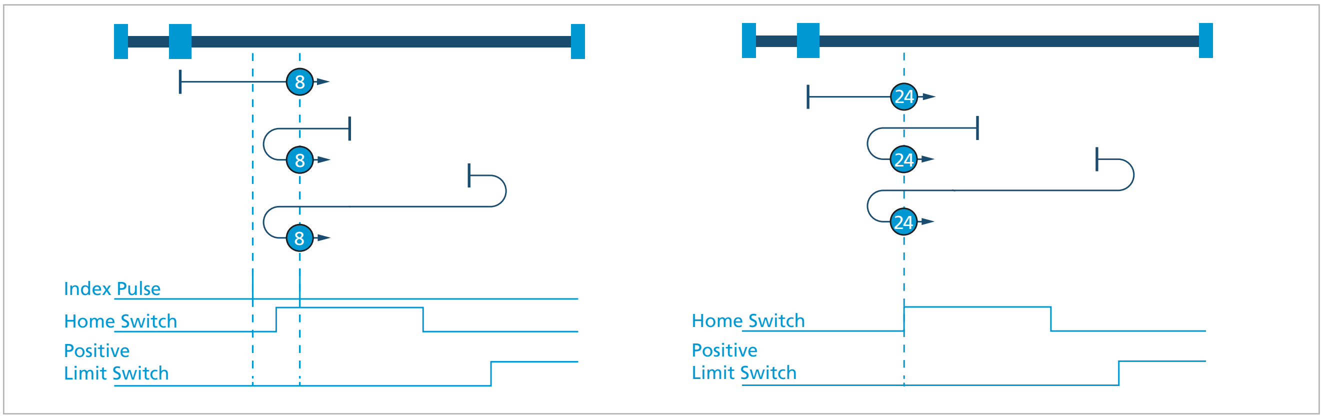


Fig. 39: Homing methods 8 and 24

- Methods 9 and 25:

  Homing at the rising flank at the top. Always starts in the positive direction.

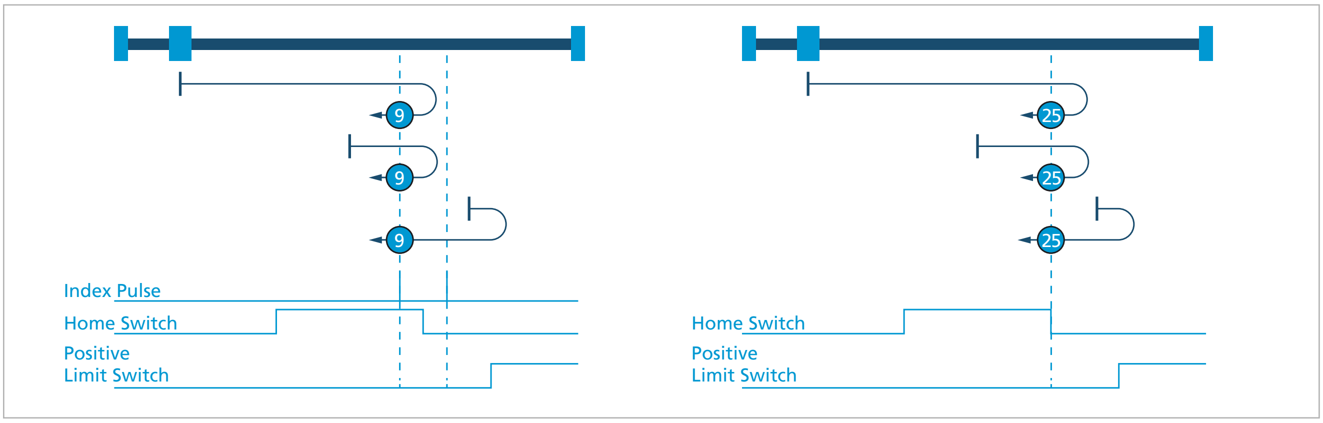


Fig. 40: Homing methods 9 and 25

- Methods 10 and 26:

  Homing at the falling flank at the top. Always starts in the positive direction.

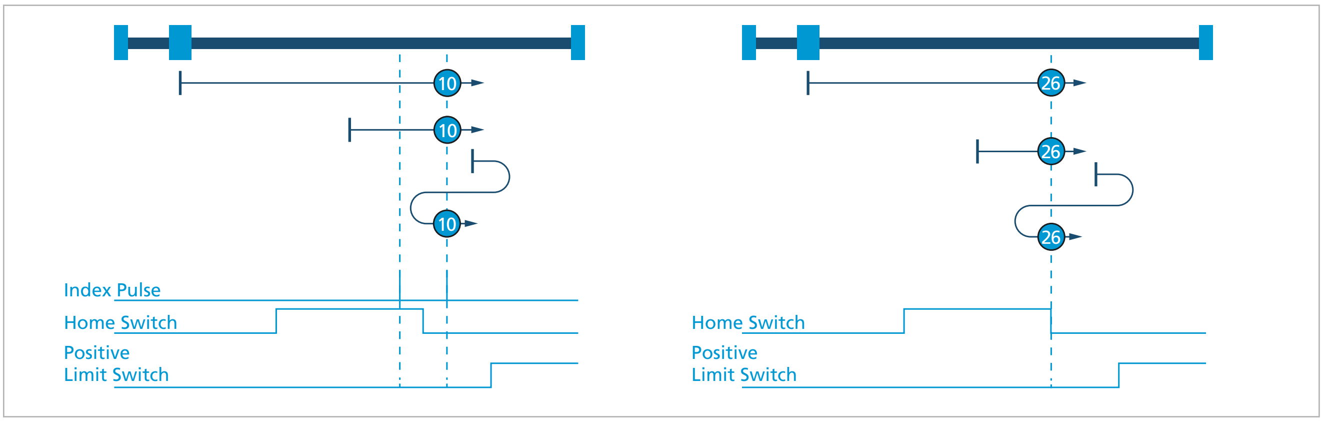


Fig. 41: Homing methods 10 and 26

- Methods 11 and 27:

  Homing at the falling flank at the top. Starts in a negative direction if the switch is inactive.

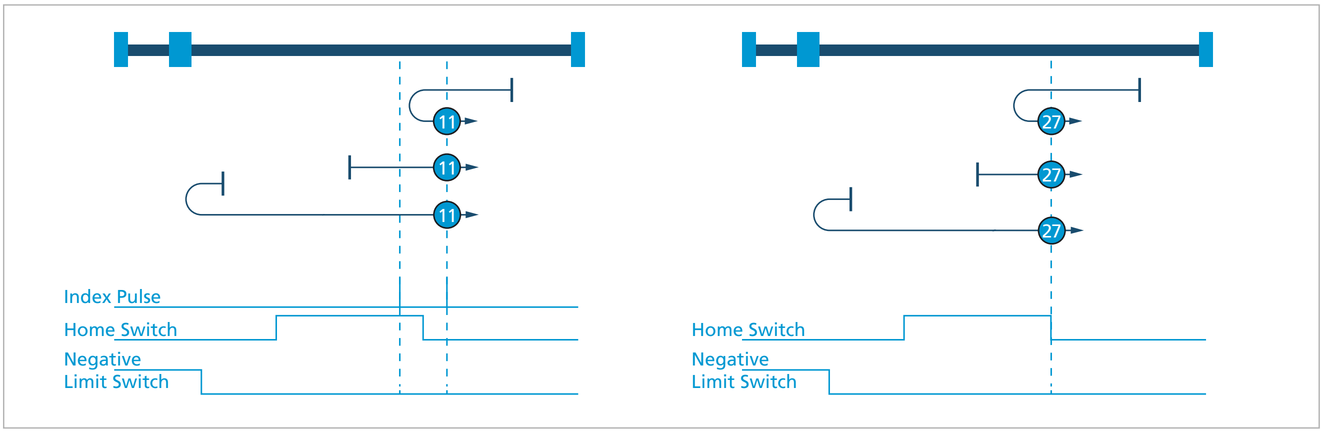


Fig. 42: Homing methods 11 and 27

- Methods 12 and 28:

  Homing at the rising flank at the top. Starts in a negative direction if the switch is inactive.

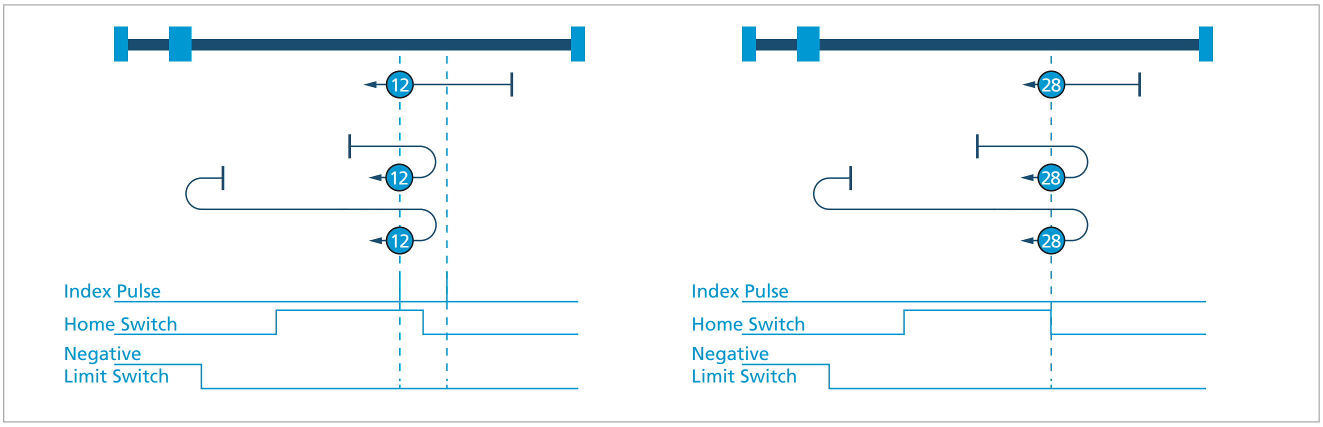


Fig. 43: Homing methods 12 and 28

- Methods 13 and 29:

  Homing at the rising flank at the bottom. Always starts in the negative direction.

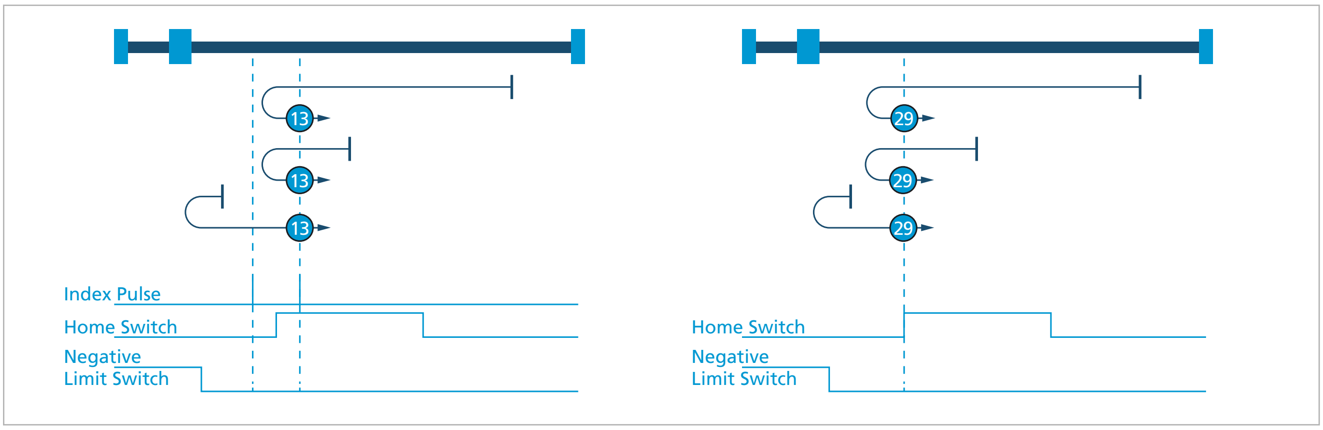


Fig. 44: Homing methods 13 and 29

- Methods 14 and 30:

  Homing at the falling flank at the bottom. Always starts in the negative direction.

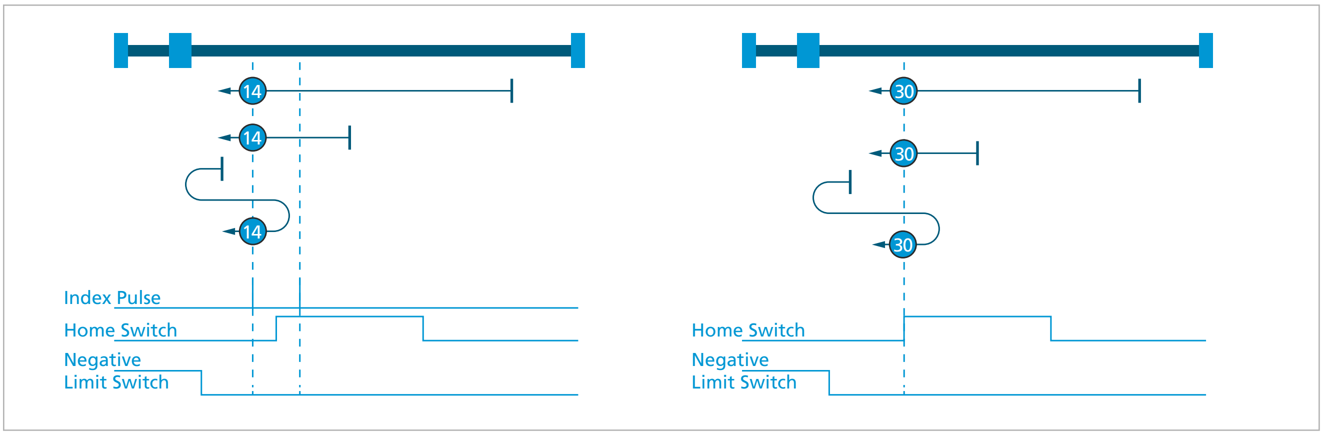


Fig. 45: Homing methods 14 and 30

Methods 33 and 34
Homing at the index impulse. The drive moves in the negative (33) or positive (34) direction to the index impulse.

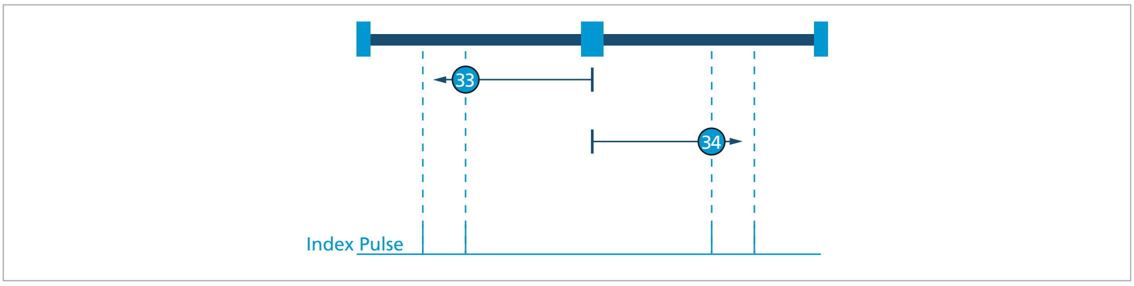


Fig. 46: Homing methods 33 and 34

Method 37
The position counter is zeroed at the current position.

Methods –1 and –3

Homing at the negative stop:

- The drive moves in the negative direction until a block is detected. In the variant –3, the position there is set to 0. In the variant –1, the drive reverses there and moves to the next index impulse. The position there is set to 0.

Methods –2 and –4

Homing at the positive stop:

- The drive moves in the positive direction until a block is detected. In the variant –4, the position there is set to 0. In the variant –2, the drive reverses there and moves to the next index impulse. The position there is set to 0.

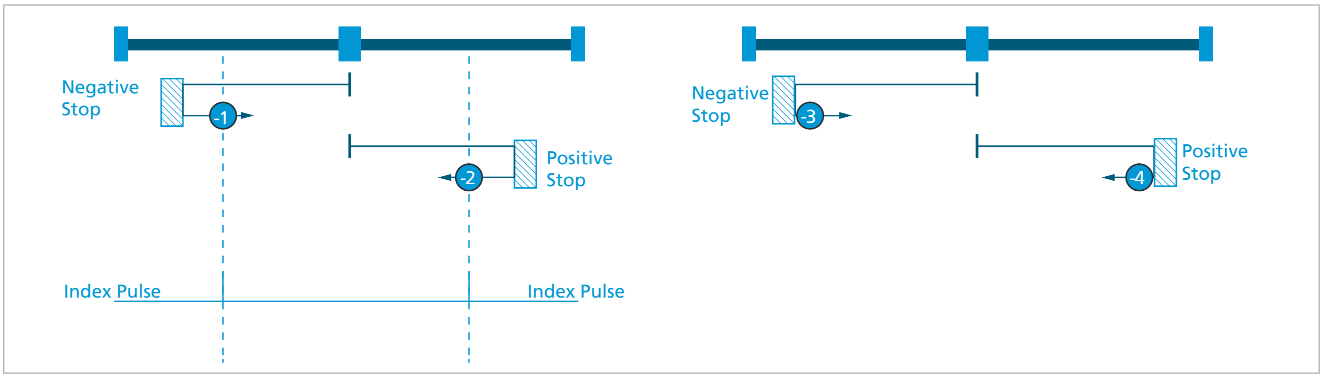


Fig. 47: Homing methods –1, –2, –3 and –4

5.4.2 Statusword/Controlword for Homing mode

In Homing operating mode, operating mode-specific bits are used in the controlword and statusword.

Tab. 43: Operating mode-specific bits of the controlword (Homing mode)

Bit	Function	Description
4	Homing operation start	0: Do not start homing 1: Start homing

Tab. 44: Operating mode-specific bits of the statusword (Homing mode)

Bit	Function	Description
10	Target Reached	0: Target not reached 1: Target reached
12	Homing attained	0: Homing procedure not completed 1: Homing procedure completed
13	Homing error	0: No fault found 1: Fault found

Tab. 45: Available bit combinations of the statusword and their meaning

Bit 13	Bit 12	Bit 10	Meaning
0	0	0	Homing procedure active
0	0	1	Homing procedure interrupted or not started
0	1	0	Homing procedure has been completed, the velocity is not yet 0
0	1	1	Homing procedure has been successfully completed
1	0	0	A homing error has occurred, velocity is not 0
1	0	1	A homing error has occurred, velocity is 0
1	1	X	Reserved

If analogue Hall signals are used, an index signal is generated internally once per pole pair.

If AES or SSI sensors are used, an index signal is generated internally once per revolution.

5.4.3 Settings

The following objects must be set when using this operating mode:

- Operating mode (0x6060 = 6)
- Homing Method (0x6098)
- Homing offset (0x607C)
- Homing speed (0x6099)
- Homing Acceleration (0x609A)

Homing Method

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6098	0x00	Homing Method	S8	rw	0	Homing method

Homing offset

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x607C	0x00	Min Range Limit	S32	rw	0	Offset of the zero position relative to the position of the reference switch

Homing speed

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6099	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Switch Seek Velocity	U32	rw	400	Switch seek velocity
	0x02	Homing Speed	U32	rw	400	Homing velocity

Homing Acceleration

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x609A	0x00	Homing Acceleration	U32	rw	50	Acceleration during homing

5.4.4 Example of a homing reference run

✓ Drive status *Operation Enabled*

✓ Modes of operation (object 0x6060) set to Homing mode

1. Assign the desired values to the following objects:
   - Homing limit switch (object 0x2310)
   - Homing Method (object 0x6098)
   - Homing speed (object 0x6099)
   - Homing Acceleration (object 0x609A)
2. Set bit 4 (Homing Operation Start) in the controlword to 1.

↳ The drive responds with 0 at bit 12 and bit 10 of the statusword.

↳ The drive will now start the reference run.

↳ When the homing position is reached and the reference run has been completed, bit 12 and bit 10 of the statusword will be set to 1.

i Before a further reference run can be performed, bit 4 in the controlword must be reset once again. This causes bit 12 in the statusword to be reset.

5.5 Cyclic Synchronous Position mode

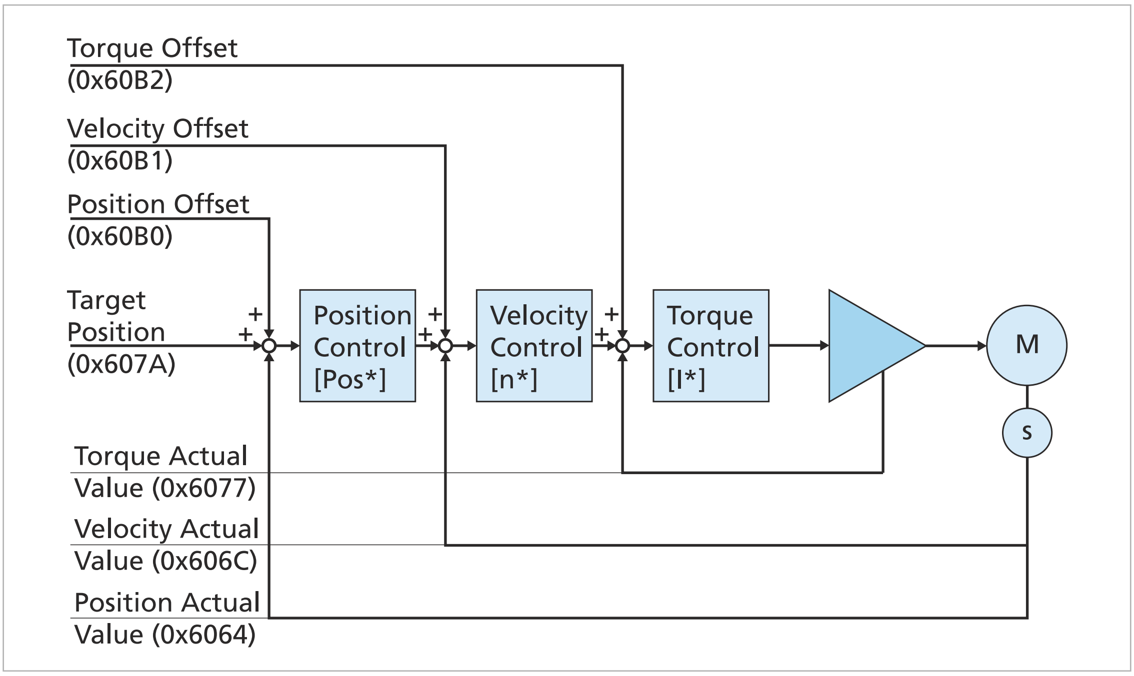


Fig. 48: Cyclic Synchronous Position mode control loop

In the Cyclic Synchronous Position operating mode the movement profile is performed in the master and not in the drive (slave). For this purpose the controller provides cyclical target positions for the drive. The drive then performs the torque control, velocity control and position control. Optionally, the master can provide additive velocities and torques to the velocity and torque pre-control.

The following actual values are provided by measurement from the drive to the controller:

- Position
- Velocity
- Torque

The Cyclic Synchronous Position mode contains the following sub-functions:

- Target value input
- Position recording
- Position control with limitation of the target position
- Velocity recording
- Velocity control with limitation of the target velocity
- Current measurement
- Current control with limitation of the current / torque / force setpoint value

5.5.1 Description of functions

From the point of view of the controller the following values are inputs:

- Target Position (object 0x607A)
- Offset position (object 0x60B0)
- Offset velocity (object 0x60B1)
- Offset torque (object 0x60B2)

The actual value of the position (object 0x6064) is the mandatory output of the controller. Further outputs can be:

- Torque Actual Value (object 0x6077)
- Velocity Actual Value (object 0x606C)
- Velocity Sensor Actual Value (object 0x6069)
- Following Error Actual Value (object 0x60F4)

The drive can have the following functions:

- Position limitation
- Velocity limitation
- Torque limitation
- Quick Stop

5.5.2 Statusword/Controlword for Cyclic Synchronous Position mode

In Cyclic Synchronous Position operating mode the controlword is not assigned to any operating mode-specific bits. Operating mode-specific bits are assigned to the statusword.

Tab. 46: Operating mode-specific bits of the statusword (Cyclic Synchronous Position mode)

Bit	Function	Description
10	Reserved	0: Reserved
12	The drive follows the command value	0: The drive does not follow the command value; the target position is ignored 1: The drive follows the operating value, the target position is used as an input to the position control
13	FollowingError	0: The actual position follows the instructions without a following error. 1: Permissible range for the following error exceeded.

5.5.3 Settings

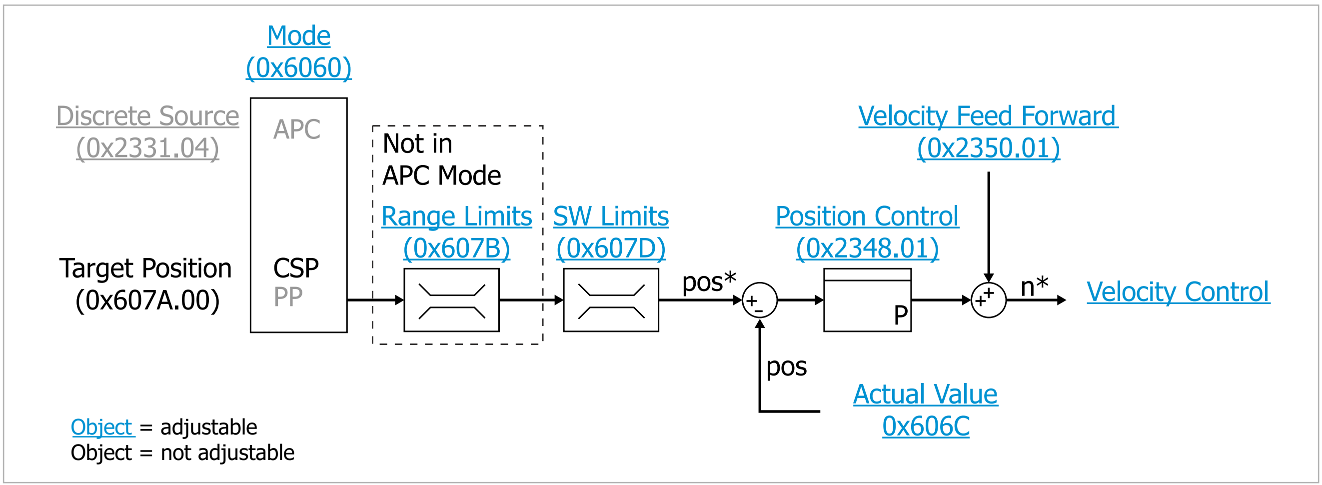


Fig. 49: Motion Manager view of the Cyclic Synchronous Position mode

The following objects must be set when using this operating mode:

- Target Position (0x607A)
- Operating mode (0x6060 = 8)

Fig. 50 shows all the objects that are effective in this operating mode. The objects shown additionally permit optional settings within this operating mode.

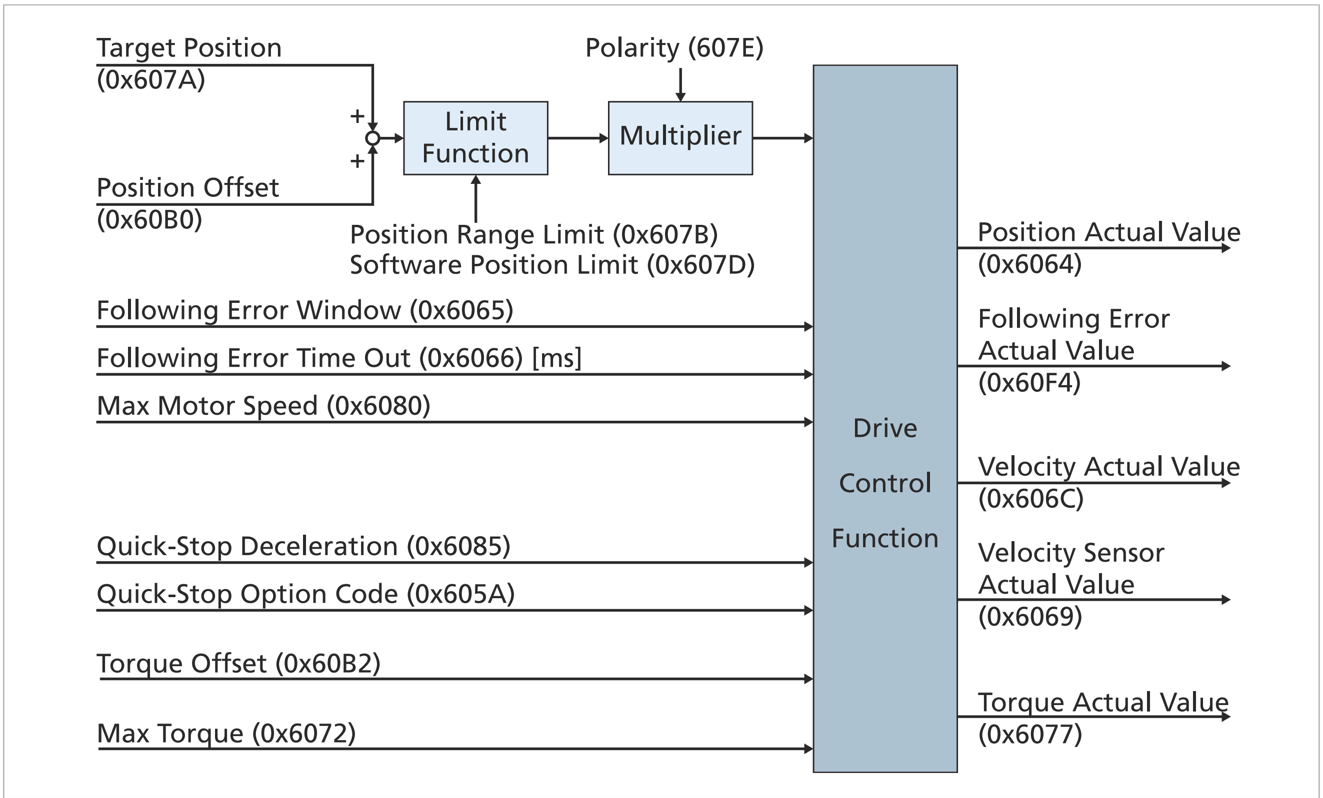


Fig. 50: Overview of all the objects that are effective in CSP operating mode

5.5.4 Example

✓ Motion Controller and drive are connected

✓ Settings have been performed according to chap. 4, p. 23

✓ Operating mode has been set (object 0x6060 = 8)

- Set the state machine of the drive to the *Operation Enabled* state.
- Enter the desired position into the target position object (0x607A).

↳ The drive will move to the set position.

5.6 Cyclic Synchronous Velocity mode

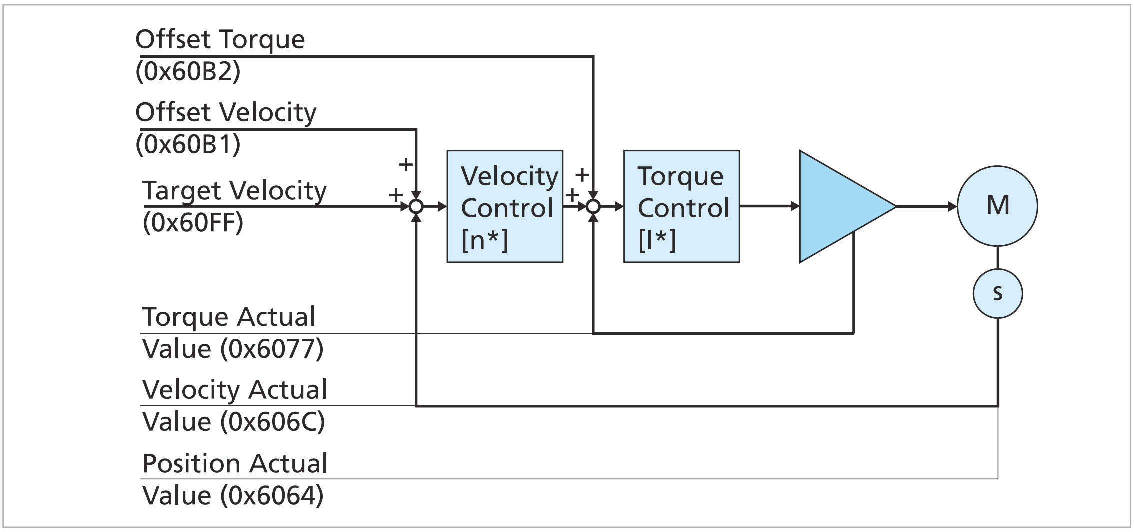


Fig. 51: Cyclic Synchronous Velocity mode control loop

In the Cyclic Synchronous Velocity operating mode, the movement profile is performed in the master and not in the drive (slave). In this case the controller provides cyclical target velocities of the drive. The drive then performs torque control and velocity control. If desired, the position controller can be connected to the communications system. In order to control the speed using two procedures, optionally an additive speed for pre-control can be provided to the master.

The following actual values are provided by measurement from the drive to the controller:

- Position
- Velocity
- Torque

Cyclic Synchronous Velocity mode contains the following sub-functions:

- Target value input
- Velocity recording
- Velocity control with limitation of the target velocity
- Current measurement
- Current control with limitation of the current / torque / force setpoint value

5.6.1 Description of functions

From the point of view of the velocity controller the following values are inputs:

- Target Velocity (object 0x60FF)
- Offset torque (object 0x60B2)

The actual value of the velocity (object 0x606C) is the mandatory output of the controller. Further outputs can be:

- Torque Actual Value (object 0x6077)
- Velocity Sensor Actual Value (object 0x6069)

The drive can have the following functions:

- Velocity limitation
- Torque limitation
- Quick Stop

5.6.2 Statusword/Controlword for Cyclic Synchronous Velocity mode

In Cyclic Synchronous Velocity operating mode the controlword is not assigned to any operating mode-specific bits. Operating mode-specific bits are assigned to the statusword.

Tab. 47: Operating mode-specific bits of the statusword (Cyclic Synchronous Velocity mode)

Bit	Function	Description
10	Reserved	0: Reserved
12	The drive follows the command value	0: The drive does not follow the command value; the target velocity is ignored 1: The drive follows the operating value, the target velocity is used as an input to the speed control
13	Reserved	0: Reserved

5.6.3 Settings

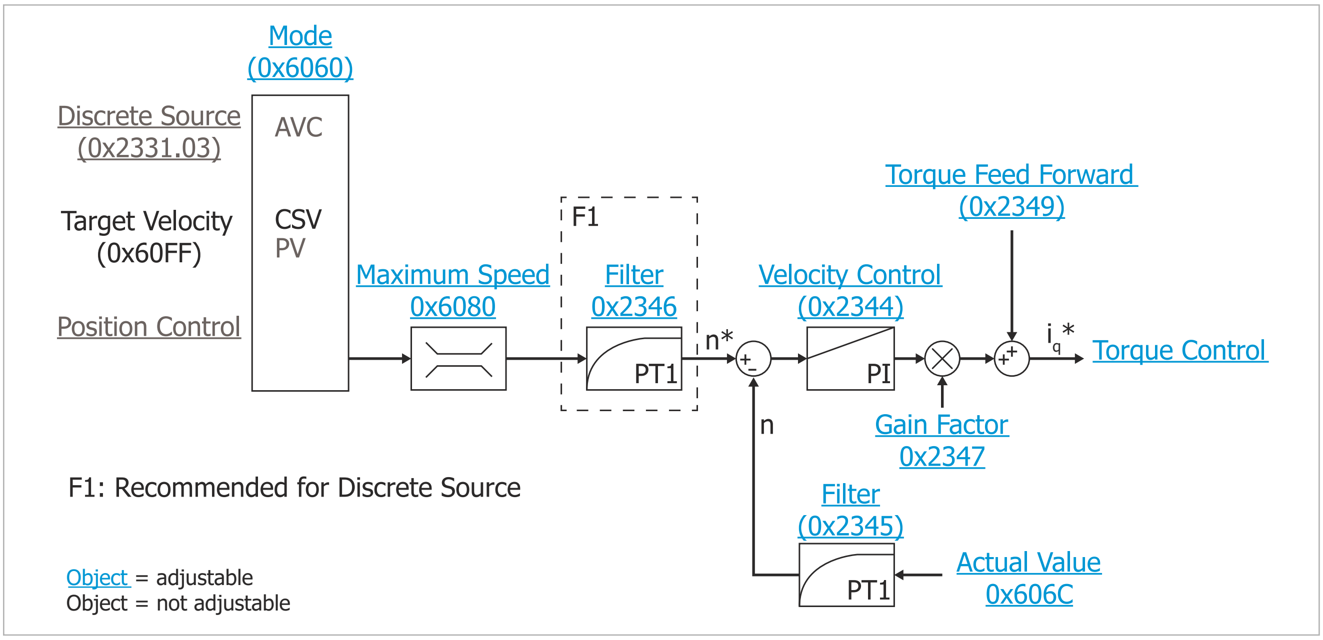


Fig. 52: Motion Manager view of the Cyclic Synchronous Velocity mode

The following objects must be set when using this operating mode:

- Target Velocity (0x60FF)
- Operating mode (0x6060 = 9)

Fig. 53 shows all the objects that are effective in this operating mode. The objects shown additionally permit optional settings within this operating mode.

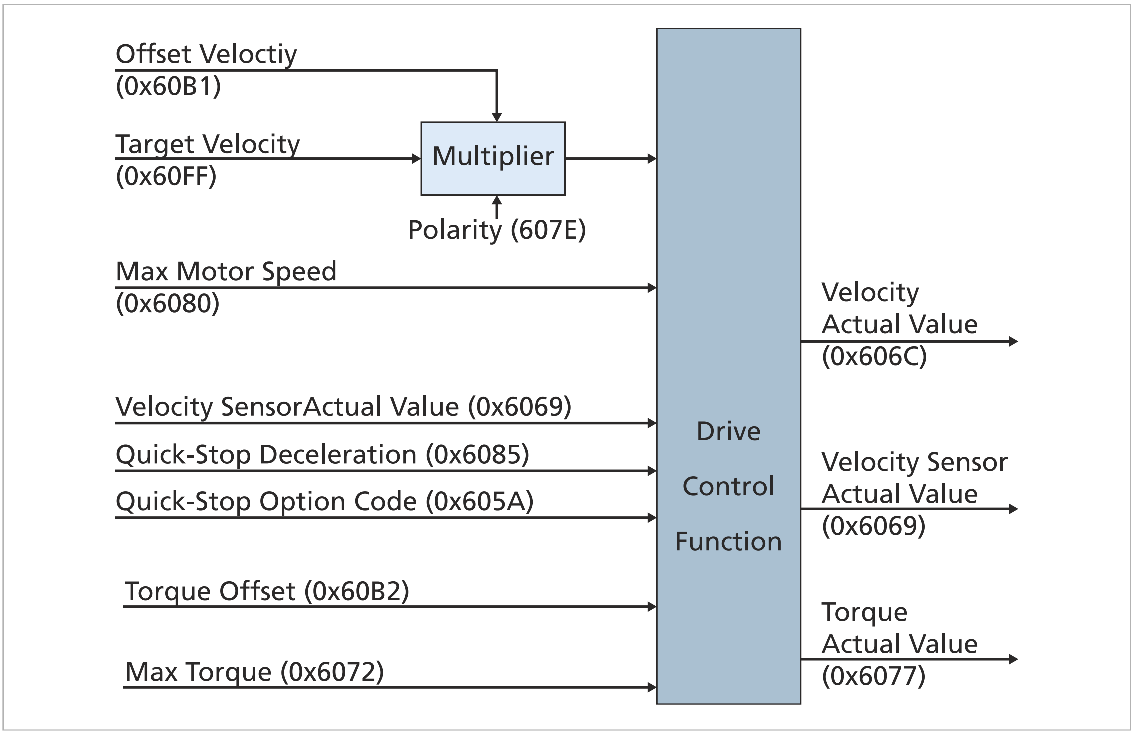


Fig. 53: Overview of all the objects that are effective in CSV operating mode

5.6.4 Example

✓ Motion Controller and drive are connected

✓ Settings have been performed according to chap. 4, p. 23

✓ Operating mode has been set (object 0x6060 = 9)

- Set the state machine of the drive to the *Operation Enabled* state.
- Enter the desired velocity in the target velocity object (0x60FF).

↳ The drive will move at the set velocity.

5.7 Cyclic Synchronous Torque mode

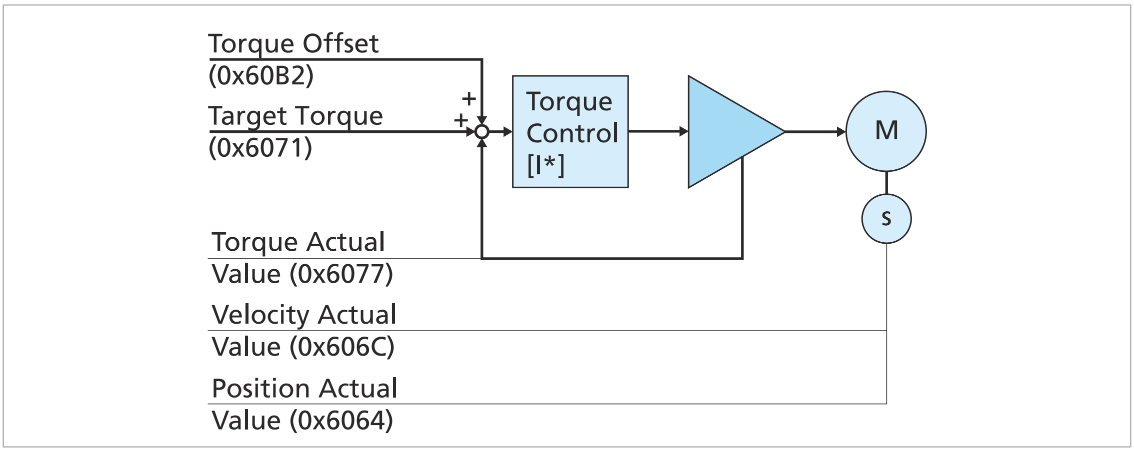


Fig. 54: Cyclic Synchronous Torque mode control loop

In the Cyclic Synchronous Torque operating mode the movement profile is performed in the controller and not in the drive. In this case the controller provides cyclical target torques of the drive. The drive then performs torque control. If desired, the position and velocity controller can be connected to the communications system.

The following actual values are provided by measurement from the drive to the controller:

- Position
- Velocity
- Torque

Cyclic Synchronous Torque mode contains the following sub-functions:

- Target value input
- Current measurement
- Current control with limitation of the current / torque / force setpoint value

5.7.1 Description of functions

From the point of view of the torque controller the following values are inputs:

- Target torque (object 0x6071)

The actual value of the torques (object 0x6077) is the mandatory output of the controller.

The drive can have the following functions:

- Torque limitation
- Quick Stop

5.7.2 Statusword/Controlword for CST

In Cyclic Synchronous Torque operating mode the controlword is not assigned to any operating mode-specific bits. Operating mode-specific bits are assigned to the statusword.

Tab. 48: Operating mode-specific bits of the statusword (Cyclic Synchronous Torque mode)

Bit	Function	Description
10	Reserved	0: Reserved
12	The drive follows the command value	0: The drive does not follow the command value; the target torque (object 0x6071) is ignored 1: The drive follows the operating value, the target torque (object 0x6071) is used as an input to the torque control
13	Reserved	0: Reserved

5.7.3 Settings

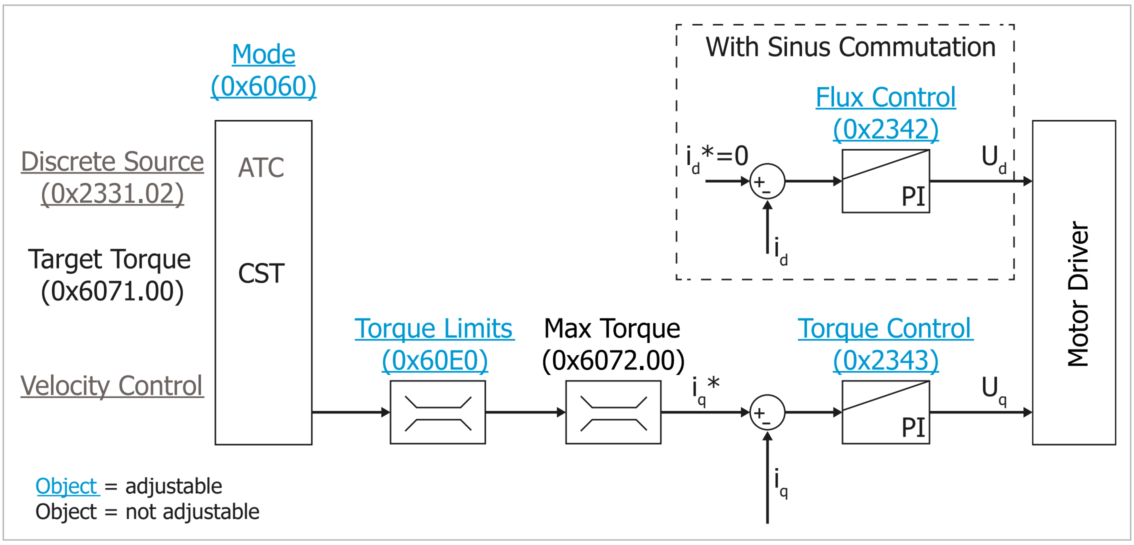


Fig. 55: Motion Manager view of the Cyclic Synchronous Torque mode

The following objects must be set when using this operating mode:

- Operation mode (0x6060)
- Target Torque (0x6071)

Fig. 56 shows the major objects that are effective in the operating mode.

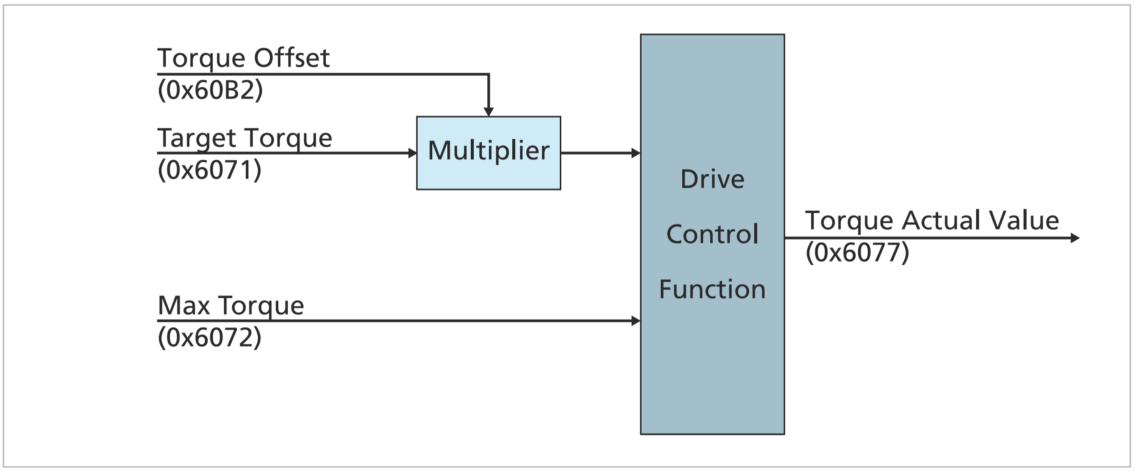


Fig. 56: Overview of all the objects that are effective in CST operating mode

5.7.4 Example

- ✓ Motion Controller and drive are connected
- ✓ Settings have been performed according to chap. 4, p. 23
- ✓ Operating mode has been set (object 0x6060 = 10)
- ▸ Set the state machine of the drive to the *Operation Enabled* state.
- ▸ Enter the desired torque in the target torque object (0x6071).
- ↳ The drive will move with the set torque.

i If the load torque is lower than the required torque, the drive will accelerate to the maximum speed.

5.8 Voltage Mode

In the Voltage mode a motor voltage is output proportional to the specified value. Current limitation still remains active. A supervisory controller can be used in Voltage mode. The controller then acts as a power amplifier. The voltage specification can be performed either via the object 0x2341 or via a discrete input such as an analogue value.

The following actual values are provided by measurement from the drive to the controller:

- Position
- Velocity
- Torque

Voltage mode contains the following sub-functions:

- Records the actual values for current, velocity and position.
- Current limitation by limitation of the output voltage.

5.8.1 Description of functions

From the point of view of the controller the following values are inputs:

- Voltage Mode Reference (Object 0x2341)

or

- Discrete Source for Ref-Voltage (object 0x2331.01, see chap. 4.8.2, p. 78)

Voltage mode has no mandatory outputs.

The drive can have the following functions:

- Current limitation
- Quick Stop

5.8.2 Statusword/Controlword for Voltage mode

In Voltage mode there are no operating mode-specific bits in the controlword or status-word.

5.8.3 Settings

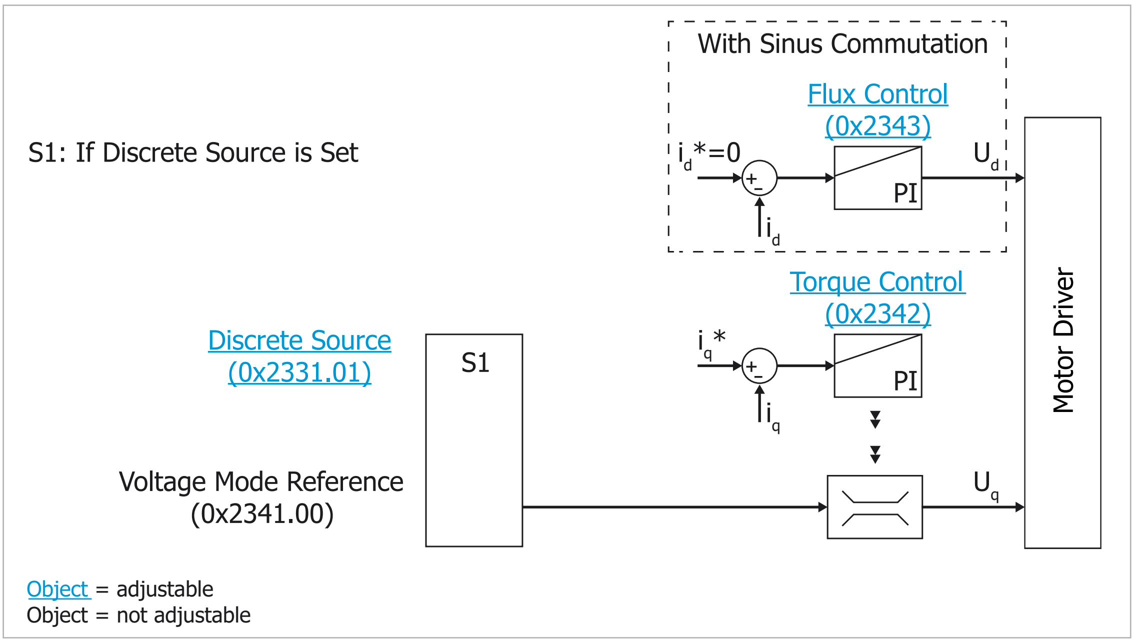


Fig. 57: Motion Manager's View of the Voltage mode

The following objects must be set when using this operating mode:

- Operating mode (0x6060 = –1)
- Voltage Mode Reference (Object 0x2341)

or

- Discrete Source for Ref-Voltage (object 0x2331.01)

Voltage Mode Reference

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2341	0x00	Voltage Mode Reference	S16	rw	0	Voltage setpoint of the Voltage mode [10 mV / digit]

Discrete Source for Ref-Voltage

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2331	0x00	Number of entries	U8	ro	4	Number of object entries
	0x01	Target Voltage Source	U8	rw	0	Selection of the discrete source for the voltage setpoint value
	0x02	Target Current Source	U8	rw	0	Selection of the discrete source for the torque setpoint value
	0x03	Target Velocity Source	U8	rw	0	Selection of the discrete source for the speed setpoint value
	0x04	Target Position Source	U8	rw	0	Selection of the discrete source for the position setpoint value

If no discrete source for the setpoint value is selected via the object 0x2331.01, the value in object 0x2341 is used as the setpoint value.

The available settings for the discrete setpoint value sources and their meanings can be found in chap. 4.8.2, p. 78.

5.8.4 Example

✓ Motion Controller and drive are connected

✓ Settings have been performed according to chap. 4, p. 23

- Set the operating mode (0x6060 = –1).
- Configure the setpoint value via the communications system in the object 0x2341 or via a discrete voltage specification in the object 0x2331.01.
  - If specifying the setpoint value via the communications system, set the setpoint value source value to **0** in the object 0x2331.01.
  - If specifying the setpoint value by discrete voltage specification, set the setpoint value as described in chap. 4.8.2, p. 78.
- Set the state machine of the drive to the *Operation Enabled* state.
- If specifying the setpoint value via the communications system, set the setpoint value via a write access to the object 0x2341.

↳ The motor will move with the specified voltage. Current, velocity and position are uncontrolled and they vary as a result of voltage and load.

5.9 Analogue Position Control mode

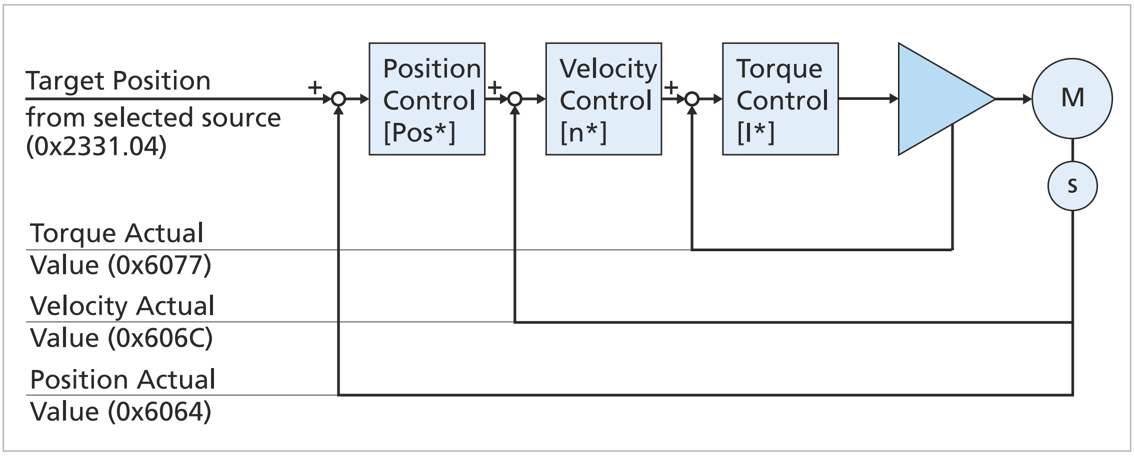


Fig. 58: Analogue Position mode control loop

In the Analogue Position Control (APC) operating mode the Motion Controller controls the position of the connected motor. The position setpoint value is specified via a discrete input. The following are supported:

- Analogue input (AnIn1/AnIn2)
- PWM input
- Position specification via the reference encoder (quadrature signal)
- Position specification via a pulsed/directional signal

The following diagnostic options are available via the communications system:

- Position
- Velocity
- Torque

Analogue Velocity Control mode contains the following sub-functions:

- Analogue signal provision (see chap. 4.6.4, p. 58 or chap. 4.8.2, p. 78)
- Target value input
- Position recording
- Position control with limitation of the target position
- Velocity recording
- Velocity control with limitation of the target velocity
- Current measurement
- Current control with limitation of the current / torque / force setpoint value

5.9.1 Description of functions

From the point of view of the position controller the following values are inputs:

- Discrete Source for Reference Position (object 0x2331.04, see chap. 4.8.2, p. 78)

The drive can have the following functions:

- Velocity limitation
- Torque limitation
- Quick Stop

5.9.2 Controlword/Statusword for Analogue Position Control mode

In Analog Position Control Mode there are no operating mode-specific bits in the control-word or statusword.

5.9.3 Settings

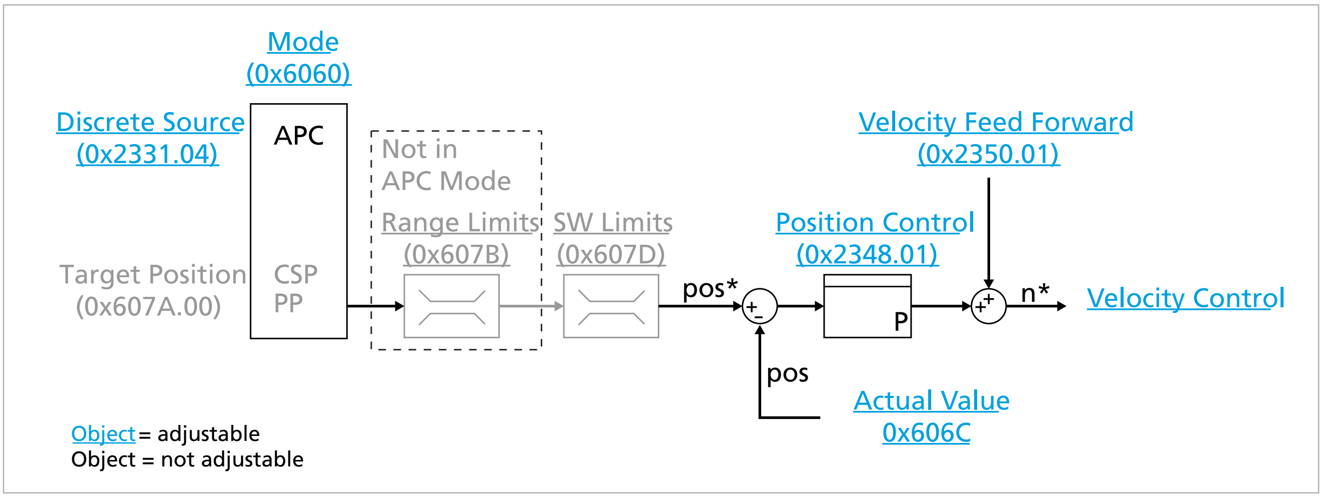


Fig. 59: Motion Manager view of the Analogue Position Control mode

The following objects must be set when using this operating mode:

- Operating mode (0x6060 = –2)
- Discrete Source for Reference Position (0x2331.04)

Fig. 60 shows all the objects that are effective in this operating mode. The objects shown additionally permit optional settings within this operating mode.

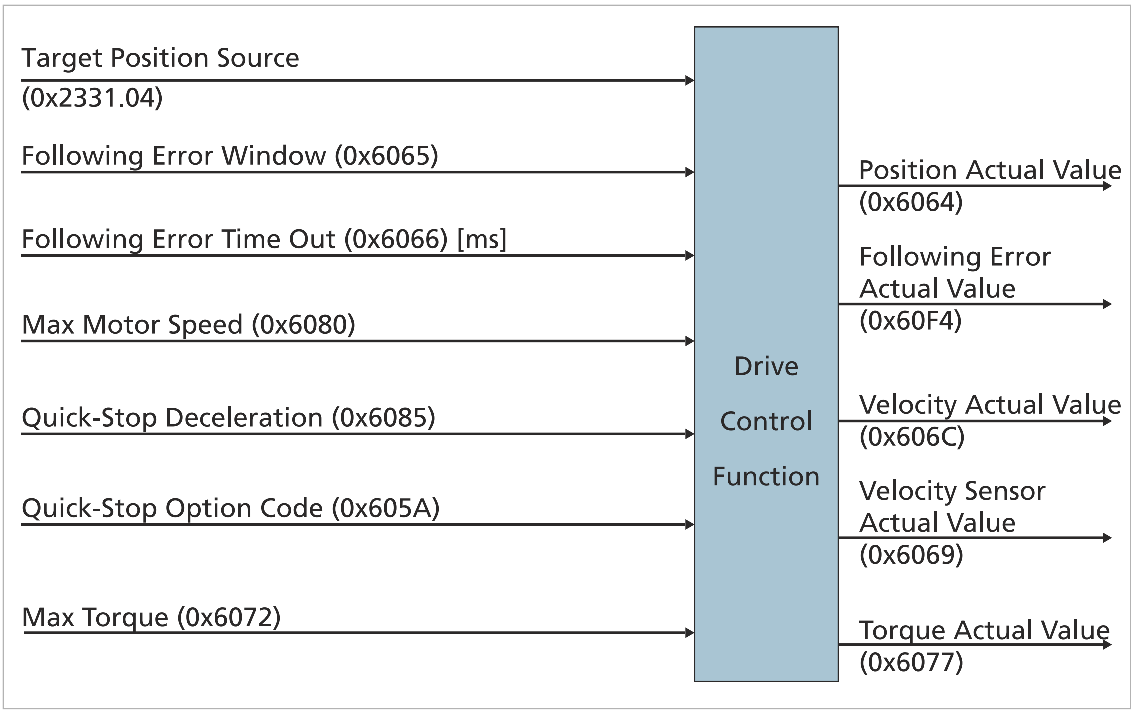


Fig. 60: Overview of all the objects that are effective in APC operating mode

5.9.4 Example

- ✓ Motion Controller and drive are connected
- ✓ Settings have been performed according to chap. 4, p. 23
- ✓ Operating mode has been set (object 0x6060 = –2)
- Enter the desired setpoint value source in the Discrete Source for Reference Position object (0x2331.04) (see chap. 4.8.2, p. 78)
- Set the state machine of the drive to the *Operation Enabled* state.
- ↳ The Motion Controller controls the position of the drive according to the specification.

5.10 Analogue Velocity Control mode

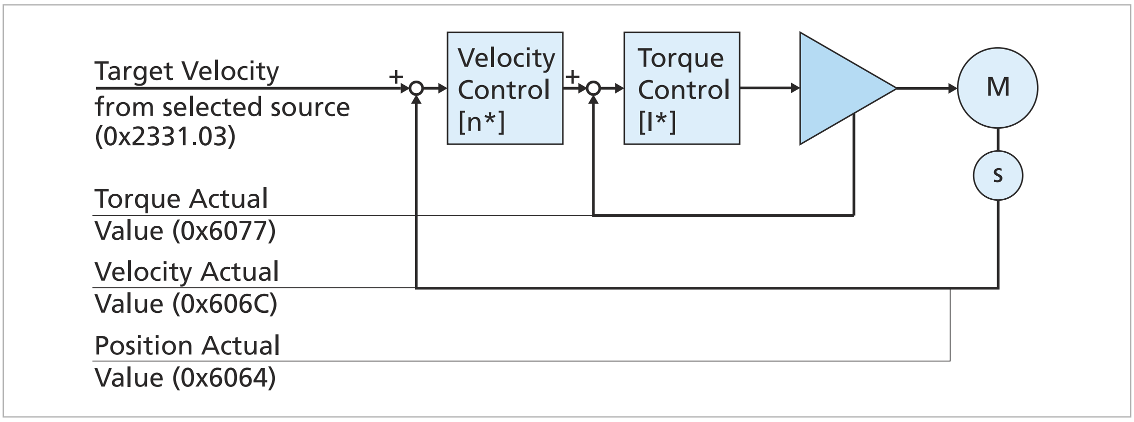


Fig. 61: Analogue Velocity Control mode control loop

In the Analogue Velocity Control (AVC) operating mode the Motion Controller controls the velocity of the connected motor. The target velocity is specified via a discrete input. The following are supported:

- Analogue input (AnIn1/AnIn2)
- PWM input
- Velocity specification via a quadrature signal at the reference encoder connection

The following diagnostic options are available via the communications system:

- Position
- Velocity
- Torque

Analogue Velocity Control mode contains the following sub-functions:

- Analogue signal provision (see chap. 4.6.4, p. 58 or chap. 4.8.2, p. 78)
- Target value input
- Velocity recording
- Velocity control with limitation of the target velocity
- Current measurement
- Current control with limitation of the current / torque / force setpoint value

5.10.1 Description of functions

From the point of view of the velocity controller the following values are inputs:

- Discrete Source for Reference Speed (object 0x2331.03, see chap. 4.8.2, p. 78)

The drive can have the following functions:

- Velocity controller
- Torque limitation
- Quick Stop

5.10.2 Controlword/Statusword for Analogue Velocity Control mode

In Analog Velocity Control Mode there are no operating mode-specific bits in the controlword or statusword.

5.10.3 Settings

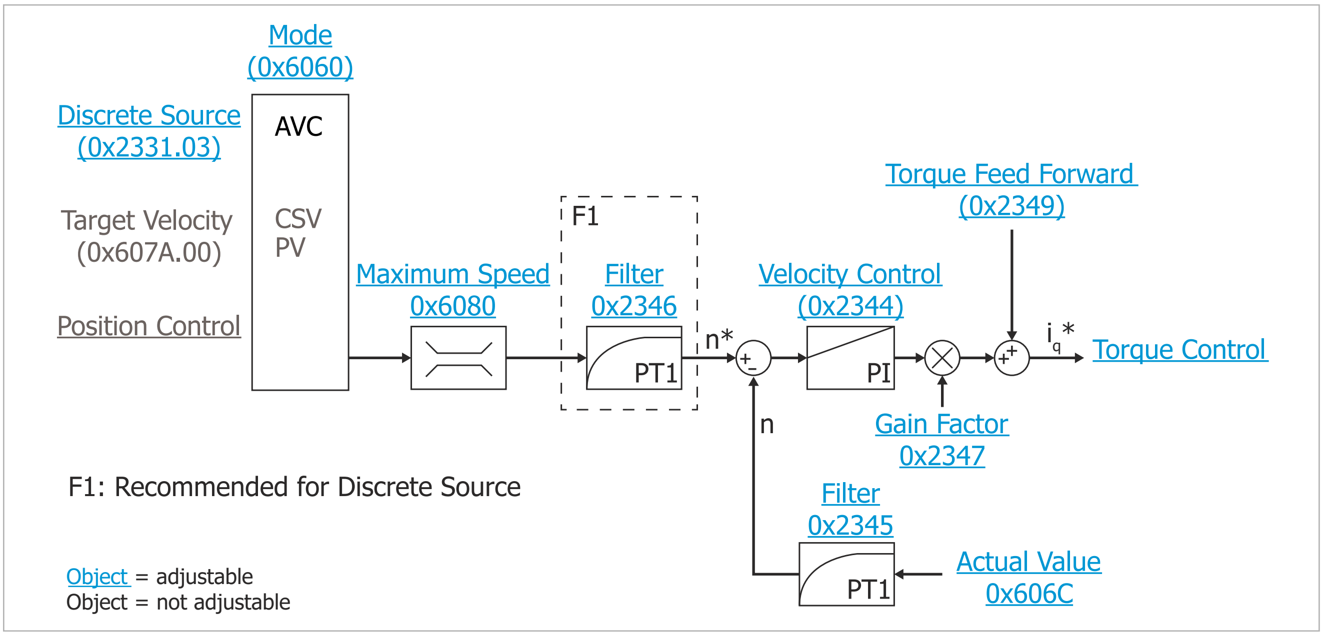


Fig. 62: Motion Manager view of the Analogue Velocity Control mode

The following objects must be set when using this operating mode:

- Operating mode (0x6060 = –3)
- Discrete Source for Reference Speed (0x2331.03)

Fig. 63 shows all the essential features that are effective in the operating mode.

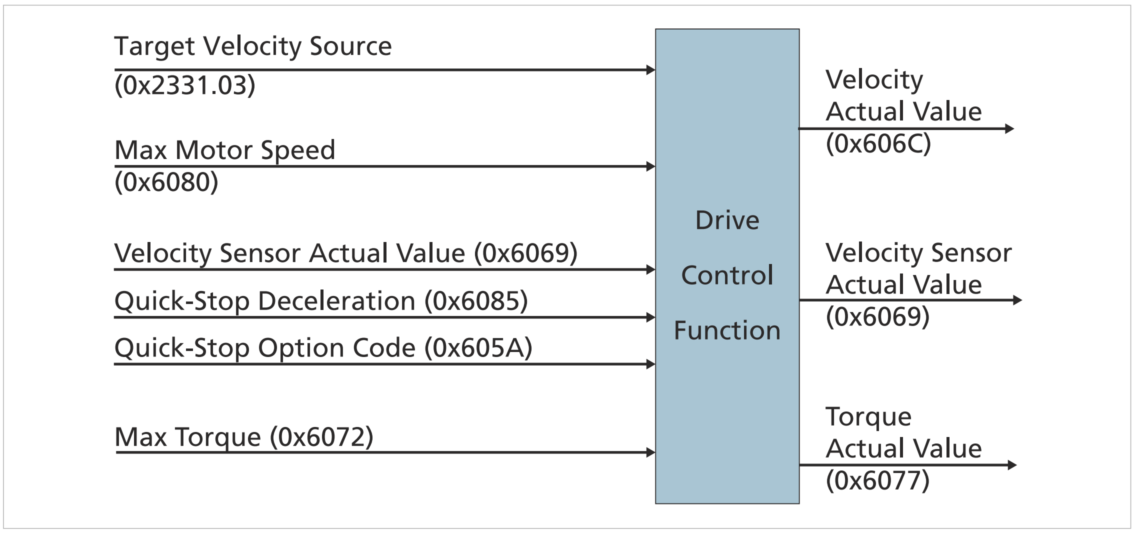


Fig. 63: Overview of all the objects that are effective in AVC operating mode

5.10.4 Example

- ✓ Motion Controller and drive are connected
- ✓ Settings have been performed according to chap. 4, p. 23
- ✓ Operating mode has been set (object 0x6060 = –3)
- ▶ Enter the desired setpoint value source in the Discrete Source for Reference Speed object (0x2331.03) (see chap. 4.8.2, p. 78)
- ▶ Set the state machine of the drive to the *Operation Enabled* state.
- ↳ The drive will move at the set velocity.

5.11 Analogue Torque Control mode

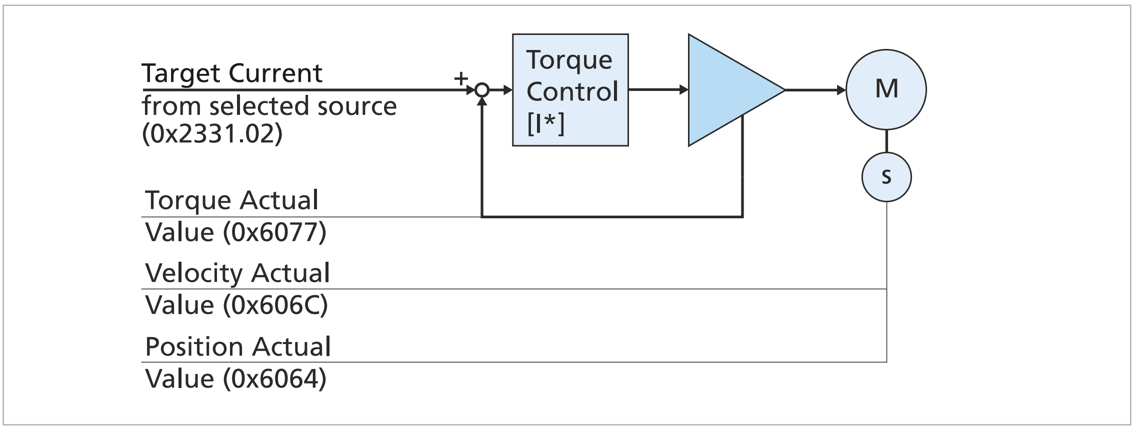


Fig. 64: Analogue Torque Control mode control loop

In the Analogue Torque Control (ATC) operating mode the Motion Controller controls the torque or the force of the connected motor via the motor current. The target torque or force is specified via a discrete input. The following are supported:

- PWM input
- Analogue input AnIn1 or AnIn2

The following diagnostic options are available via the communications system:

- Position
- Velocity
- Torque

Analogue Torque Control mode contains the following sub-functions:

- Analogue signal provision (see chap. 4.6.4, p. 58 or chap. 4.8.2, p. 78)
- Target value input
- Current measurement
- Current control with limitation of the current / torque / force setpoint value

5.11.1 Description of functions

From the point of view of the torque controller the following values are inputs:

- Discrete Source for Reference Current (object 0x2331.02, see chap. 4.8.2, p. 78)

The torque and force are controlled in the Motion Controller by the related motor current. A setpoint value of 1000 corresponds to the rated current of the motor. The input used as the setpoint value must be appropriately scaled for the purpose.

The drive can have the following functions:

- Torque limitation
- Quick Stop

5.11.2 Controlword/Statusword for Analogue Torque Control mode

In Analogue Torque Control mode there are no operating mode-specific bits in the controlword or statusword.

5.11.3 Settings

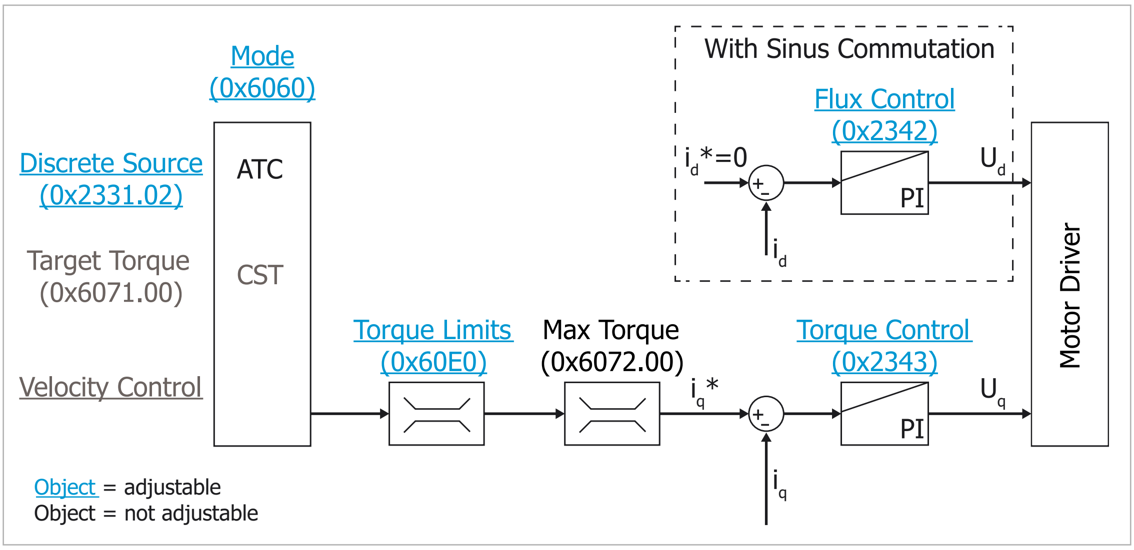


Fig. 65: Motion Manager view of the Analogue Torque Control mode

The following objects must be set when using this operating mode:

- Operating mode (0x6060 = –4)
- Discrete Source for Reference Current (0x2331.02)

Fig. 66 shows all the essential features that are effective in the operating mode.

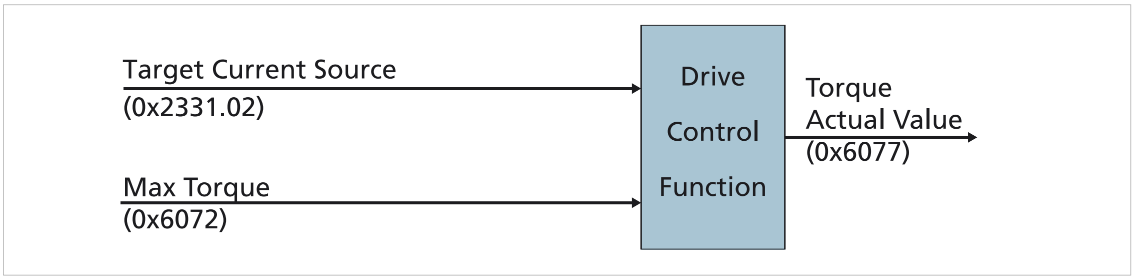


Fig. 66: Overview of all the objects that are effective in ATC operating mode

5.11.4 Example

✓ Motion Controller and drive are connected

✓ Settings have been performed according to chap. 4, p. 23

✓ Operating mode has been set (object 0x6060 = –4)

- Enter the desired setpoint value source in the discrete source for the reference current object (0x2331.02) (see chap. 4.8.2, p. 78).
- Set the state machine of the drive to the *Operation Enabled* state.

↳ The drive will move at the set torque or the set force.

6 Protection and monitoring devices

FAULHABER Motion Controllers have a range of protective safeguards for the output stage and the connected motor:

- Thermal models for the connected motor and output stage
- Current and torque limitation via the current controller
- Overvoltage control in braking mode
- Undervoltage monitoring

6.1 Temperature rise

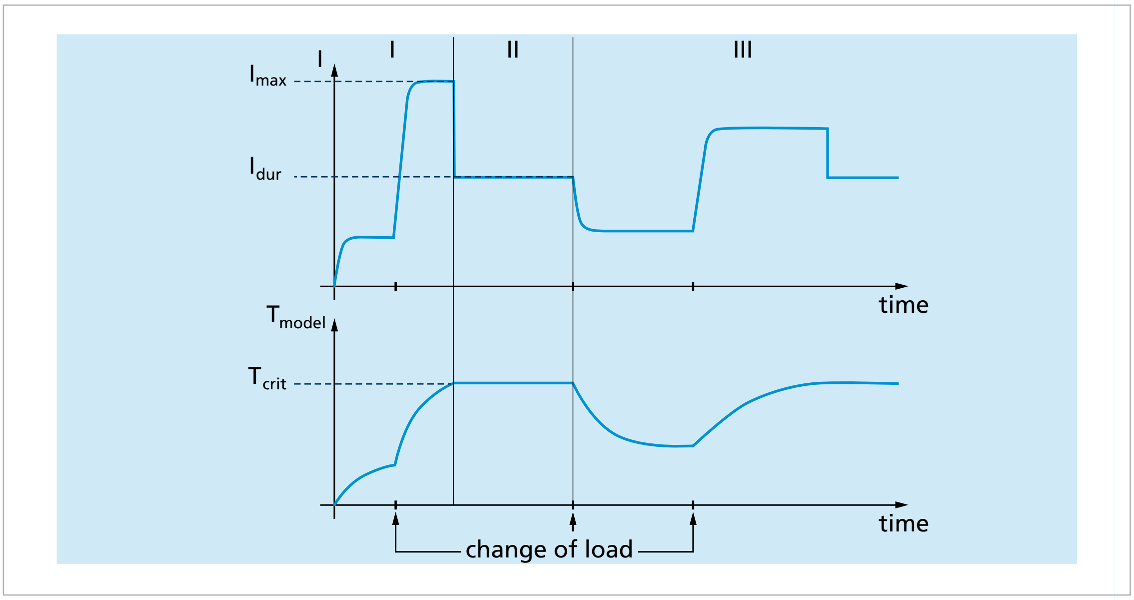


Fig. 67: Functionality of the I^2t current limitation

When the motor is cold, the current setpoint values are initially set to the applicable peak current value (range I).

A thermal current model, which calculates a model temperature from the current actually flowing, runs in parallel with this. If this model temperature exceeds a certain critical value, the drive switches to the continuous current and the motor current is limited to this (range II).

Only when the load is so low that the critical model temperature is no longer exceeded, the peak current is once again enabled as the limit. If the model temperature exceeds the critical value once again, the drive reverts to the continues current as the limit (range III)

At room temperature the peak current in S2 mode is typically available for several 100ms.

The purpose of this I²t current limitation is to select a suitable value for the continuous current so as to avoid the motor becoming overheated beyond the thermally permissible temperature. On the other hand, it should be possible to accept a high load for a brief period in order to have the capability to achieve very dynamic response movements.

CAUTION!
Depending on the installation situation and the ambient temperature, the motor winding temperature calculated by the Motion Controller can vary from the actual temperature.

- If necessary, reduce the permissible continuous current of the motor below the value stated on the datasheet.
- Touch the motor only when wearing protective clothing.

NOTICE!
If the continuous current of the motor is not configured to match the installation situation, the motor can overheat and the windings be damaged.

- Configure the data for the thermally permissible continuous current and the reduction of the thermal resistance in accordance with the installation situation.

6.2 Force or torque limitation

The current controller of the Motion Controller allows the torque output by a motor or the force output by a linear motor to be limited. Thus gearbox input stages can be protected against overloading.

To limit the current, the setpoint value of the current controller can be limited in a positive or negative direction via the objects 0x60E0 and 0x60E1.

6.3 Undervoltage monitoring

If the electronics power supply voltage of the Motion Controller falls below the lower limit set in the object 0x2325.01, an error is signalled in bit 19 of the device statusword (0x2324.01) and the output stage is immediately switched off. There is no automatic restart.

The lower threshold for the motor power supply is stated in object 0x2325.02. If the voltage at the connection for the motor power supply falls below this value for longer than the waiting time stated in object 0x2325.05, an undervoltage error for the motor power supply is signalled in bit 20 of the device statusword (0x2324.01). As for an undervoltage at the electronics power supply, the output stage is immediately switched off.

If the object 0x2325.03 lists 0 V as the undervoltage limit the motor power supply is not monitored for undervoltage.

6.4 Overvoltage controller

If the drive is in braking mode or generator mode, energy is fed back into the electrical mains. Usually the power supply units are not in a position to accept this energy. For this reason, in such circumstances the supply voltage can rise.

To avoid damage to components, FAULHABER Motion Controllers for brushless motors are equipped with a controller which adjusts the rotor displacement angle when the limit voltage set in the object 0x2325.04 is exceeded. In addition the braking power for all types of motor is reduced as necessary.

In order to be able to use the full dynamic response of the motor even in braking mode, when the load to be moved is large, a brake chopper should be connected to the DC power supply in parallel with the Motion Controller.

7 Diagnostics

7.1 Device monitoring

The diagnostic component of the FAULHABER Motion Controller monitors the device status on a cyclical basis.

The monitoring includes:

- Power supplies
- Temperatures
- Dynamic drive status

The results of the checks are stored as bits in the device statusword 0x2324.01. In addition, signals such as the state of the limit switch can also be evaluated centrally by means of the device statusword.

The device statusword can be queried by the communications interface. In addition selected statuses can also be signalled by a selectable digital output.

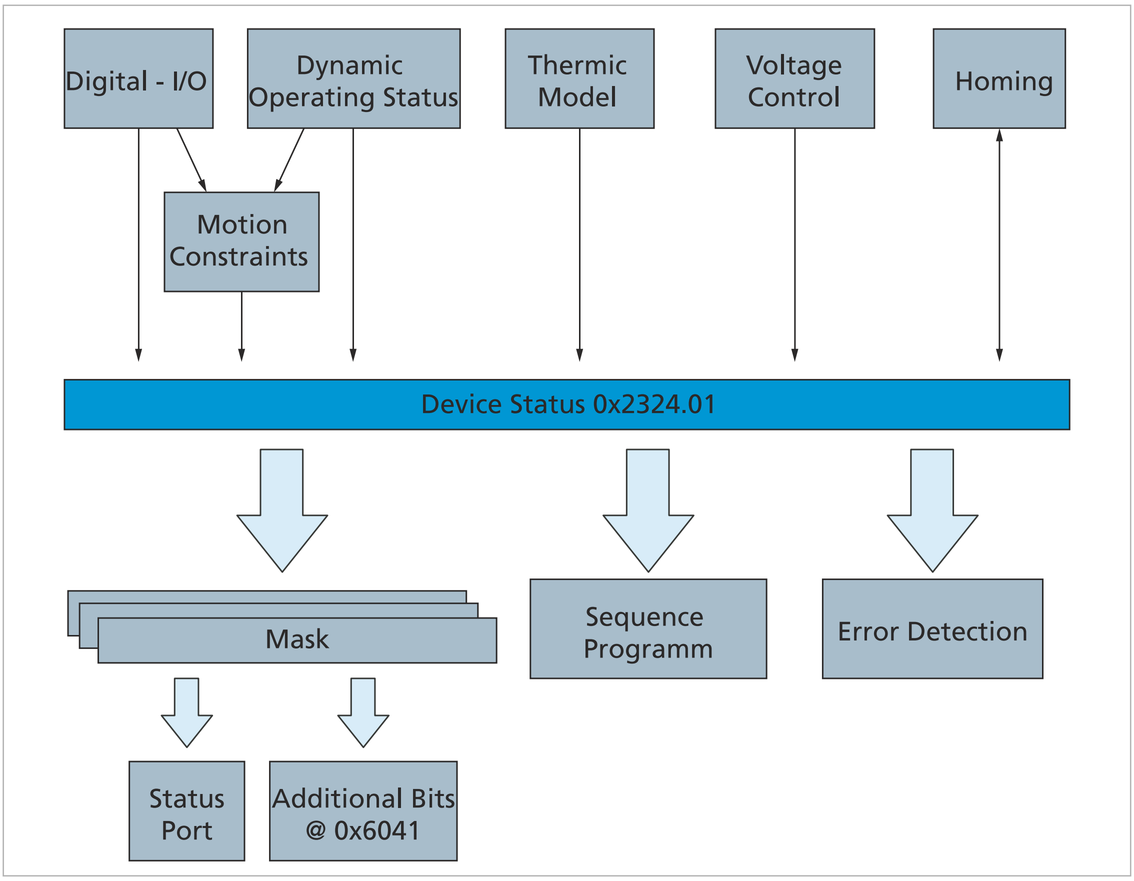


Fig. 68: Diagram of the device statusword

The statuses of the first 8 digital inputs of the Motion Controller are displayed in the upper 8 bits of the device statusword. This allows the sequence program to respond easily to a change of state of one of the inputs.

7.1.1 Checking the dynamic state of the drive

In the PP and PV operating modes, the position window and the velocity window are monitored to determine whether the specified setpoint values have been reached.

In addition the drive continuously monitors whether the drive is in motion or is stationary (n=0 monitoring). In PV operating mode this information is also displayed in the statusword 0x6041.

A check is made for the output voltage, the current control and velocity control, to monitor whether the output value or the setpoint values are currently limited.

A limited setpoint value for the velocity control or current control, in conjunction with multiple axes, can lead to a failure to achieve the desired value for the position or velocity within the expected period of time.

7.1.2 Checking the power supplies

The electronics power supply and motor power supply of the FAULHABER Motion Controller are monitored continuously.

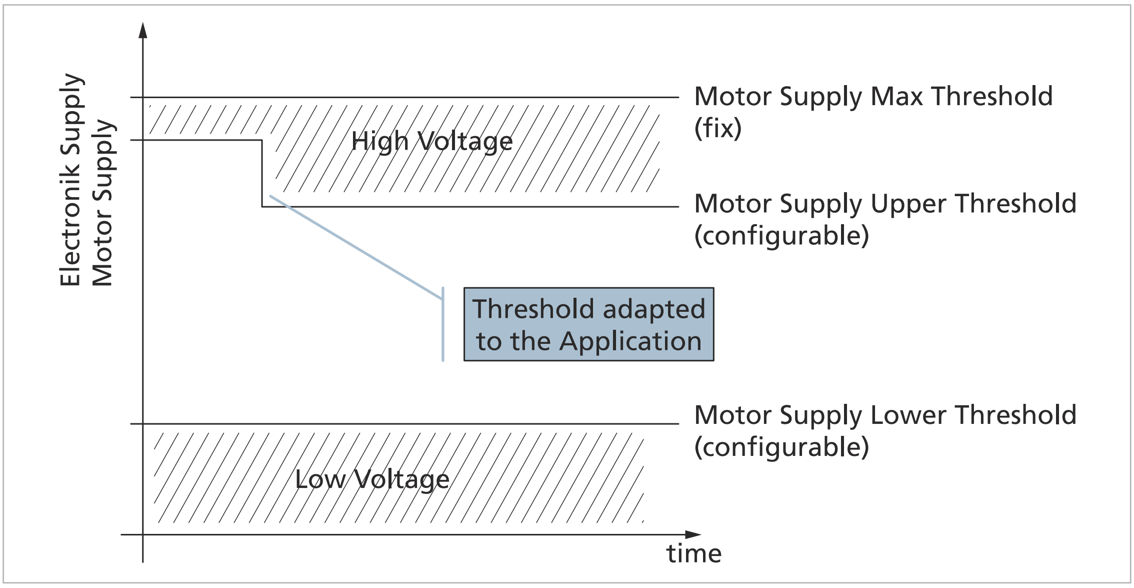


Fig. 69: Checking the power supplies

The limit values and the current values for the power supplies can be found in the object 0x2325:

- The maximum upper threshold (0x2325.03) and the minimum voltage for the electronics power supply (0x2325.01) are specified as fixed values.
- The lower threshold for the motor power supply (0x2325.02) can be adapted to the application.
- A delay time for error detection can be adapted using the object 0x2325.05.
- The upper threshold for the motor power supply (0x2325.04) is variable and can be raised to the maximum voltage.
- If the power supply at the start of monitoring is already above the upper limit or below the lower limit, the change of state into the *Operation Enabled* state will not be performed.

- In dynamic operation it may occur that due to the regeneration of energy in braking mode the supply voltage rises above the limit value defined by the upper threshold. In this case the overvoltage controller is activated and, if necessary, the braking power is reduced.

NOTICE!

Using the default settings, the FAULHABER Motion Controller does not limit the regenerated energy until the default value for the upper limit has been exceeded. Therefore for Motion Controllers with a permissible maximum voltage of 50 V, voltage peaks up to 50 V can be tolerated without affecting the dynamic behaviour. If the drive is operating on a 24 V power supply system, this can lead to malfunctions by additional connected devices and even cause them irreparable damage.

- Set the upper limit for the power supply to a lower value.

FAULHABER Motion Controllers MC 5010, MC 5005 and MC 5004 are equipped with a Power Good LED. As long as the monitored voltages lie within the set limits, the LED lights up green.

If the power supply remains outside the permissible ranges for longer than specified in the delay time, the relevant status bit is set in the device statusword. The Power Good LED is then no longer lit.

7.1.3 Temperature models for the motor winding and the output stage

As a rule, small drives have no sensors for the winding temperature. The winding temperature is estimated using a thermal model, in order to protect the motor against overloading, allowing the full dynamic response. In addition the temperature of the output stage is estimated:

- The measured or estimated temperatures, together with the respective limit values are summarised in the object 0x2326.
- The model parameters for the motor can be found in the object 0x232A. For Motion Controllers, the model parameters are written by the Motion Manager motor selection wizard; for drives with integrated electronics (Motion Control Systems) these values are pre-defined by the factory.
- The following parameters allow the motor data to be adapted to the installation situation.
  - Ambient temperature (0x232A.08)
  - Reduction of the thermal resistance (0x232A.09) arising from the installation
  - Thermally permissible continuous current (0x2329.02)

Depending on the installation situation it may be necessary to enter in the object 0x2329.02 a value for the thermally permissible continuous current different to the value listed in the motor datasheet. The thermally permissible current in an application does not exceed the permissible temperatures for the winding or the (integrated) electronics. Therefore if the thermal coupling is good, it may be possible to tolerate a continuous current higher than that listed in the datasheet. In unfavourable installation situations the continuous current may also be lower than that listed in the datasheet.

Temperatures exceeding the warning and limit temperatures are signalled in the device statusword. As soon as one of the estimated temperatures exceeds the warning threshold set by the values in the object 0x2326, the limit value for the current controller is additionally limited to the thermally permissible continuous current. The full range of dynamic behaviour is not available again until the drive has cooled down sufficiently.

7.1.4 Device statusword 0x2324.01

Tab. 49: Meaning of the entries of the device statusword

Bit	Mask	Meaning	Description
0	0x 00 00 00 01	n=0 monitoring	The actual speed lies within the speed corridor around the 0.
1	0x 00 00 00 02	Target speed reached	The actual speed lies within the speed corridor around the target velocity (evaluated only in the PV operating mode).
2	0x 00 00 00 04	Speed deviation lies outside the corridor	The control deviation of the velocity controller (setpoint value – actual value) lies outside the corridor defined by 0x2344.03 and 0x2344.04 (blockage detection).
3	0x 00 00 00 08	Target position reached	The actual position lies within the speed corridor around the target position (evaluated only in the PP operating mode).
4	0x 00 00 00 10	Position has crossed the setpoint value	The actual position has crossed the setpoint value position (evaluated only in the PP operating mode).
5	0x 00 00 00 20	Following error outside the corridor	The control deviation of the position controller (setpoint value – actual value) lies outside the corridor defined by 0x6065 and 0x6066.
6	0x 00 00 00 40	Positive limit switch active	The positive limit switch is active.
7	0x 00 00 00 80	Negative limit switch active	The negative limit switch is active.
8	0x 00 00 01 00	Positive software position limit reached	The actual position is greater than or equal to the positive software position limits.
9	0x 00 00 02 00	Negative software position limit reached	The actual position is greater than or equal to the negative software position limits.
10	0x 00 00 04 00	Reference input detected [a)]	The configured reference input was detected.
11	0x 00 00 08 00	Encoder index detected [a)]	The configured encoder index was detected.
12	0x 00 00 10 00	The drive is referenced	The drive has performed at least one successful reference run.
13	0x 00 00 20 00	Voltage limited	The motor control is at the voltage limit (output limited).
14	0x 00 00 40 00	Torque limited	The motor control is at the current limit (setpoint value limited).
15	0x 00 00 80 00	Speed limited	The motor control is at the speed limit (setpoint value limited).
16	0x 00 01 00 00	Temperature warning limit reached	One of the monitored temperatures has exceeded the warning limit. The setpoint value of the current controller will be set to the continuous current of the motor.
17	0x 00 02 00 00	Temperature cut-out limit reached	One of the monitored temperatures has exceeded the cut-out limit.

Bit	Mask	Meaning	Description
18	0x 00 04 00 00	Supply voltage too high	One of the monitored voltages has exceeded the upper threshold. ▪ Electronics power supply > max threshold ▪ Motor power supply > upper threshold The overvoltage controller may have tripped.
19	0x 00 08 00 00	Electronics power supply too low	The electronics power supply lies below the lower threshold set in 0x2325.01.
20	0x 00 10 00 00	Motor power supply too low	The motor power supply lies below the lower threshold set in 0x2325.02.
21	0x 00 20 00 00	Velocity error	The velocity is higher than the maximum motor velocity set by 0x2344.05.
22	0x 00 40 00 00	Reserved	-
23	0x 00 80 00 00	Reserved	-
24	0x 01 00 00 00	DigIn01	DigIn01
25	0x 02 00 00 00	DigIn02	DigIn02
26	0x 04 00 00 00	DigIn03	DigIn03
27	0x 08 00 00 00	DigIn04	DigIn04
28	0x 10 00 00 00	DigIn05	DigIn05
29	0x 20 00 00 00	DigIn06	DigIn06
30	0x 40 00 00 00	DigIn07	DigIn07
31	0x 80 00 00 00	DigIn08	DigIn08

a) evaluation only during active reference run

7.1.5 Indication of the dynamic state via the status LED

The dynamic state of the drive is signalled by the status LED.

Tab. 50: Available displays of the status LED

Colour	Status	Meaning
Green	Flashing	The drive is ready. However the state machine has not yet reached the *Operation Enabled* state. The controller and output stage are switched off.
Green	Continuously lit	The drive is ready, the output stage is switched on.
Red	Continuously flashing	The drive has switched to a fault state. The output stage will be switched off or has already been switched off.
Red	Flash code	Boot procedure failed. Please contact FAULHABER Support.
Red	Continuously lit	The device is in boot loader mode.

7.1.6 Status port

Any desired configurable combination of status bits can be signalled via a digital output of the Motion Controller. For this purpose a maximum of 4 ports can be defined in the object 0x2312.

One of the digital outputs of the device is switched if the respective pin is selected. If a 0 is entered as a pin, the function of that port is switched off.

The mask allows definition of which bits in the device statusword are to be evaluated. The selected output is set if the bit-based combination of the mask configured for the port and the device statusword yields at least one set status bit.

Output = (mask & device statusword) > 0 (bit-based link)

Parameter	Port 1	Port 2	Port 3	Port 4
Pin selection	0x2312.03	0x2312.05	0x2312.07	0x2312.09
Mask (U 32)	0x2312.04	0x2312.06	0x2312.08	0x2312.0A

7.1.7 Additional bits in the statusword 0x6041

Instead of digital outputs, bits 14 and 15 in the statusword of the device control (0x6041) can be used to signal selected device states. The evaluation logic corresponds to the status ports:

The additional bit is set if the bit-based combination of the mask configured for the port and the device statusword yields at least one set status bit.

Additional bit = (mask & device statusword) > 0 (bit-based link)

Otherwise the additional bit is reset.

Parameters	Statusword bit 14	Statusword bit 15
Mask (U 32)	0x233A.01	0x233A.02

7.2 Error handling

FAULHABER Motion Controllers and Motion Control Systems have two mechanisms available for error handling:

- Error handling according to CiA 402 (Servodrive Profile):

  Permits response to communication errors in CANopen and EtherCAT networks. If communication is absent or malfunctioning, the drive can be stopped or switched off.

  In addition object 0x2400.04 in the communication settings can be used to set whether one of the available buses should be monitored for errors.

- FAULHABER error word 0x2320:

  Permits response both to communication errors and to equipment faults or application errors.

Communication Settings (0x2400.04)
Object 0x2400.04 in the communication settings can be used to define what message types should be sent via each of the various interfaces.

Index	Byte 3	Byte 2	Byte 1	Byte 0
0x2400.04	Reserved	RS232 settings	USB settings	CANopen settings

The communications settings are bit-coded:

Code	Designation	Description
0x00 00 00 01	CAN-NMT mandatory	The drive function can be started only when the CANopen or EtherCAT interface has reached the status *Operational* via its network management (NMT). The loss of communications can then be handled as an error within the error response procedure according to CiA 402.
0x00 00 00 02	Transmit async PDOs and EMCYs via CAN	Controls the transmission of asynchronous messages at changes of state in the CiA 402 drive state machine and the transmission of EMCY messages.
0x00 00 01 00	Transmit EMCYs via USB	Controls the transmission of error messages via the USB interface.
0x00 00 02 00	Transmit async Messages via USB	Controls the transmission of asynchronous messages at changes of state of the CiA 402 drive state machine via the USB interface.
0x00 01 00 00	Transmit EMCYs via RS232	Controls the transmission of error messages via the RS232 interface. No error messages are transmitted if the drive is in net mode (several drives at one RS232 interface).
0x00 02 00 00	Transmit async Messages via RS232	Controls the transmission of asynchronous messages at changes of state of the CiA 402 drive state machine via the RS232 interface No asynchronous messages are transmitted if the drive is in net mode (several drives at one RS232 interface).

i The settings at object 0x2400.04 are effective for both error handling mechanisms.

Response to communications errors

The following table shows the errors to which the two error handling mechanisms permit a response.

Error type	Error	Error handling to CiA 402 [a)]	FAULHABER error word 0x2320 [b)]
Protocol errors	PDO with the wrong length	–	✓
Low-level errors	Bus off	✓	✓
	Buffer overflow	–	✓
Unexpected change of state	Exit *Operational* status	✓	–
Guarding or heartbeat errors	–	✓	✓

a) *To determine the response according to CiA 402 see chap. 7.2.1, p. 143*

b) *To determine the response to the FAULHABER error word see chap. 7.2.2, p. 144*

7.2.1 Error handling to according CiA 402 (servo drive profiles)

For drives according to CiA 402 Servodrive, the response to a communications error can be specified via object 0x6007.00.

Value	Response to communications errors
0	No response. The drive continues to advance to the set setpoint.
1	The drive switches into the error state of the CiA 402 drive state machine. ▪ Depending on the settings in the object 0x605E (Fault Option Code) the drive can be brought to a controlled stop. ▪ If a holding brake is configured, it is activated.
2	The drive switches into the *Switch On Disabled* state of the CiA 402 drive state machine. The change is equivalent to the command **Disable Operation**. ▪ The drive is not brought to a stop, instead it coasts on. ▪ If a holding brake is configured, it is activated.
3	Switch into the *Quick Stop* state of the CiA 402 drive state machine. All outstanding movement jobs are discarded. The settings in the object 0x605A (Quick Stop Option Code) can be used to specify the following responses: ▪ The drive is brought to a controlled stop. ▪ The drive is then held in a controlled state at n = 0 or after completion of the braking operation the output stage is switched off.

7.2.2 Error handling with the FAULHABER error word

7.2.2.1 Error detection

Errors are collected from the device diagnostics, the communications interface and the hardware drivers and combined in the FAULHABER error word (0x2320). The mask in the object 0x2321 allows definition of how the system responds to the errors that are detected.

Each error is additionally assigned an error code as listed in CiA 301/CiA 402, which is loaded in addition to an EMCY error message.

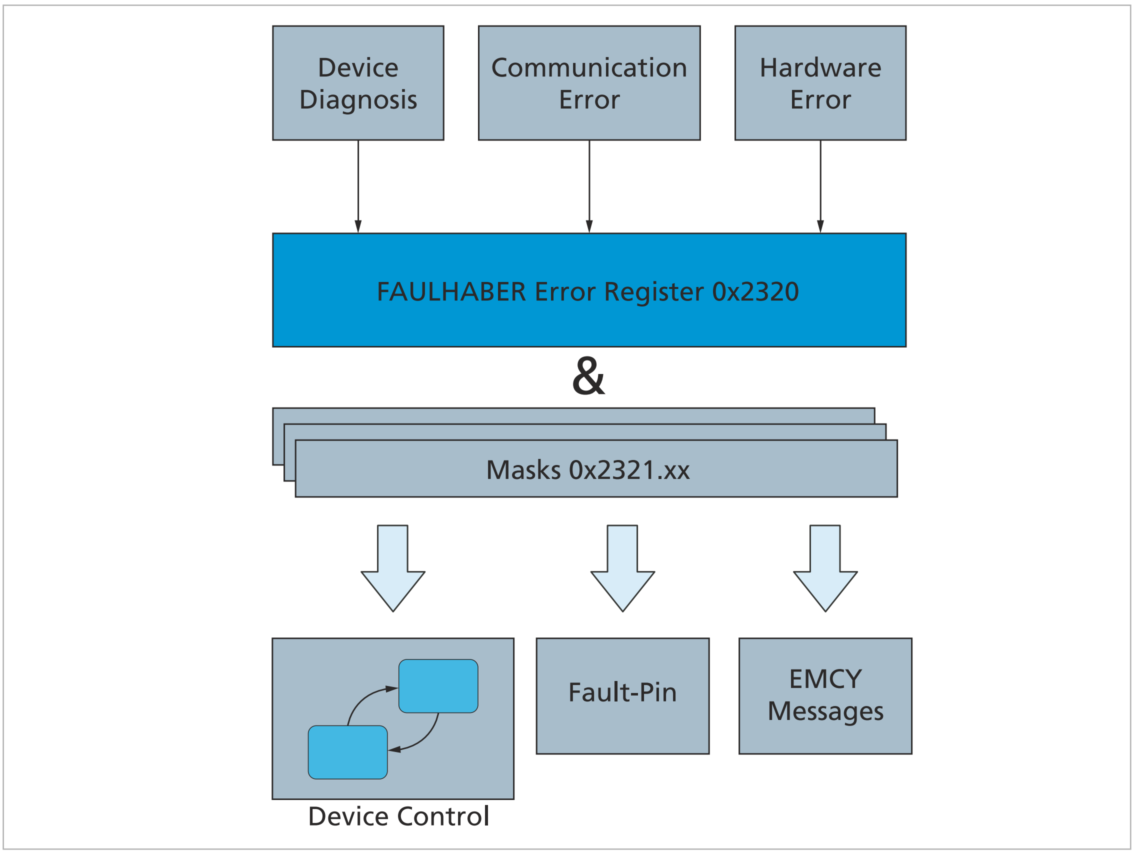


Fig. 70: Diagram of error detection

Tab. 51: Overview of settings for the response to faults

Object	Mask
0x2321.01	Emergency mask: identifies faults for which an EMCY message is to be sent. Default: 0xFFFF
0x2321.02	Error mask: identifies faults for which the drive is switched into the *Fault* state of the drive state machine. Default: 0x0000
0x2321.03	Fault pin mask: identifies faults for which the fault pin is to be activated. Default: 0x0000
0x2321.04	Disable voltage factory mask: identifies faults for which the drive must be switched off without switching to the fault state. The disable voltage factory mask cannot be changed by the user, and takes effect in the event of overvoltage errors and temperature errors Default: 0x024

Object	Mask
0x2321.05	Disable voltage mask: identifies faults for which the drive is to be switched off without switching to the fault state. The disable voltage mask can be changed by the user. Default: 0x0000
0x2321.06	Quick stop mask: identifies faults for which the drive is to be switched into the *Quick Stop* state. The Quick Stop option code is respected.

7.2.2.2 FAULHABER error word 0x2320

Tab. 52: Meaning of the entries of the FAULHABER error word

Bit	Mask	Meaning	CiA 402 error code
0	0x 00 00 00 01	SpeedDeviationError	Deviation error: 0x84F0
1	0x 00 00 00 02	FollowingError	FollowingError: 0x8611
2	0x 00 00 00 04	OverVoltageError	Overvoltage: 0x3210
3	0x 00 00 00 08	UnderVoltageError a)	Undervoltage: 0x3220
4	0x 00 00 00 10	TempWarning	Temperature warning: 0x43F0
5	0x 00 00 00 20	TempError	Temperature Error: 0x4320
6	0x 00 00 00 40	Encoder error	Sensor fault: 0x7300 ▪ Analogue Hall signals are defective or not compensated ▪ Impermissible combination of digital Hall signals ▪ CRC checking of an AES encoder returns an error
7	0x 00 00 00 80	Int HW error	Output stages: 0x5410
8	0x 00 00 01 00	Reserved	–
9	0x 00 00 02 00	Current Measurement Error	Current Measurement Error: 0x7200
10	0x 00 00 04 00	Reserved	–
11	0x 00 00 08 00	Com Error	CAN overrun: 0x8110 CAN guarding failed: 0x8120 CAN recovered from bus off: 0x8140 CAN PDO length: 0x8210 RS232Overrun: 0x8310 Process data watchdog timeout: 0x8130 PDO length: 0x8210
12	0x 00 00 10 00	Calc error	Software error: 0x6100
13	0x 00 00 20 00	Dynamic Limit	Dynamic Limit: 0x84FF
14	0x 00 00 40 00	Reserved	–
15	0x 00 00 80 00	Reserved	–

a) The shortfall below the lower threshold of the motor power supply is monitored only if a value > 0 V is entered as the threshold in the object 0x2325.02.

The assignment of the error register is specified by the CiA 301.

7.2.2.3 Switching off the error response of the drive

For each error that is detected a check is made whether a response by the state machine of the drive should or must be generated.

The status is switched if the bit-wise AND link of the FAULHABER errorword to the error mask defined in the object 0x2321 yields at least one match. For this purpose the check is performed in the following sequence:

Prio	Description	Object	Default value
1	Disable voltage factory mask	0x2321.04	0x 00 24
2	Disable voltage mask	0x2321.05	0x 00 00
3	Quick-Stop Mask	0x2321.06	0x 00 00
4	Error mask	0x2321.02	0x 00 00

The disable voltage factory mask is pre-defined at the factory. In the following cases the drive is switched off:

- If one of the power supplies is above the threshold specified in object 0x2325.04 for longer than the tolerance time specified (in object 0x2325.05).
- If the electronics power supply is below the threshold specified in object 0x2325.01 for longer than the tolerance time specified (in object 0x2325.05).
- If one of the monitored temperatures exceeds the cut-out threshold.
- If one of the power supplies exceeds the maximum value for the power supply (object 0x2325.03) by more than 30%.

In these cases the drive is brought to a standstill, no longer following a ramp. If a holding brake is configured, it is activated.

If an error which causes the drive to be switched off is present the drive cannot be enabled (see the following example).

Example

An undervoltage is configured as an error and should cause the drive to be switched off (Disable Voltage Mask).

The motor power supply voltage is less than the configured lower threshold.

Consequently: The drive cannot be switched on as long as the undervoltage remains detected.

7.2.2.4 Sending out an error message in response to an error

The EMCY mask in the object 0x2321.01 allows selection of which of the detected errors triggers the sending of an error message via the communications system. The default setting is that every detected errors triggers a message.

Records of the error messages:

- The error register 0x1001
- The CiA error code
- The FAULHABER error register 0x2320

Tab. 53: Assignment of the error entries in the FAULHABER error word 0x2320 to the error register 0x1001

Bit	Meaning	Assignment of the FAULHABER error word (0x2320) to the bits in the error register (0x1001)
0	Generic	Bit 0 will be set to 1 if a further error is signalled.
1	Current error	Bit 4: TempWarning
2	Voltage error	Bit 2: OverVoltageError or Bit 3: UnderVoltageError
3	Temperature error	Bit 5: TempError
4	Error from a communications stack	Bit 11: Com Error
5	Drive-specific error	Bit 0: SpeedDeviationError or Bit 1: FollowingError
6	not used	-
7	Manufacturer-specific error	All HW errors (bit 6 - bit 9 and bit 12)

The latest 8 error messages (EMCY) are held in the error log of the device and can be read via the object 0x1003. The number of messages held in the error log can be viewed via the entry 0x1003.00.

Error messages are sent out asynchronously without being explicitly requested by the supervisory control system. The message type used depends on the communications system. The options for the communications interfaces (object 0x2400.04) allow selection of the interfaces via which an asynchronous message will be sent.

If RS232 net mode is activated, no asynchronous messages can be sent via the RS232 interface. No further error messages are transmitted automatically via the RS232 interface.

7.2.2.5 Setting the error response fault pin

The fault pin mask in the object 0x2321.03 allows selection of the detected errors for which the fault pin should be set. In the default setting no errors are signalled at the fault pin.

The digital output that is used as the fault pin can be configured via the object 0x2312.01. The factory default is that there is no pre-configured fault pin.

8 Parameter description

8.1 Manufacturer-specific objects

Number of I/Os (0x2300)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2300	0x00	Number of entries	U8	ro	3	Number of object entries
	0x01	Number of Digital Inputs	U8	ro		Number of digital inputs
	0x02	Number of Digital Outputs	U8	ro		Number of digital outputs
	0x03	Number of Analogue Inputs	U8	ro		Number of analogue inputs

Digital Input Settings (0x2310)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2310	0x00	Number of entries	U8	ro	18	Number of object entries
	0x01	Select Lower Limit Switch Inputs	U8	rw	0	Selection of the lower limit switch
	0x02	Select Upper Limit Switch Inputs	U8	rw	0	Selection of the upper limit switch
	0x03	Limit Switch Option Code	S16	rw	2	Limit Switch Option Code
	0x04	Select Reference Switch Input	U8	rw	0	Determine the digital input of the reference switch
	0x10	Input Polarity	U8	rw	0	Polarity of the inputs
	0x11	Input Threshold Level	U8	rw	0	Set the switching level for all digital inputs: ▪ 0: 5V-TTL ▪ 1: 24V-PLC
	0x12	Input Filter Active	U8	rw	0	Bit-coded for 8 inputs. ▪ 0: No flank filter for the input ▪ 1: Flank filter for the input active

Settings of the digital inputs in accordance with the bit mask in Tab. 19

Digital I/O Status (0x2311)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2311	0x00	Number of entries	U8	ro	4	Number of object entries
	0x01	Digital Input Logical State	U8	ro	0	Logical states of the digital inputs
	0x02	Digital Input Physical State	U8	ro	0	Physical states of the digital inputs
	0x03	Digital Output Status	U8	ro	0	Status of the digital output
	0x04	Write Digital Outputs	U16	rw	0	Directly setting, deleting, toggling digital outputs.

States of the digital inputs in accordance with the bit mask in Tab. 19

Digital Output Settings (0x2312)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2312	0x00	Number of entries	U8	ro	16	Number of object entries
	0x01	Select Fault Output Pin	U8	rw	0	Number of digital outputs to be used as fault pins.
	0x02	Select Brake Control Pin	U8	rw	0	Number of digital outputs to be used for actuation of a holding brake
	0x03	Brake Delay Time	U8	rw	0	Brake delay
	0x08	DiagOutput 1 Pin Selection	U8	rw	0	See chap. 4.6.3.3, p. 57
	0x09	DiagOutput 1 Mask	U32	rw	0	See chap. 4.6.3.3, p. 57
	0x0A	DiagOutput 2 Pin Selection	U8	rw	0	See chap. 4.6.3.3, p. 57
	0x0B	DiagOutput 2 Mask	U32	rw	0	See chap. 4.6.3.3, p. 57
	0x0C	DiagOutput 3 Pin Selection	U8	rw	0	See chap. 4.6.3.3, p. 57
	0x0D	DiagOutput 3 Mask	U32	rw	0	See chap. 4.6.3.3, p. 57
	0x0E	DiagOutput 4 Pin Selection	U8	rw	0	See chap. 4.6.3.3, p. 57
	0x0F	DiagOutput 4 Mask	U32	rw	0	See chap. 4.6.3.3, p. 57
	0x10	Output Polarity	U8	rw	0	Polarity of the outputs

Analog Inputs (0x2313)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2313	0x00	Number of entries	U8	ro	21	Number of object entries
	0x01	AnIn 1 Gain (Numerator/ Divisor)	S32	rw	0x7FFF8000	AnIn 1 Gain (Numerator/Denominator) ▪ Bit 1...16: Denominator ▪ Bit 17...32: Numerator
	0x02	AnIn 1 Offset	S16	rw	0	AnIn 1 Offset
	0x03	AnIn 1 Filter Time	U16	rw	0	AnIn 1 Filter time in 100 µs
	0x04	AnIn 1 User Scaled Value	S32	ro	-	Scaled AnIn 1 value
	0x05	AnIn 1 Resolu-tion as Encoder	U16	rw	1000	AnIn 1 Resolution of the encoder
	0x06	AnIn 1 Min Input Limit	S16	rw	–32768	AnIn 1 Lower limit for the input value
	0x07	AnIn 1 Max Input Limit	S16	rw	32767	AnIn 1 Upper limit for the input value
	0x08	AnIn 1 Select Dir Pin	U8	rw	0	AnIn 1 Polarity input: 0: No polarity input used 1...8: Digital input used as polarity input
	0x11	AnIn 2 Gain (Numerator/ Divisor	S32	rw	0x7FFF8000	AnIn 2 Gain (Numerator/Denominator) ▪ Bit 1...16: Denominator ▪ Bit 17...32: Numerator
	0x12	AnIn 2 Offset	S16	rw	0	AnIn 2 Offset
	0x13	AnIn 2 Filter Time	U16	rw	0	AnIn 2 Filter time in 100 µs
	0x14	AnIn 2 User Scaled Value	S32	ro	-	Scaled AnIn 2 value
	0x15	AnIn 2 Resolu-tion as Encoder	U16	rw	1000	AnIn 2 Resolution of the encoder
	0x16	AnIn 2 Min Input Limit	S16	rw	–32768	AnIn 2 Lower limit for the input value
	0x17	AnIn 2 Max Input Limit	S16	rw	32767	AnIn 2 Upper limit for the input value
	0x18	AnIn 2 Select Dir Pin	U8	rw	0	AnIn 2 Polarity input: 0: No polarity input used 1...8: Digital input used as polarity input

Manufacturer Scaled Analog Input Values (0x2314)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2314	0x00	Number of entries	U8	ro	12	Number of object entries
	0x01	AnIn 1 Scaled Value (IA)	S16	ro	-	Scaled value at input 1 (IA) [a)]
	0x02	AnIn 2 Scaled Value (IB)	S16	ro	-	Scaled value at input 2 (IB) [a)]
	0x03	AnIn 3 Scaled Value (IC)	S16	ro	-	Scaled value at input 3 (IC) [a)]
	0x04	AnIn 4 Scaled Value (Hall A)	S16	ro	-	Scaled value at input 4 (Hall A)
	0x05	AnIn 5 Scaled Value (Hall B)	S16	ro	-	Scaled value at input 5 (Hall B)
	0x06	AnIn 6 Scaled Value (Hall C)	S16	ro	-	Scaled value at input 6 (Hall C)
	0x07	AnIn 7 Scaled Value	S16	ro	-	Scaled value at input 7
	0x08	AnIn 8 Scaled Value	S16	ro	-	Scaled value at input 8
	0x09	AnIn 9 Scaled Value	S16	ro	-	Scaled value at input 9
	0x0A	AnIn 10 Scaled Value0	S16	ro	-	Scaled value at input 10
	0x0B	AnIn 11 Scaled Value	S16	ro	-	Scaled value at input 11
	0x0C	AnIn 12 Scaled Value	S16	ro	-	Scaled value at input 12

a) *A value of 1000 corresponds to the device rated current set in object 2327.01*

Motor Encoder (0x2315)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2315	0x00	Number of entries	U8	ro	3	Number of object entries
	0x01	Operation Mode, Index Polarity	U16	rw	0	Selection of the encoder type
	0x02	Resolution	U16	rw	0x0800	Encoder resolution in increments/revolutions (4-flank evaluation)
	0x03	Motor Encoder Position	S32	ro		Current position value

Reference Encoder (0x2316)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2316	0x00	Number of entries	U8	ro	5	Number of object entries
	0x01	Operation Mode, Index Polarity	U16	rw	0	Selection of the additional encoder
	0x02	Resolution	U16	rw	2048	Encoder resolution
	0x03	Reference Encoder Position	S32	ro	0	Encoder position
	0x04	Gain (Numerator / Divisor)	S32	rw	0x40004000	Conversion of the raw value of the external encoder into the position value used internally
	0x05	Offset	S32	wo	0	Position offset for the reference encoder.

PWM Input (0x2317)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2317	0x00	Number of entries	U8	ro	7	Number of object entries
	0x01	Digital Input Pin	U8	rw	0	PWM input: 1: DigIn1 = PWM input 2: DigIn2 = PWM input
	0x02	PWM Input Frequency	U32	ro		Frequency of the PWM signal
	0x03	Duty Cycle Raw Value	S16	ro		Duty cycle of the PWM signal (unscaled)
	0x04	Duty Cycle Gain (Numerator / Divisor)	U32	rw	0x7FFF8000	PWM In Gain (numerator / denominator)
	0x05	Duty Cycle Offset	S16	rw	0	PWM In Offset
	0x06	Duty Cycle Scaled Value	S32	ro		Scaled pulse width
	0x07	Resolution As Encoder	S16	rw	1000	Resolution of the encoder

Analog Hall Configuration (0x2318)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2318	0x00	Number of entries	U8	ro	3	Number of object entries
	0x01	Hall Sensor Type	U8	rw	0	Bit-coded selection of the sensor type (see chap. 4.6.4.3, p. 60)
	0x02	Enable Adaptation	U8	rw	0	0: static adjustment switched off 1: static adjustment active
	0x03	Adaption Threshold Speed	U32	rw	1000	Minimum velocity in [min^{-1}], from which the Hall signals are compensated.

FAULHABER Error Register (0x2320)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2320	0x00	FAULHABER Error Register	U16	ro	0	FAULHABER error word (see chap. 7, p. 136)

Error mask (0x2321)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2321	0x00	Number of entries	U8	ro	6	Number of object entries
	0x01	Emergency Mask	U16	rw	0x00FF	Errors for which an error message is sent
	0x02	Fault Mask	U16	rw	0x0000	Errors for which the state machine of the drive switches into *Fault Reaction Active* state
	0x03	Error Out Mask	U16	rw	0x00FF	Errors for which the error output pin is set
	0x04	Disable Voltage Mask	U16	ro	0x0000	Errors which switch the drive off (not configurable)
	0x05	Disable Voltage User Mask	U16	rw	0x0000	Errors which switch the drive off (configurable)
	0x06	Quick Stop Mask	U16	rw	0x0000	Errors for which the state machine of the drive switches into *Quick Stop Active* state

Device Status (0x2324)

The actual device status is monitored via the device status object (0x2324).

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2324	0x00	Number of entries	U8	ro	1	Number of object entries
	0x01	Device Status Word	U32	ro	0	Device status

Voltage Monitor (0x2325)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2325	0x00	Number of entries	U8	ro	7	Number of object entries
	0x01	Device Supply Lower Threshold	U16	ro	1200	Lower threshold value of the power supply of the device [a)]
	0x02	Motor Supply Lower Threshold	U16	rw	1200	Lower threshold value of the power supply of the motor [a)]
	0x03	Motor Supply Max Threshold	U16	ro	5200	Maximum threshold value of the power supply of the motor [a)]
	0x04	Motor Supply Upper Threshold	U16	rw	5200	Upper threshold value of the power supply of the motor [a)]
	0x05	Voltage Error Delay Time	U16	rw	200	Delay time in ms until a voltage error is signalled

Index	Subindex	Name	Type	Attr.	Default value	Meaning
	0x06	Device Supply Voltage	U16	ro	-	Actual power supply of the electronics
	0x07	Motor Supply Voltage	U16	ro	-	Actual power supply of the motor

a) *All voltages are in 10 mV per digit*

Device Temperature (0x2326)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2326	0x00	Number of entries	U8	ro	9	Number of object entries
	0x01	CPU Temperature	S16	ro	0	Temperature of the processor [°C]
	0x02	Power Stage Temperature	S16	ro	0	Temperature of the output stage [°C]
	0x03	Winding Temperature	S16	ro	0	Temperature of the winding [°C]
	0x04	CPU Temperature Shutdown Threshold	S16	ro	115	Temperature cut-out threshold of the processor [°C]
	0x05	CPU Temperature Warning Threshold	S16	ro	105	Temperature warning threshold of the processor [°C]
	0x06	Power Stage Temperature Shutdown Threshold	S16	ro	140	Temperature cut-out threshold of the output stage [°C]
	0x07	Power Stage Temperature Warning Threshold	S16	ro	135	Temperature warning threshold of the output stage [°C]
	0x08	Winding Temperature Shutdown Threshold	S16	rw	125	Temperature cut-out threshold of the motor [°C]
	0x09	Winding Temperature Warning Threshold	S16	rw	115	Temperature warning threshold of the motor [°C]

Device Data (0x2327)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2327	0x00	Number of entries	U8	ro	3	Number of object entries
	0x01	Device Nominal Current	U16	ro	Device-specific	Nominal current at the Motion Controller
	0x02	Device Peak Current	U16	ro	Device-specific	Peak current of the device
	0x03	Device Nominal Voltage	U16	ro	Device-specific	Nominal voltage of the device

Device Data Thermal Model (0x2328)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2328	0x00	Number of entries	U8	ro	10	Number of object entries
	0x01	Controller Type	U8	ro	Device-specific	0: no MCS 1: MCS
	0x02	Power Stage R_{dson}	U8	ro	Device-specific	On-state resistance of a MOSFET [mOhm]
	0x03	Power Stage LossFactor	U8	ro	Device-specific	Factor for internal calculation of the switching losses of the output stage
	0x04	Power Stage R_{th1}	U16	ro	Device-specific	Thermal resistance of the output stage
	0x05	Power Stage Time Constant 1	U16	ro	Device-specific	Time constant of the output stage
	0x06	Power Stage R_{th2}	U16	ro	Device-specific	Thermal resistance of the output stage
	0x07	Power Stage Time Constant 2	U16	ro	Device-specific	Time constant of the output stage
	0x08	Power Stage R_{th3}	U16	ro	Device-specific	Thermal resistance of the output stage
	0x09	Power Stage R_{th4}	U16	ro	Device-specific	Thermal resistance of the output stage

Motor and Application Data / Motor Control (0x2329)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2329	0x00	Number of entries	U8	ro	11	Number of object entries
	0x01	Rated Current	U16	rw	–	Rated current of the motor [mA]
	0x02	Continuous Current	U16	rw	–	Continuous current of the motor [mA]
	0x03	Peak Current	U16	rw	–	Peak current of the motor [mA]
	0x04	Torque Constant / Force Constant	U32	rw	–	▪ For rotating motors: Torque constant km [mNm x $1e^{-3}$] ▪ For linear motors: Force constant kf [N/A x $1e^{-3}$]
	0x05	Terminal Inductance	U16	rw	–	Connection inductance LA of the motor [µH]
	0x06	Inductance L_d	U16	rw	–	Longitudinal inductance L_d of the motor [µH]; if not explicitly declared set the longitudinal inductance to the same value as the connection inductance
	0x07	Number of Pole Pairs	U8	rw	–	Number of pole pairs
	0x08	Phase Angle Offset	S16	rw	0	Phase angle offset, 32767 digits = 180° electrical
	0x09	Rotor Inertia / Rod weight	U32	rw	–	▪ For rotating motors: Mass inertia of the rotor J_{Motor} [gcm² x $1e^{-3}$] ▪ For linear motors: Mass of the traveller m_{Motor} [g]

Index	Subindex	Name	Type	Attr.	Default value	Meaning
	0x0A	Load Inertia / Load Mass	U32	rw	–	▪ For rotating motors: Mass inertia of the load translated to the motor [gcm² x 1e^{-3}] ▪ For linear motors: Mass of the load translated to the motor [g]
	0x0B	Motor Type	U8	rw	0	0: rotating motor 1: linear motor (with traveller)
	0x0C	Magnetic Pitch of Linear Motor	U8	rw	–	Pole spacing of the linear motor [mm]

Motor and Application Data / Thermal Motor Model (0x232A)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x232A	0x00	Number of entries	U8	ro	9	Number of object entries
	0x01	Terminal Resistance	U32	rw	a)	Connection resistance R of the motor in mOhm
	0x02	Friction	U32	rw	a)	Frictional torque [mNm x 1e^{-6}], equivalent to c0 for BL motors and MR for DC motors
	0x03	Friction, dynamic	U32	rw	a)	Dynamic frictional torque cv [mNm/rpm x 1e^{-9}] (only for BL motors)
	0x04	Thermal Resistance 1	U16	rw	a)	Thermal resistance 1 of the motor [mK/W] (winding to housing)
	0x05	Thermal Resistance 2	U16	rw	a)	Thermal resistance 2 of the motor [mK/W] (housing to environment)
	0x06	Thermal Time Constant 1	U16	rw	a)	Thermal time constant 1 of the motor [s] (winding to housing)
	0x07	Thermal Time Constant 2	U16	rw	a)	Thermal time constant 2 of the motor [s] (housing to environment)
	0x08	Ambient Temperature	U8	rw	a)	Ambient temperature of the motor [°C]
	0x09	Reduction of Thermal Resistance 2	U8	rw	a)	Reduction of the thermal resistance 2 of the motor [%] (housing to ambient)

a) *Motor-specific, is set by the motor selection wizard*

Switch Position for Actual Values (0x2330)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2330	0x00	Number of entries	U8	ro	3	Number of object entries
	0x01	Actual Commutation Angle Source	U8	rw	1	Commutation angle actual value source
	0x02	Actual Velocity Source	U8	rw	1	Velocity actual value source
	0x03	Actual Position Source	U8	rw	1	Position actual value

Discrete Sources (0x2331)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2331	0x00	Number of entries	U8	ro	4	Number of object entries
	0x01	Target Voltage Source	U8	rw	0	Selection of the discrete source for the voltage setpoint value
	0x02	Target Current Source	U8	rw	0	Selection of the discrete source for the torque setpoint value
	0x03	Target Velocity Source	U8	rw	0	Selection of the discrete source for the speed setpoint value
	0x04	Target Position Source	U8	rw	0	Selection of the discrete source for the position setpoint value

Manufacturer Specified Bits (0x233A)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x233A	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Bit Mask for Bit 14	U32	rw	0x0	Device states that should be shown in the object 0x2324.01 (Device Status Word)
	0x02	Bit Mask for Bit 15	U32	rw	0x0	Device states that should be shown in the object 0x2324.01 (Device Status Word)

Operation Mode Options (0x233F)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x233F	0x00	OpMode Options	U16	rw	0x0001	Bit-coded

See chap. 5.1.3, p. 85.

General parameters (0x2340)

Index	Subindex	Name	Typ e	Attr.	Default value	Meaning
0x2340	0x00	Number of entries	U8	ro	8	Number of object entries
	0x01	Commutation Type	U8	rw	3	Commutation type 0: switched off 1: DC motor 2: BL motor with block commutation 3: BL motor with sinusoidal commutation
	0x02	Motor Output Voltage DC	S16	rw	-	Motor output voltage DC [a)]
	0x03	Motor Output Voltage BL Block	S16	rw	-	Motor output voltage, BL block [a)]
	0x04	Motor Output Voltage X_d	S16	rw	-	Motor output voltage X_d [a)]
	0x05	Motor Output Voltage X_q	S16	rw	-	Motor output voltage X_q [a)]
	0x06	Sinus Output Voltage U_a	U16	ro	-	Phase voltage U_a [a)]

Index	Subindex	Name	Type	Attr.	Default value	Meaning
	0x07	Sinus Output Voltage U_b	U16	ro	-	Phase voltage U_b a)
	0x08	Sinus Output Voltage U_c	U16	ro	-	Phase voltage U_c a)

a) *All voltages are in multiples of 10 mV*

Target Voltage (0x2341)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2341	0x00	Voltage Mode Reference	S16	rw	0	Voltage setpoint of the Voltage mode [10 mV / digit]

Torque Control Parameters (0x2342)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2342	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Gain $K_{P,I}$	U32	rw	a)	Controller gain [mOhm]
	0x02	Integral Time $T_{N,I}$	U16	rw	a)	Controller reset time [µs], Range: 150–600 µs

a) *Motor-specific, is set by the motor selection wizard.*

Flux Control Parameters (0x2343)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2343	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Gain $K_{P,I}$	U32	rw	a)	Controller gain [mOhm]
	0x02	Integral Time $T_{N,I}$	U16	rw	a)	Controller reset time [µs], Range: 150–600 µs

a) *Motor-specific, is set by the motor selection wizard.*

Velocity Control Parameters (0x2344)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2344	0x00	Number of entries	U8	ro	4	Number of object entries
	0x01	Gain K_P	U32	rw	a)	Controller Gain [As $1e^{-6}$]
	0x02	Integral time TN	U16	rw	a)	Controller reset time [100 µs]
	0x03	Velocity Deviation Threshold	U16	rw	65535	Maximum permissible control deviation
	0x04	Velocity Deviation Time	U16	rw	100	Maximal permissible duration of a control deviation outside the corridor
	0x05	Velocity Warning Threshold	U32	rw	30000	Warning threshold for the velocity, see 0x2324.01 bit 21

a) *Motor-specific, is set by the motor selection wizard.*

Velocity Filter Parameters (0x2345)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2345	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Actual Velocity Filter T_F	U16	rw	a)	Filter time T_F [100 µs]
	0x02	Display Velocity Filter	U16	rw	20	Filter time for displaying the actual velocity [100 µs]

a) *Motor-specific, is set by the motor selection wizard.*

Setpoint Velocity Filter Parameters (0x2346)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2346	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Setpoint Velocity Filter T_F	U16	rw	a)	Filter time T_F [100 µs]
	0x02	Setpoint Filter Enable	U8	rw	0	0: inactive 1: active

a) *Motor-specific, is set by the motor selection wizard.*

Gain Scheduling (0x2347)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2347	0x00	Number of entries	U8	ro	1	Number of object entries
	0x01	Gain factor	U8	rw	128	Gain factor (used by the velocity controller in PP mode on the K_P) 0: Reduction of the gain to 0 in the target 128: no variable gain 255: Doubling the gain in the target

Position Control Parameters (0x2348)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2348	0x00	Number of entries	U8	ro	1	Number of object entries
	0x01	Gain K_v	U8	rw	a)	Controller gain [1/s], range: 1–255

a) *Motor-specific, is set by the motor selection wizard.*

Current Feedforward Parameters (0x2349)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2349	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Current FeedForward Factor	U8	rw	0	Factor for the torque or force control 0: 0% activation of the pre-control value 128: 100% pre-control
	0x02	Current FeedForward Delay	U16	rw	0	Setpoint delay: 0: undelayed activation 1: Activation delayed by one sampling

Velocity Feedforward Parameters (0x234A)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x234A	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Velocity feed forward factor	U8	rw	0	Factor for the torque or force control 0: 0% pre-control 128: 100% pre-control
	0x02	Velocity feed forward delay	U16	rw	0	Setpoint delay: 0: undelayed activation 1: Activation delayed by one sampling

Actual Values (0x2360)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2360	0x00	Number of entries	U8	ro	8	Number of object entries
	0x01	Motor Current I_d Actual Value a)	S16	ro	-	Motor current actual value (I_D)
	0x02	Motor Current I_q Actual Value a)	S16	ro	-	Motor current actual value (I_q)
	0x03	Motor Current IBlock Actual Value a)	S16	ro	-	Motor current actual value (IBlock)
	0x04	Motor Current IDC Actual Value a)	S16	ro	-	Motor current actual value (IDC)
	0x05	Velocity Actual Internal Value	S16	ro	-	Actual value the velocity [min^{-1}] (internal value)
	0x06	Position Actual Internal Value	S32	ro	-	Actual value of the position (internal value)

a) *A value of 1000 corresponds to the device rated current set in object 2327.01*

Trace Configuration (0x2370)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2370	0x00	Number of entries	U8	ro	10	Number of object entries
	0x01	Trigger Source	U32	wo	0	Trigger source
	0x02	Trigger Threshold	S32	rw	0	Trigger threshold
	0x03	Trigger Delay Offset	S16	rw	0	Trigger delay
	0x04	Trigger Mode	U16	rw	0	Trigger mode
	0x05	Buffer Length	U16	rw	100	Buffer length
	0x06	Sample Time	U8	rw	1	Recording sampling rate 1: in every sampling step
	0x07	Trace Source of Channel 1	U32	wo	0	Trace source of channel 1
	0x08	Trace Source of Channel 2	U32	wo	0	Trace source of channel 2

Index	Subindex	Name	Type	Attr.	Default value	Meaning
	0x09	Trace Source of Channel 3	U32	wo	0	Trace source of channel 3
	0x0A	Trace Source of Channel 4	U32	wo	0	Trace source of channel 4

Trace Buffer (0x2371)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2371	0x00	Number of entries	U8	ro	5	Number of object entries
	0x01	Trigger State	U16	ro	0	Trigger state
	0x02	Trace Value of Channel 1	Vis string	ro	–	Signal buffer, channel 1
	0x03	Trace Value of Channel 2	Vis string	ro	–	Signal buffer, channel 2
	0x04	Trace Value of Channel 3	Vis string	ro	–	Signal buffer, channel 3
	0x05	Trace Value of Channel 4	Vis string	ro	–	Signal buffer, channel 4

Communication Parameter (0x2400)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2400	0x00	Number of entries	U8	ro	7	Number of object entries
	0x01	CAN Rate	U8	rw	9	CAN rate (automatic Baud rate determination)
	0x02	RS232 rate	U8	rw	3	RS232–Rate
	0x03	Node ID	U8	rw	1	Node number
	0x04	Communication Settings	U32	rw	0x00 03 03 02	Communication settings bit-coded: 0x00 00 00 01: CAN-NMT mandatory 0x00 00 00 02: Transmit async PDOs and EMCYs via CAN 0x00 00 01 00: Transmit EMCYs via USB 0x00 00 02 00: Transmit async Messages via USB 0x00 01 00 00: Transmit EMCYs via RS232 0x00 02 00 00: Transmit async Messages via RS232
	0x05	RS232 Net Mode	U8	rw	0	RS232 net mode
	0x08	Explicit Device ID	U16	rw	0	Identification of the drive

Internal Filter Parameters (0x2502)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x2502	0x00	Number of Entries	U8	ro	12	Number of object entries
	0x01	Filter Time Actual Velocity	U16	rw	256	Filter time, effective on the value in 0x2360.05.

8.2 Objects of the drive profile CiA 402

Abort Connection Option Code (0x6007)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6007	0x00	Abort Connection Option Code	S16	rw	1	Reaction of the controller on loss of connection: 0: no reaction 1: Switch into fault state 2: Switch into the *Switch On Disabled* state 3: Switch into the *Quick Stop* state

See chap. 7.2.1, p. 143.

Controlword (0x6040)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6040	0x00	Controlword	U16	rw	0	Controlword

See chap. 3.2, p. 16.

Statusword (0x6041)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6041	0x00	Statusword	U16	ro	0	Statusword

See chap. 3.3, p. 19.

Quick Stop Option Code (0x605A)

i When the drive exits the *Operation Enabled* state, the motor can be stopped prior to this. The option codes (object 0x605A to 0x605E) define the behaviour at the transition (see CiA 402 and chap. 3.4, p. 21)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x605A	0x00	Quick Stop Option Code	S16	rw	2	Halt options for the **Quick Stop** command

Shut Down Option Code (0x605B)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x605B	0x00	Shut Down Option Code	S16	rw	0	Halt options for the **Shut Down** command

See chap. 3.4, p. 21.

Disable Operation Option Code (0x605C)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x605C	0x00	Disable Operation Option Code	S16	rw	1	Halt options for the **Disable Operation** command

See chap. 3.4, p. 21.

Halt Option Code (0x605D)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x605D	0x00	Halt Option Code	S16	rw	1	Halt options when setting the halt bit in the controlword

See chap. 3.4, p. 21.

Fault Reaction Option Code (0x605E)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x605E	0x00	Fault Reaction Option Code	S16	rw	2	Halt options at the transition into an error state

See chap. 3.4, p. 21.

Modes of Operation (0x6060)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6060	0x00	Modes of operation	S8	rw	0	Select the operating mode –4: ATC –3: AVC –2: APC –1: Voltage Mode 0: Controller not activated 1: PP 3: PV 6: Homing 8: CSP 9: CSV 10: CST

See chap. 5, p. 82.

Modes of Operation Display (0x6061)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6061	0x00	Modes of Operation Display	S8	ro	–	Display of the selected operating mode

See chap. 5, p. 82.

Position Demand Value (0x6062)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6062	0x00	Position Demand Value	S32	ro	–	Position setpoint value in user-defined scaling

See chap. 4.4, p. 42.

Position Actual Internal Value (0x6063)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6063	0x00	Position Actual Internal Value	S32	ro	–	Position Actual Value in internal scaling

Position Actual Value (0x6064)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6064	0x00	Position Actual Value	S32	ro	–	Position Actual Value in user-defined scaling

See chap. 4.4, p. 42.

Following Error Window (0x6065)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6065	0x00	Following Error Window	U32	rw	32	Corridor for the control deviation of the position controller

See chap. 5, p. 82.

Following Error Time Out (0x6066)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6066	0x00	Following Error Time Out	U16	rw	48	Minimum time for which a following error must lie outside the defined corridor before the error is reported

See chap. 5, p. 82.

Position Window (0x6067)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6067	0x00	Position Window	U32	rw	32	Corridor around the setpoint value position in user-defined scaling

See chap. 5, p. 82.

Position Window Time (0x6068)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6068	0x00	Position Window Time	U16	rw	48	Minimum residence time within the corridor in PP operating mode, until the setpoint value position is reported as achieved.

See chap. 5, p. 82.

Velocity Demand Value (0x606B)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606B	0x00	Velocity Demand Value	S32	ro	–	Setpoint value of the velocity in user-defined scaling

See chap. 4.4, p. 42.

Velocity Actual Value (0x606C)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606C	0x00	Velocity Actual Value	S32	ro	–	Velocity in user-defined scaling

See chap. 4.4, p. 42.

Velocity Window (0x606D)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606D	0x00	Velocity Window	U16	rw	32	Corridor around the setpoint value velocity [in user-defined scaling]

See chap. 4.3.4, p. 30.

Velocity Window Time (0x606E)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606E	0x00	Velocity Window Time	U16	rw	48	Minimum residence time within the corridor in ms

See chap. 4.3.4, p. 30.

Velocity Threshold (0x606F)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x606F	0x00	Velocity Threshold	U16	rw	32	Corridor at n = 0

See chap. 4.3.4, p. 30.

Velocity Threshold Time (0x6070)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6070	0x00	Velocity Threshold Time	U16	rw	48	Monitoring time [ms]. If the velocity lies outside the corridor for longer than is listed here, the velocity is reported as not equal to 0.

See chap. 4.3.4, p. 30.

Target Torque (0x6071)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6071	0x00	Target Torque	S16	rw	0	Setpoint of the torque in relative scaling [a)]

a) 1000 = motor rated torque

See chap. 5.7, p. 119.

Maximum Torque (0x6072)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6072	0x00	Maximum Torque	U16	rw	6000	Torque limitation in relative scaling [a)]

a) 1000 = motor rated torque

See chap. 5, p. 82.

Torque Demand (0x6074)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6074	0x00	Torque Demand	S16	ro	0	Setpoint of the torque (value from the movement profile) in relative scaling [a)]

a) 1000 = motor rated torque

See chap. 4.4, p. 42.

Torque Actual Value (0x6077)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6077	0x00	Torque Actual Value	S16	ro	0	Actual value of the torque in relative scaling [a]

a) *1000 = motor rated torque*

See chap. 4.4, p. 42.

Current Actual Value (0x6078)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6078	0x00	Current Actual Value	S16	ro	0	Actual value of the current in relative scaling [a]

a) *1000 = motor rated torque*

Target Position (0x607A)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x607A	0x00	Target Position	S32	rw	0	Position setpoint value in user-defined scaling

See chap. 4.4, p. 42.

Position Range Limit (0x607B)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x607B	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Min Position Range Limit	S32	rw	-2147483648	Lower limit of the position range in user-defined scaling
	0x02	Max Position Range Limit	S32	rw	2147483647	Upper limit of the position range in user-defined scaling

See chap. 5, p. 82.

Homing Offset (0x607C)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x607C	0x00	Min Range Limit	S32	rw	0	Offset of the zero position relative to the position of the reference switch

See chap. 5.4, p. 103.

Software Position Limit (0x607D)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x607D	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Min Position Limit	S32	rw	-2147483648	Lower limit of the position range
	0x02	Max Position Limit	S32	rw	2147483647	Upper limit of the position range

See chap. 5, p. 82.

Polarity (0x607E)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x607E	0x00	Polarity	U8	rw	0	Bit-coded

See chap. 4.7.6, p. 71.

Maximum Motor Speed (0x6080)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6080	0x00	Maximum Motor Speed	U32	rw	32767	Maximum velocity of the motor in user-defined scaling

See chap. 4.3.4, p. 30.

Profile Velocity (0x6081)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6081	0x00	Profile Velocity	U32	rw	32767	Profile velocity in user-defined scaling

See chap. 4.4, p. 42.

Profile Acceleration (0x6083)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6083	0x00	Profile acceleration	U32	rw	32767	Acceleration [$1/s^2$]

See chap. 4.4, p. 42.

Profile Deceleration (0x6084)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6084	0x00	Profile deceleration	U32	rw	32767	Braking rate [$1/s^2$]

See chap. 4.4, p. 42.

Quick Stop Deceleration (0x6085)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6085	0x00	Quick Stop Deceleration	U32	rw	32767	Quick-Stop deceleration [$1/s^2$]

See chap. 7.2.1, p. 143.

Motion Profile Type (0x6086)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6086	0x00	Motion Profile Type	S16	rw	0	Motion profile type: 0: Linear profile 1: Sin^2 velocity

See chap. 4.4, p. 42.

Position Encoder Resolution (0x608F)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x608F	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Encoder Increments	U32	ro	4096	Encoder increments
	0x02	Motor Revolutions	U32	ro	1	Motor revolutions

See chap. 4.7.1, p. 68.

Velocity Encoder Resolution (0x6090)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6090	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Encoder Increments	U8	ro	4096	Position resolution of the sensor that is configured
	0x02	Motor Revolutions	U8	ro	1	Number of motor revolutions for the impulse number specified in subindex 1

See chap. 4.7.2, p. 69.

Gear Ratio (0x6091)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6091	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Motor Shaft Revolutions	U32	ro	1	Revolutions of the gearbox input shaft
	0x02	Driving Shaft Revolutions	U32	rw	1	Revolutions of the gearbox driving shaft

See chap. 4.7.4, p. 70.

Feed Constant (0x6092)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6092	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Feed	U32	rw	4096	Feed
	0x02	Shaft Revolutions	U32	rw	1	Revolutions

See chap. 4.7.5, p. 71.

Velocity Factor (0x6096)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6096	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Numerator	U32	rw	1	Denominator
	0x02	Denominator	U32	rw	4096	Numerator

See chap. 4.7.3, p. 69.

Homing Method (0x6098)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6098	0x00	Homing Method	S8	rw	0	Homing method

See chap. 5.4, p. 103.

Homing Speeds (0x6099)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6099	0x00	Number of entries	U8	ro	2	Number of object entries
	0x01	Switch Seek Velocity	U32	rw	400	Switch seek velocity
	0x02	Homing Speed	U32	rw	400	Homing velocity

See chap. 5.4, p. 103.

Homing Acceleration (0x609A)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x609A	0x00	Homing Accelera-tion	U32	rw	50	Acceleration during homing

See chap. 5.4, p. 103.

Position Offset (0x60B0)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60B0	0x00	Position Offset	S32	rw	0	Position Offset

See chap. 5.5, p. 112.

Velocity Offset (0x60B1)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60B1	0x00	Velocity Offset	S32	rw	0	Velocity Offset

See chap. 5.5, p. 112, and chap. 5.6, p. 116.

Torque Offset (0x60B2)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60B2	0x00	Torque Offset	S16	rw	0	Torque Offset

See chap. 5.5, p. 112, chap. 5.6, p. 116 and chap. 5.7, p. 119.

Touch Probe Function (0x60B8)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60B8	0x00	Touch Probe Func-tion	U16	rw	0	Touch probe function

See chap. 4.6.5, p. 65.

Touch Probe Status (0x60B9)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60B9	0x00	Touch Probe Status	U16	ro	0	Touch Probe Status

See chap. 4.6.5, p. 65.

Touch Probe 1 Positive Edge (0x60BA)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60BA	0x00	Touch Probe 1 Pos-itive Edge	S32	ro	0	Touch probe position 1 positive value

See chap. 4.6.5, p. 65.

Touch Probe 1 Negative Edge (0x60BB)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60BB	0x00	Touch Probe 1 Negative Edge	S32	ro	0	Touch probe position 1 negative value

See chap. 4.6.5, p. 65.

Touch Probe 2 Positive Edge (0x60BC)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60BC	0x00	Touch Probe 2 Positive Edge	S32	ro	0	Touch probe position 2 positive value

See chap. 4.6.5, p. 65.

Touch Probe 2 Negative Edge (0x60BD)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60BD	0x00	Touch Probe 2 Negative Edge	S32	ro	0	Touch probe position 2 negative value

See chap. 4.6.5, p. 65.

Maximum Acceleration (0x60C5)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60C5	0x00	Maximum Acceleration	U32	rw	30000	PP mode maximum acceleration [$1/s^2$]

See chap. 4.6.5, p. 65.

Maximum Deceleration (0x60C6)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60C6	0x00	Maximum Deceleration	U32	rw	30000	PP mode maximum braking rate [$1/s^2$]

See chap. 4.6.5, p. 65.

Touch Probe 1 Positive Edge Counter (0x60D5)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60D5	0x00	Touch Probe 1 Positive Edge Counter	U16	ro	–	Counter of the positive flanks at input 1

See chap. 4.6.5, p. 65.

Touch Probe 1 Negative Edge Counter (0x60D6)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60D6	0x00	Touch Probe 1 Negative Edge Counter	U16	ro	–	Position of the negative flanks at input 1

See chap. 4.6.5, p. 65.

Touch Probe 2 Positive Edge Counter (0x60D7)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60D7	0x00	Touch Probe 2 Positive Edge Counter	U16	ro	–	Counter of the positive flanks at input 2

See chap. 4.6.5, p. 65.

Touch Probe 2 Negative Edge Counter (0x60D8)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60D8	0x00	Touch Probe 2 Negative Edge Counter	U16	ro	–	Position of the negative flanks at input 2

See chap. 4.6.5, p. 65.

Positive Torque Limit Value (0x60E0)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60E0	0x00	Positive torque limit value	U16	rw	6000	Upper limit value in relative scaling [a)]

a) 1000 = motor rated torque

See chap. 6.2, p. 134.

Negative Torque Limit Value (0x60E1)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60E1	0x00	Negative torque limit value	U16	rw	6000	Lower limit value in relative scaling [a)]

a) 1000 = motor rated torque

See chap. 6.2, p. 134.

Following Error Actual Value (0x60F4)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60F4	0x00	Following Error Actual Value	U32	ro	0	Deviation between position setpoint value and position actual value

See chap. 5.5, p. 112, and chap. 5.9, p. 125.

Target Velocity (0x60FF)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x60FF	0x00	Target Velocity	S32	rw	0	Velocity setpoint value

See chap. 4.7.1, p. 68.

Motor Catalogue Number (0x6403)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6403	0x00	Motor Catalogue Number	Vis string	rw	0	Motor number

Supported Drive Modes (0x6502)

Index	Subindex	Name	Type	Attr.	Default value	Meaning
0x6502	0x00	Supported Drive Modes	U32	ro	Bit-coded see CiA 402	Supported operating modes

7000.05048, 2nd edition, 9-12-2016

DR. FRITZ FAULHABER GMBH & CO. KG
Antriebssysteme

Daimlerstraße 23 / 25
71101 Schönaich • Germany
Tel. +49(0)7031/638-0
Fax +49(0)7031/638-100
info@faulhaber.de
www.faulhaber.com