FAST Application Software

Technology module

Basic Motion _____

Reference Manual EN

13531708

Lenze

Contents

1 About this documentation

This documentation ...

- contains detailed information on the functionalities of the "Basic Motion" technology module;
- is part of the "Controller-based Automation" manual collection. It consists of the following sets of documentation:

Documentation type	Subject
Product catalogue	Controller-based Automation (system overview, sample topologies) Lenze Controller (product information, technical data)
System manuals	Visualisation (system overview/sample topologies)
Communication manuals **Online helps**	Bus systems • Controller-based Automation EtherCAT® • Controller-based Automation CANopen® • Controller-based Automation PROFIBUS® • Controller-based Automation PROFINET®
Reference manuals **Online helps**	Lenze Controllers: • Controller 3200 C • Controller c300 • Controller p300 • Controller p500
Software manuals **Online helps**	Lenze Engineering Tools: • »PLC Designer« (programming) • »Engineer« (parameter setting, configuration, diagnostics) • »VisiWinNET® Smart« (visualisation) • »Backup & Restore« (data backup, recovery, update)

More technical documentation for Lenze components

Further information on Lenze products which can be used in conjunction with Controller-based Automation can be found in the following sets of documentation:

Planning / configuration / technical data	
☐	**Product catalogues** • Controller-based Automation • Controllers • Inverter Drives/Servo Drives
Mounting and wiring	
🕮	**Mounting instructions** • Controllers • Communication cards (MC-xxx) • I/O system 1000 (EPM-Sxxx) • Inverter Drives/Servo Drives • Communication modules
☐	**Hardware manuals** • Inverter Drives/Servo Drives
Parameter setting / configuration / commissioning	
☐	**Online help/reference manuals** • Controllers • Inverter Drives/Servo Drives • I/O system 1000 (EPM-Sxxx)
☐	**Online help/communication manuals** • Bus systems • Communication modules
Sample applications and templates	
☐	**Online help / software and reference manuals** • i700 application sample • Application Samples 8400/9400 • FAST Application Template Lenze/PackML • FAST technology modules

Symbols:

- 🕮 Printed documentation
- ☐ PDF file / online help in the Lenze engineering tool

Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.lenze.com

Target group

This documentation is intended for all persons who plan, program and commission a Lenze automation system on the basis of the Lenze FAST Application Software.

1.1 Document history

Version			Description
2.2	05/2017	TD17	• Content structure has been changed. • General revisions
2.1	04/2016	TD17	General revisions
2.0	10/2015	TD17	• Corrections and additions • Content structure has been changed.
1.0	05/2015	TD17	First edition

1.2 Conventions used

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

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Text		
Program name	» «	»PLC Designer« ...
Variable names	*italics*	By setting *bEnable* to TRUE...
Function blocks	**bold**	The **L_MC1P_AxisBasicControl** function block ...
Function libraries		The **L_TT1P_TechnologyModules** function library ...

Source code	Font
	"Courier new"
		`...`
		`dwNumerator := 1;`
		`dwDenominator := 1;`
		`...`

Icons		
Page reference	(📖 6)	Reference to further information: Page number in PDF file.

Variable names

The conventions used by Lenze for the variable names of Lenze system blocks, function blocks, and functions are based on the "Hungarian Notation". This notation makes it possible to identify the most important properties (e.g. the data type) of the corresponding variable by means of its name, e.g. xAxisEnabled.

1.3 Definition of the notes used

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

Safety instructions

Layout of the safety instructions:

Pictograph and signal word!

(characterise the type and severity of danger)

Note

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

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

Application notes

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

2 Safety instructions

Please observe the safety instructions in this documentation when you want to commission an automation system or a plant with a Lenze Controller.

The device documentation contains safety instructions which must be observed!

Read the documentation supplied with the components of the automation system carefully before you start commissioning the Controller and the connected devices.

Danger!

High electrical voltage

Injury to persons caused by dangerous electrical voltage

Possible consequences

Death or severe injuries

Protective measures

Switch off the voltage supply before working on the components of the automation system.

After switching off the voltage supply, do not touch live device parts and power terminals immediately because capacitors may be charged.

Observe the corresponding information plates on the device.

Danger!

Injury to persons

Risk of injury is caused by ...

- unpredictable motor movements (e.g. unintended direction of rotation, too high velocities or jerky movement);
- impermissible operating states during the parameterisation while there is an active online connection to the device.

Possible consequences

Death or severe injuries

Protective measures

- If required, provide systems with installed inverters with additional monitoring and protective devices according to the safety regulations valid in each case (e.g. law on technical equipment, regulations for the prevention of accidents).
- During commissioning, maintain an adequate safety distance to the motor or the machine parts driven by the motor.

Stop!

Damage or destruction of machine parts

Damage or destruction of machine parts can be caused by ...

- Short circuit or static discharges (ESD);
- unpredictable motor movements (e.g. unintended direction of rotation, too high velocities or jerky movement);
- impermissible operating states during the parameterisation while there is an active online connection to the device.

Protective measures

- Always switch off the voltage supply before working on the components of the automation system.
- Do not touch electronic components and contacts unless ESD measures were taken beforehand.
- If required, provide systems with installed inverters with additional monitoring and protective devices according to the safety regulations valid in each case (e.g. law on technical equipment, regulations for the prevention of accidents).

3 Functional description for "Basic Motion"

The technology module provides basic functions for the establishment or continuous movement of an axis.

3.1 Overview of the functions

In addition to the basic functions for operating the **L_MC1P_AxisBasicControl** function block, the **Stop function** and the **Holding function**, the technology module offers the following functionalities:

- Manual jog (jogging) (📖 20)
- Homing (📖 21)
- Absolute positioning (📖 22)
- Relative positioning (📖 23)
- Continuous travel with a defined speed (📖 24)

3.2 Important notes on how to operate the technology module

Controlled start of the axes

Motion commands that are set in the inhibited axis state (*xAxisEnabled* = FALSE) after enable (*xRegulatorOn* = TRUE) must be activated again by a FALSE↗TRUE edge.

In this way it is prevented that the drive starts in an uncontrolled manner after controller enable.

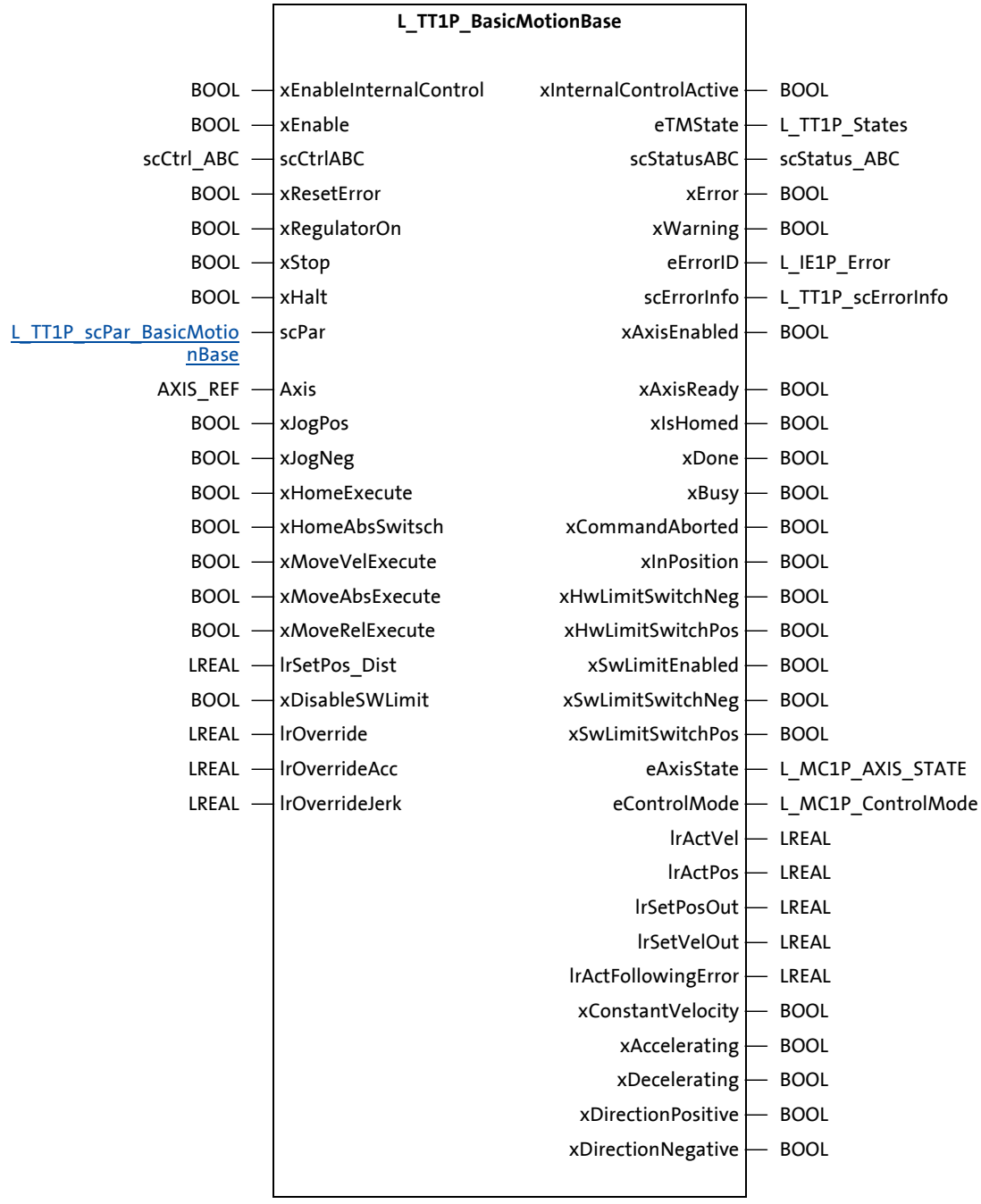
Example Manual jog (jogging) (📖 20)**:**

1. In the inhibited axis state (*xAxisEnabled* = FALSE), *xJogPos* is set to TRUE.
   - *xRegulatorOn* = FALSE (axis is inhibited.)
     ==> "READY" state (*xAxisEnabled* = FALSE)
   - *xJogPos* = TRUE (manual jog is to be executed.)
2. Enable axis.
   - *xRegulatorOn* = TRUE
     ==> "READY" state (*xAxisEnabled* = TRUE)
3. Execute manual jog.
   - *xJogPos* = FALSE↗TRUE
     ==> "JOGPOS" state

3.3 L_TT1P_BasicMotionBase function block

The figure shows the inputs and outputs of the function block.

The base version offers the full functionality of the technology module.



3.3.1 Inputs and outputs

Designator Data type	Description
Axis AXIS_REF	Reference to the axis

3.3.2 Inputs

Designator Data type	Description	
xEnableInternalControl BOOL	TRUE	In the visualisation, the internal control of the axis can be selected via the "Internal Control" axis.
xEnable BOOL	Execution of the function block	
	TRUE	The function block is executed.
	FALSE	The function block is not executed.
scCtrlABC scCTRL_ABC	Input structure for the **L_MC1P_AxisBasicControl** function block • scCtrlABC can be used in "Ready" state. • If there is a request, the state changes to "Service". • The state change from "Service" back to "Ready" takes place if there are no more requests.	
xResetError BOOL	TRUE	Reset axis error or software error.
xRegulatorOn BOOL	TRUE	Activate controller enable of the axis (via the **MC_Power** function block).
xStop BOOL	TRUE	Cancel the active movement and brake the axis to a standstill with the deceleration defined in the lrStopDec parameter. • The state changes to "Stop". • The technology module remains in the "Stop" state as long as xStop = TRUE. • The input is also active with "Internal Control".
xHalt BOOL	TRUE	Cancel the active movement and brake the axis to a standstill with the deceleration defined in the lrHaltDec parameter. • The state changes to "Halt". The function can be deactivated via the following Execute inputs (xHalt = FALSE): • xHomeExecute = TRUE • xMoveVelExecute = TRUE • xMoveAbsExecute = TRUE • xMoveRelExecute = TRUE A corresponding state change takes place (see State machine (📖 19)).
scPar L_TT1P_scPar_BasicMotionBase	The parameter structure contains the parameters of the technology module. The data type depends on the version used (Base/State/High).	
xJogPos BOOL	TRUE	Traverse axis in positive direction (manual jog). If xJogNeg is also TRUE, the traversing direction selected first remains set.
xJogNeg BOOL	TRUE	Traverse axis in negative direction (manual jog). If xJogPos is also TRUE, the traversing direction selected first remains set.
xHomeExecute BOOL	The input is edge-controlled and evaluates the rising edge.	
	FALSE↗ TRUE	Start homing. The function is aborted via the xStop input.

Designator Data type	Description	
xHomeAbsSwitch BOOL	TRUE	Connection for reference switch: For homing modes with a reference switch, connect this input to the digital signal which maps the state of the reference switch.
xMoveVelExecute BOOL	The input is edge-controlled and evaluates the rising edge.	
	FALSE↗ TRUE	Start continuous travel with the speed defined. (Abort of the function via the xStop or xHalt input.)
xMoveAbsExecute BOOL	The input is edge-controlled and evaluates the rising edge.	
	FALSE↗ TRUE	Activate absolute positioning. (Abort of the function via the xStop or xHalt input.)
xMoveRelExecute BOOL	The input is edge-controlled and evaluates the rising edge.	
	FALSE↗ TRUE	Activate relative positioning. (Abort of the function via the xStop or xHalt input.)
lrSetPos_Dist LREAL	Absolute positioning: target position • The absolute position describes the distance from the zero position to the target position. • Absolute position = target position • Unit: units Relative positioning: distance to be travelled • The relative position takes the current actual position at the starting time of the travel command into consideration. • Relative position = target position - actual position • Unit: units	
xDisableSWLimit BOOL	TRUE	Deactivate software limit positions.
	FALSE	Activate software limit positions.
lrOverride LREAL	Factor for the global speed override • Value range: 0.1 ... 1.0 (10 ... 100 %) • Initial value: 1.0 (100 %)	
lrOverrideAcc LREAL	Factor for the global acceleration override • Value range: 0.1 ... 1.0 (10 ... 100 %) • Initial value: 1.0 (100 %)	
lrOverrideJerk LREAL	Factor for the global jerk override • Value range: 0.1 ... 1.0 (10 ... 100 %) • Initial value: 1.0 (100 %)	

3.3.3 Outputs

Designator Data type	Description	
xInternalControlActive BOOL	TRUE	The internal control of the axis is activated via the visualisation. (xEnableInternalControl input = TRUE)
eTMState L_TT1P_States	Current state of the technology module ▸ State machine (📖 19)	
scStatusABC scStatus_ABC	Structure of the status data of the **L_MC1P_AxisBasicControl** function block	
xError BOOL	TRUE	There is an error in the technology module.
xWarning BOOL	TRUE	There is a warning in the technology module.
eErrorID L_IE1P_Error	ID of the error or warning message if xError = TRUE or xWarning = TRUE. **"FAST technology modules" reference manual:** Here you can find information on error or warning messages.	
scErrorInfo L_TT1P_scErrorInfo	Error information structure for a more detailed analysis of the error cause	
xAxisEnabled BOOL	TRUE	The axis is enabled.
xAxisReady BOOL	TRUE	The axis is ready for operation.
xIsHomed BOOL	TRUE	The axis has been referenced (reference known).
xDone BOOL	TRUE	The request/action has been completed successfully.
xBusy BOOL	TRUE	The request/action is currently being executed.
xCommandAborted BOOL	TRUE	The function activated has been aborted by another function block.
xInPosition BOOL	TRUE	The target position has been reached.
xHwLimitSwitchNeg BOOL	TRUE	The negative hardware limit switch has been reached or approached. • To control the HW limit switches, the **L_MC1P_HWLimitSwitchInterface** function block must be used. This function block is an interface to the connection of the HW limit switches of an axis. • The drive is braked to a standstill with the deceleration set in the lrStopDec parameter. • If the HW limit switch has been overtravelled, a change to the "ERROR" state with the '20501' error message (HWLimitNeg) takes place.
xHwLimitSwitchPos BOOL	TRUE	The positive hardware limit switch has been reached or approached. • To control the HW limit switches, the **L_MC1P_HWLimitSwitchInterface** function block must be used. This function block is an interface to the connection of the HW limit switches of an axis. • The drive is braked to a standstill with the deceleration set in the lrStopDec parameter. • If the HW limit switch has been overtravelled, a change to the "ERROR" state with the '20500' error message (HWLimitPos) takes place.
xSwLimitEnabled BOOL	TRUE	Activate the monitoring of the software limit positions.
xSwLimitSwitchNeg BOOL	TRUE	The negative software limit position has been reached or approached. • If the SW limit switch has been overtravelled, a change to the "ERROR" state with the '20307' error message (SWLimitNeg) takes place.

Designator Data type	Description	
xSwLimitSwitchPos BOOL	TRUE	The positive software limit position has been reached or approached. • If the SW limit switch has been overtravelled, a change to the "ERROR" state with the '20306' error message (SWLimitPos) takes place.
eAxisState L_MC1P_AXIS_STATE	Current PLCopen state of the axis	
	0	Init
	1	ErrorStop
	2	Disabled
	3	StandStill
	4	Stopping
	5	DiscMotion
	6	SyncMotion
	7	ContMotion
	8	Homing
eControlMode L_MC1P_ControlMode	Current control mode of the axis	
	0	PosCtrlDrive
	1	PosCtrlPLC
	2	SpeedCtrl
	3	TorqueCtrl
lrActVel LREAL	Current velocity • Unit: units/s	
lrActPos LREAL	Current position • Unit: units	
lrSetPosOut LREAL	Setpoint position • Unit: units	
lrSetVelOut LREAL	Setpoint speed • Unit: units/s	
lrActFollowingError LREAL	Current following error • Unit: units	
xConstantVelocity BOOL	TRUE	The speed defined in the lrVel parameter has been reached for the first time.
xAccelerating BOOL	TRUE	The axis is accelerated. (Increase in the speed value.)
xDecelerating BOOL	TRUE	The axis is decelerated/braked. (Decrease in the speed value.)
xDirectionPositive BOOL	TRUE	The axis travels in positive direction.
xDirectionNegative BOOL	TRUE	The axis travels in negative direction.

3.3.4 Parameters

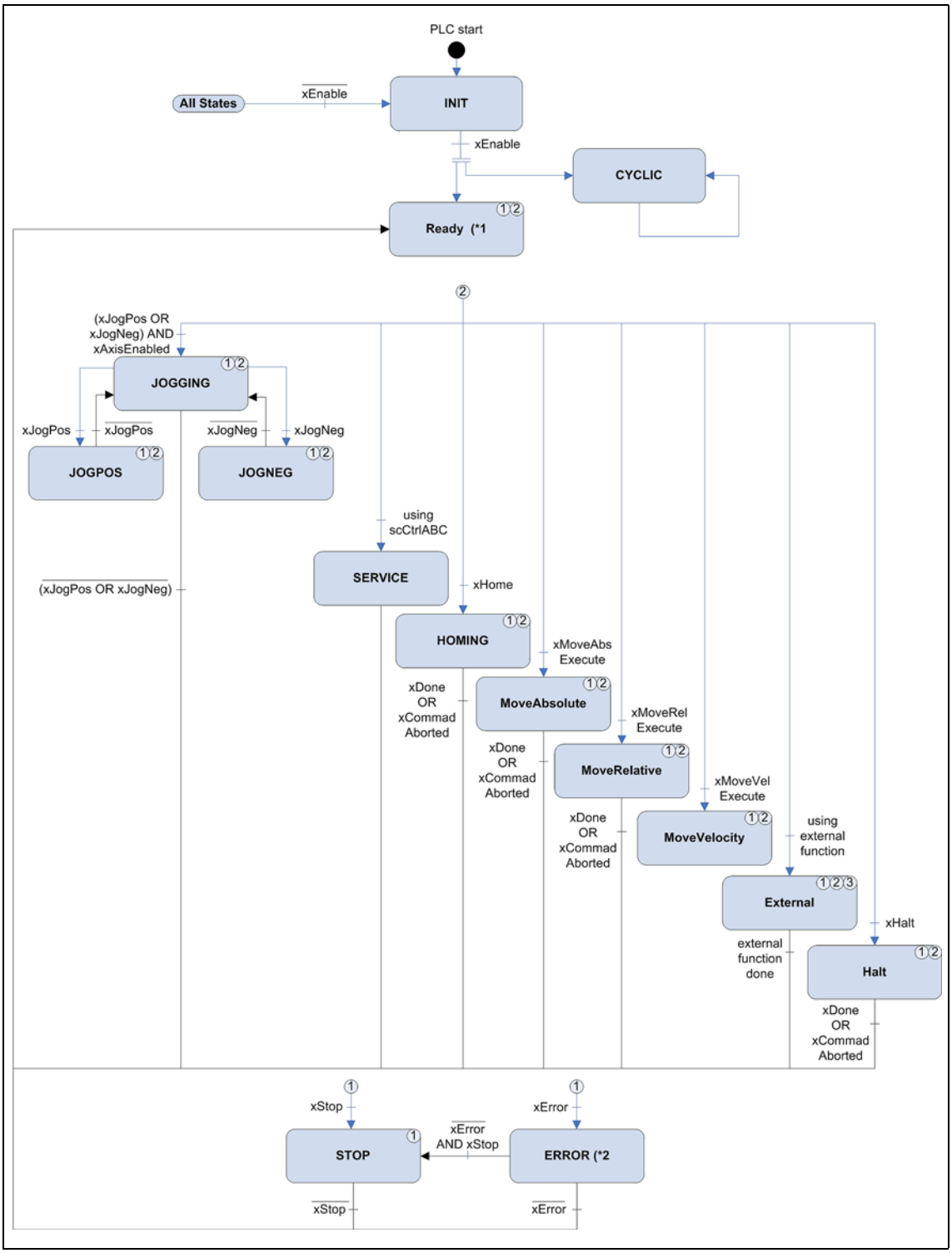
L_TT1P_scPar_BasicMotionBase

The **L_TT1P_scPar_BasicMotionBase** structure contains the parameters of the technology module.

Designator Data type	Description	
lrStopDec LREAL	Deceleration for the stop function and when hardware/software limit switches and the following error monitoring function are triggered • Unit: units/s^2 • Initial value: 10000	
lrStopJerk LREAL	Jerk for the stop function and for the triggering of the hardware limit switches, software limit positions, and the following error monitoring function • Unit: units/s^3 • Initial value: 100000	
lrHaltDec LREAL	Deceleration for the holding function Specification of the maximum speed variation which is to be used for deceleration to standstill. • Unit: units/s^2 • Initial value: 3600 • Only positive values are permissible.	
lrJerk LREAL	Jerk for compensating an offset value, trimming, clutch, or holding function • Unit: units/s^3 • Initial value: 100000	
lrJogJerk LREAL	Jerk for manual jog • Unit: units/s^3 • Initial value: 10000	
lrJogVel LREAL	Maximum speed to be used for manual jog. • Unit: units/s • Initial value: 10	
lrJogAcc LREAL	Acceleration for manual jog Specification of the maximum speed variation which is to be used for acceleration. • Unit: units/s^2 • Initial value: 100	
lrJogDec LREAL	Deceleration for manual jog Specification of the maximum speed variation which is to be used for deceleration to standstill. • Unit: units/s^2 • Initial value: 100	
lrHomePosition LREAL	Home position for a reference run (homing) • Unit: units • Initial value: 0	
xUseHomeExtParameter BOOL	Selection of the homing parameters to be used • Initial value: FALSE	
	FALSE	The homing parameters defined in the axis data are used.
	TRUE	The **scHomeExtParameter** homing parameters from the application are used.
scHomeExtParameter L_MC1P_HomeParameter	Homing parameters from the application • Only relevant if xUseHomeExtParameter = TRUE.	
scHomeExtTP MC_TRIGGER_REF	Transfer of an external touch probe event • Only relevant for "External source" touch probe configuration. • For describing the MC_TRIGGER_REF structure, see the **MC_TouchProbe** function block.	
lrVel LREAL	Velocity Selection of the maximum speed at which continuous travel is to be executed. • Unit: units/s • Initial value: 50	

Designator Data type		Description
lrAcc LREAL		Acceleration Specification of the maximum speed variation which is to be used for acceleration. • Unit: units/s^2 • Initial value: 100
lrDec LREAL		Deceleration Selection of the maximum speed variation which is to be used for deceleration to standstill. • Unit: units/s^2 • Initial value: 100
eDirection MC_DIRECTION		Traversing direction: • mcCurrentDirection: retain current direction (only for Modulo axis) • mcPositiveDirection: positive direction • mcNegativeDirection: negative direction • mcShortestWay: shortest path (only for Modulo axis) • Initial value: 'mcPositiveDirection'
xContinuousUpdate BOOL	TRUE	Check and acceptance of parameters • The parameters lrVel, lrAcc, lrDec and lrJerk are continuously checked every 5th clock cycle with regard to changes and accepted. • The lrSetPos_Dist input is cyclically checked for changes and accepted. • Initial value: FALSE (no check and acceptance) **Note:** In addition to the xExecute control input, some motion commands have a xContinuousUpdate control input. As long as this input is set to TRUE (while xBusy = TRUE), the profile parameters are accepted continuously. Thus, the motion profile can be changed during the processing phase. In contrast to a renewed triggering of the Execute input, this update method only influences the current motion. The status outputs and the axis state remain unchanged.

3.4 State machine



[3-1] State machine of the technology module

(*1 In the "Ready" state, xRegulatorOn has to be set to TRUE.

(*2 In the "ERROR" state, xResetError has to be set to TRUE in order to acknowledge and reset the errors.

(*3 The "External" state is set if a function outside the technology module generates setpoints for the connected axis (e.g. MC_CamIn, MC_MoveAbsolute etc.).

3.5 Manual jog (jogging)

Precondition

- The technology module is in the "Ready" state.
- The slave axis is enabled (*xRegulatorOn* = TRUE).

Execution

For manual jog of the axis, the manual jog speed *lrJogVel* is used.

If the *xJogPos* input is TRUE, the axis is traversed in positive direction and if the *xJogNeg* input is TRUE, the axis is traversed in negative direction. The axis is executed for as long as the input remains set to TRUE.

The current travel command cannot be replaced by another jog command. Only if both inputs have been reset, the State machine (📖 19) changes to the "Ready" state again.

Parameters to be set

The parameters for the manual jog are located in the L_TT1P_scPar_BasicMotionBase (📖 17) parameter structure.

```
lrJogVel : LREAL := 10;      // Velocity [units/s]
lrJogAcc : LREAL := 100;     // Acceleration [units/s^2]
lrJogDec : LREAL := 100;     // Deceleration [units/s^2]
lrJogJerk : LREAL := 10000;  // Jerk [units/s^3]
```

The parameter values can be changed during operation. They are accepted when the *xJogPos* or *xJogNeg* input is set to TRUE again.

3.6 Homing

Precondition

- The technology module is in the "Ready" state.
- The slave axis is enabled (*xRegulatorOn* = TRUE).

Execution

Homing is started with a rising edge (FALSE↗TRUE) at the *xHomeExecute* input. The axis will be travelling until the home position is reached. After successful homing, the State machine (📖 19) changes back again to the "Ready" state.

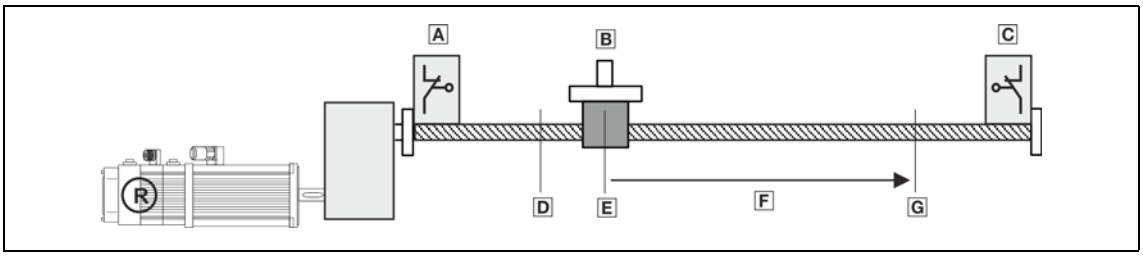
The homing process is not interrupted if the *xHomeExecute* input is set to FALSE too early. The function is aborted via the *xStop* input.

Parameters to be set

The parameters for homing are located in the L_TT1P_scPar_BasicMotionBase (📖 17) parameter structure.

```
xUseHomeExtParameter : BOOL := FALSE;
lrHomePosition : LREAL := 0.0;
scHomeExtParameter : L_MC1P_HomeParameter;
scHomeExtTP : MC_TRIGGER_REF;
```

3.7 Absolute positioning



[3-2] Example: Absolute positioning of a linear axis

Pos.	Description
A	Negative hardware limit switch
B	Load (e.g. slide)
C	Positive hardware limit switch
D	Zero position of the measuring system
E	Current actual position
F	Distance to be travelled
G	Target position (lrSetPos_Dist input)

With absolute positioning, the target position in the *lrSetPos_Dist* input (G) is defined as a unique absolute position, relating to the zero position of the measuring system (D). The distance to be travelled (F) results from the distance between the current actual position (E) and the absolute target position (G).

Precondition

- The technology module is in the "Ready" state.
- The slave axis is enabled (*xRegulatorOn* = TRUE).

Execution

With a rising edge (FALSE↗TRUE) at the *xMoveAbsExecute* input, a travel to the absolute target position in the *lrSetPos_Dist* input is started on the basis of the parameters for the motion profile (see below). The technology module changes to the "MoveAbsolute" state.

When the target position has been reached, ...

- the axis is brought to a standstill via the *lrDec* parameter.
- a state change back to the "Ready" state takes place.

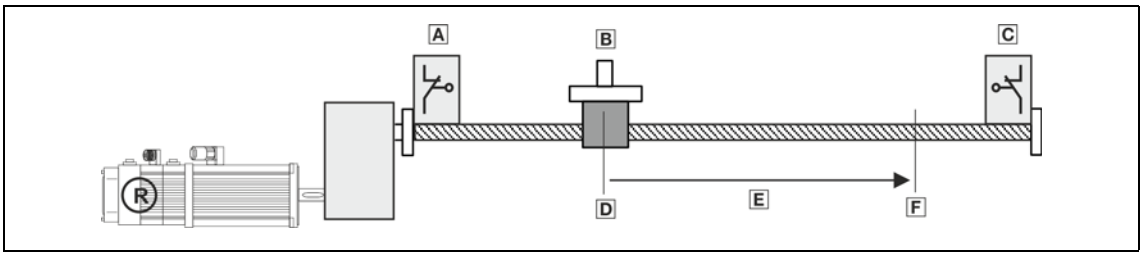
During the travel process, the absolute target position in the *lrSetPos* input can be changed, which requires triggering the Execute input for positioning again.

Parameters to be set

The parameters for continuous travel are located in the L_TT1P_scPar_BasicMotionBase (📖 17) parameter structure.

```
lrVel : LREAL := 50;          // units/s
lrAcc : LREAL := 100;         // units/s2
lrDec : LREAL := 100;         // units/s2
lrJerk : LREAL := 100000;     // units/s3
eDirection : MC_DIRECTION;  // mcCurrentDirection, mcPositiveDirection,
                            // mcNegativeDirection, mcShortestWay
```

3.8 Relative positioning



[3-3] Example: Relative positioning of a linear axis

Pos.	Description
A	Negative hardware limit switch
B	Load (e.g. slide)
C	Positive hardware limit switch
D	Current actual position
E	Distance to be travelled (= lrSetPos_Dist input)
F	Target position (lrSetPos_Dist input)

In the case of the relative positioning, the target position defined in the *lrSetPos_Dist* input corresponds to the distance to be travelled (F = E), meaning that the drive exactly travels the distance defined from the current actual position (D).

Precondition

- The technology module is in the "Ready" state.
- The slave axis is enabled (*xRegulatorOn* = TRUE).

Execution

With a rising edge (FALSE↗TRUE) at the *xMoveRelExecute* input, a travel to the relative target position (*lrSetPos_Dist* input) is started on the basis of the parameters for the motion profile (see below). The technology module changes to the "MoveRelative" state.

When the target position has been reached, ...

- the axis is brought to a standstill via the *lrDec* parameter.
- a state change back to the "Ready" state takes place.

During the travel process, the absolute target position in the *lrSetPos* input can be changed, which requires triggering the Execute input for positioning again.

Parameters to be set

The parameters for continuous travel are located in the L_TT1P_scPar_BasicMotionBase (📖 17) parameter structure.

```
lrVel : LREAL := 50;          // units/s
lrAcc : LREAL := 100;         // units/s2
lrDec : LREAL := 100;         // units/s2
lrJerk : LREAL := 100000;     // units/s3
eDirection : MC_DIRECTION;  // mcCurrentDirection, mcPositiveDirection,
                            // mcNegativeDirection, mcShortestWay
```

3.9 Continuous travel with a defined speed

Precondition

- The technology module is in the "Ready" state.
- The slave axis is enabled (*xRegulatorOn* = TRUE).

Execution

With a rising edge (FALSE↗TRUE) at the *xMoveVelExecute* input, the continuous travel is started on the basis of the parameters for the motion profile (see below). The technology module changes to the "MoveVelocity" state.

The axis continues travelling until a software limit position has been reached (*xDisableSWLimit* input = FALSE, *xSwLimitEnabled*, *xSwLimitSwitchPos/Neg* outputs = TRUE) or a hardware limit switch has been approached (*xHwLimitSwitchPos/Neg* output = TRUE).

- If a software limit position has been reached, the axis is brought to a standstill via the *lrDec* parameter.
- If a hardware limit position has been reached, the axis is brought to a standstill via the *lrStopDec* parameter.

Then a state change back to the "Ready" state takes place.

Parameters to be set

The parameters for continuous travel are located in the L_TT1P_scPar_BasicMotionBase (📖 17) parameter structure.

```
lrStopDec : LREAL := 10000;  // units/s2
lrStopJerk : LREAL := 100000; // units/s3
lrVel : LREAL := 50;         // units/s
lrAcc : LREAL := 100;        // units/s2
lrDec : LREAL := 100;        // units/s2
lrJerk : LREAL := 100000;     // units/s3
eDirection : MC_DIRECTION;  // mcCurrentDirection, mcPositiveDirection,
                            // mcNegativeDirection, mcShortestWay
```

3.10 CPU utilisation (example Controller 3231 C)

The following table shows the CPU utilisation in microseconds using the example of the 3231 C controller (ATOM™ processor, 1.6 GHz).

Versions	Interconnection of the technology module	CPU utilisation	
		Average	Maximum peak
Base	Combination of functions: • xMoveVelExecute • xMoveAbsExecute • xMoveRelExecute • xStop • xHalt • xJogPos • xJogNeg	35	70

Index

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, 31763 Hameln
Hans-Lenze-Straße 1, 31855 Aerzen
GERMANY
HR Hannover B 205381
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, 32699 Extertal
GERMANY
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com