



Technology module

Electronic Gearbox _____

Reference manual

EN

Contents

1	About this document	3
1.1	Document history	5
1.2	Conventions used	6
1.3	Layout of the safety instructions	7
2	Safety instructions	8
3	Overview of the technology module functions	10
3.1	Functional description	10
3.2	Functional overview of the TM	11
3.3	Important notes on how to operate the technology module	12
4	Application ranges	13
5	L_TS2P_ElectronicGearboxBase function block	14
5.1	Inputs	15
5.2	Outputs	17
5.3	Parameter	19
6	State machine	22
7	Signal flow diagram	24
7.1	Structure of the signal flow	25
7.2	Structure of the access points	26
8	Description of the basic functions	27
8.1	Manual jog (jogging)	28
8.2	Homing	30
8.3	QSP application	33
8.4	Halt	34
8.5	Absolute positioning	35
8.6	Relative positioning	37
8.7	Continuous travel with a defined velocity	40
8.8	Aborting and completing a positioning process	43
8.9	Override for speed	45
9	Description of the technology functions	47
9.1	Speed synchronism clutching-in/declutching mechanism	47
9.1.1	Set gearbox factor	47
9.1.2	Invert gearbox factor	49
9.1.3	Clutching-in/declutching	50
9.2	Position trimming	52
9.2.1	Trimming per velocity offset	52
9.2.2	Trimming per position offset	54
9.3	Position offset via profile generator	57
9.4	Velocity offset via profile generator	59
9.5	Synchronism with basic speed	61
9.6	Limit setpoints to allowed travel directions	63
	Index	64
	Your opinion is important to us	65

1 About this document

This documentation ...

- contains detailed information on the functionalities of the "Electronic Gearbox Base" 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 controllers (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« (parameterisation, configuration, diagnostics) • »VisiWinNET® Smart« (visualisation) • »Backup & Restore« (backup, restore, update)

Further sets of technical documentation with regard to Lenze products

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

Design / 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
Parameterisation / 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 sets of 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 About this document

1.1 Document history

1.1 Document history

Version			Description
4.0	04/2021	TD06	Error corrections & supplements
3.0	06/2020	TD06	Error corrections & supplements New: Limit setpoints to allowed travel directions (📖 63)
2.0	12/2019	TD06	Error corrections & supplements
1.0	07/2019	TD29	First edition

1 About this document

1.2 Conventions used

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

Type of information	Writing	Examples/notes
Numeric notation		
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Text		
Program name	» «	»PLC Designer« ...
Variable identifiers	<i>Italics</i>	By setting <i>bEnable</i> 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; ...
Symbols		
Page reference	 6	Reference to further information: Page number in PDF file.

Variable identifiers

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 About this document

1.3 Layout of the safety instructions

DANGER!

Indicates an extremely hazardous situation. Failure to comply with this instruction will result in severe irreparable injury and even death.

WARNING!

Indicates an extremely hazardous situation. Failure to comply with this instruction may result in severe irreparable injury and even death.

CAUTION!

Indicates a hazardous situation. Failure to comply with this instruction may result in slight to medium injury.

NOTICE

Indicates a material hazard. Failure to comply with this instruction may result in material damage.

Application notes

Pictograph	Signal word	Meaning
	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.

NOTICE

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 Overview of the technology module functions

3.1 Functional description

3 Overview of the technology module functions

3.1 Functional description

With this technology module, an electrical shaft can be projected.

An electrical shaft uses control techniques to generate the synchronism of axes, consisting of a master axis (master value encoder) and slave axes, the speed and position of which are determined by the master value encoder.

Speed synchronism/relative phase synchronism

The current speed of the master axis coupling factor determines the speed of the slave axis; i.e. the speed of the slave drive is controlled so the slave and master axes are synchronous in a drift-free manner depending on a specified coupling factor. The slave axis can be coupled to the speed of the master axis without a step change if the slave axis is stationary or if the slave axis is rotating at a basic speed.

3.2 Functional overview of the TM

In addition to the functions for operating the function block L_MC1P_AxisBasicControlV2, the technology module offers the following functionalities:

Functionality	Description
Basic Motion Base (V2):	
Manual jog (jogging) (📖 28)	Traverse axis in manual operation
Homing (📖 30)	Reference the axis
OSP application (📖 33)	Stop motion via an application quick stop deceleration ramp (AxisRef)
Halt (📖 34)	Stop motion via a stop deceleration ramp
Relative positioning (📖 37)	Move axis to a target position relative to the current position
Continuous travel with a defined velocity (📖 40)	Execute continuous travel
Aborting and completing a positioning process (📖 43)	Stop positioning
Override for speed (📖 45)	Specify overrides for speed
Electronic Gearbox Base (V2):	
Speed synchronism clutching-in/declutching mechanism (📖 47) • Set gearbox factor (📖 47) • Clutching-in/declutching (📖 50)	Control drive speeds so the speed of rotation of the master and slave axes is synchronous depending on a specified coupling factor.
Trimming per velocity offset (📖 52)	Adjust the position of the slave axis in relation to the master axis by means of inch-operated velocity offset.
Trimming per position offset (📖 54)	Adjust the position of the slave axis in relation to the master axis by means of position difference.
Velocity offset via profile generator (📖 59)	Move slave axis with a variable velocity offset to the master axis.
Position offset via profile generator (📖 57)	Move slave axis with a variable position offset to the master axis.
Synchronism with basic speed (📖 61)	Starting from a basic speed, synchronize the speed of the slave axis to the speed of the master axis without a step change.
Limit setpoints to allowed travel directions (📖 63)	Block profile generation in the unauthorized direction.

Furthermore the technology module can be used for status and error messages and for resetting an error.



»PLC Designer« online help!

Here you will find detailed information about the function module L_MC1P_AxisBasicControlV2.

3.3

Important notes on how to operate the technology module

The technology module supports linear and rotary axes.

NOTICE

The master axis and the slave axis must either both be a linear axis or both be a rotary axis.

Start of the axes

The start of the axes after the controller has been inhibited depends on the requested movement type.

A distinction is made between the movement type with level-activated function and the movement type with edge-activated function.

- The movement type with level-activated function is automatically activated when the controller is enabled, as long as no higher priority function is active. This movement type includes
 - [Manual jog \(jogging\)](#) (28)
 - [OSP application](#) (33)
- The movement type with edge-activated function must be activated with a FALSE→TRUE edge after the controller has been enabled. This movement type includes
 - [Homing](#) (30)

**Example for a movement type with level-activated function: manual jog (jogging)**

1. In the inhibited axis state ($xOperationEnabled = FALSE$), $xJogPos$ is set to TRUE.
 - $xEnableOperation = FALSE$ (axis is inhibited.)
==> "STANDBY" state ($xOperationEnabled = FALSE$)
 - $xJogPos = TRUE$ (manual jog is to be executed.)
2. Enable axis.
 - $xEnableOperation = TRUE$
==> "JOGGING" state ($xOperationEnabled = TRUE$)

**Example for a movement type with edge-activated function: homing**

1. In the inhibited axis state ($xOperationEnabled = FALSE$), $xHomeExecute$ is set to TRUE.
 - $xEnableOperation = FALSE$ (axis is inhibited.)
==> "STANDBY" state ($xEnableOperation = FALSE$)
 - $xHomeExecute = TRUE$ (homing is to be executed.)
2. Enable axis.
 - $xEnableOperation = TRUE$
==> "STANDBY" state (output $xOperationEnabled = TRUE$)
3. Execute homing.
 - $xHomeExecute = FALSE \rightarrow TRUE$
==> "HOMING" state

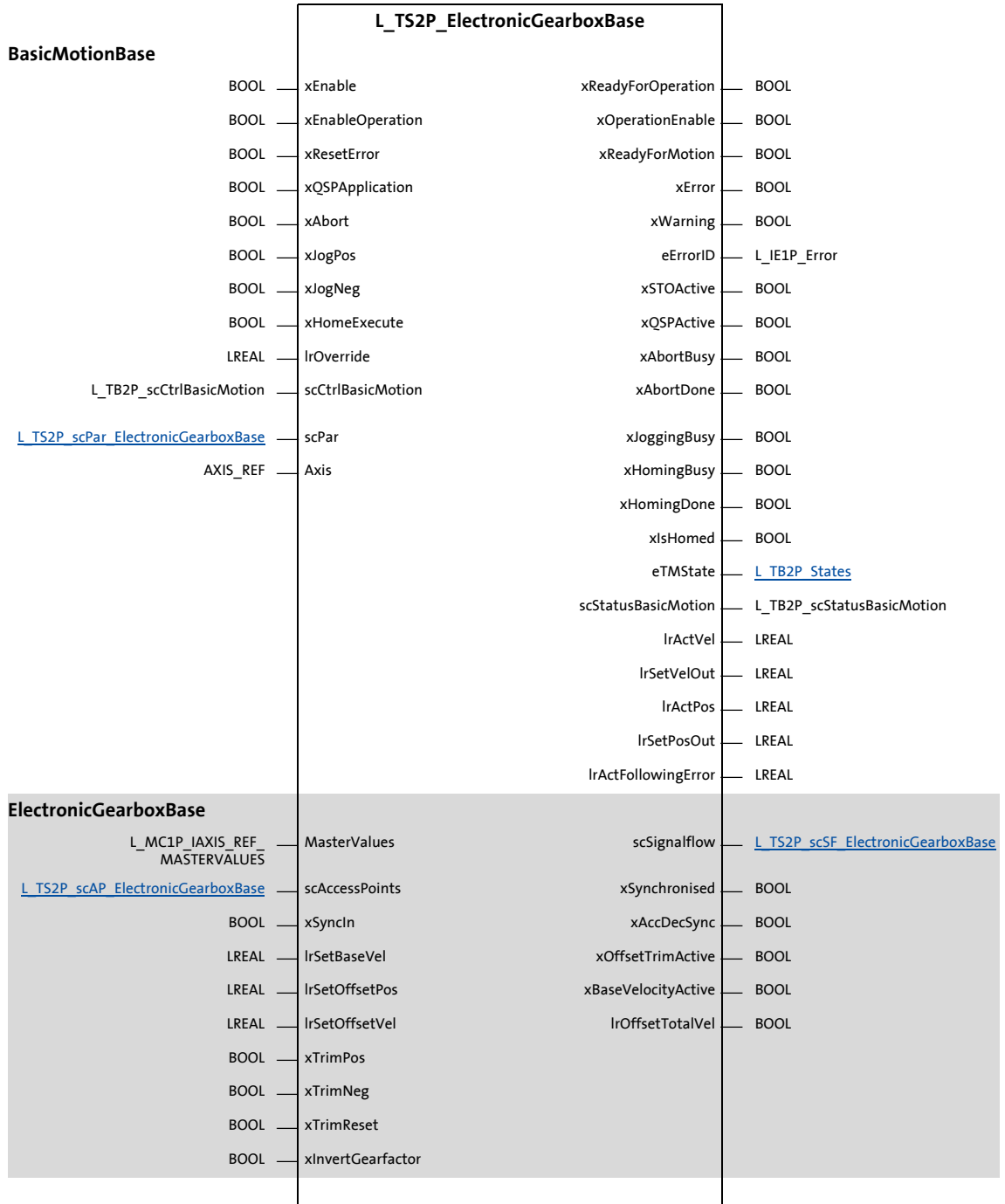
4 Application ranges

Typical applications:

- Slitters
- Calender drives
- Line drives
- Conveying belts
- Vibrators
- Roller mills
- Stretching machines
- Wire drawing machines

5 L_TS2P_ElectronicGearboxBase function block

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



5.1

Inputs

Designator	Data type	Description
xEnable	BOOL	Execution of the function block
		TRUE The function block is executed.
		FALSE The function block is not executed.
xEnableOperation	BOOL	TRUE The axis is switched to ready for operation. Controller enable is issued.
		FALSE The axis is not ready for operation.
xResetError	BOOL	The input is edge-controlled and evaluates the rising edge.
		FALSE Reset axis error or software error. TRUE The error is reset.
xQSPApplication	BOOL	The input is level-controlled and evaluates the high level.
		TRUE Active motion is aborted and axis is brought to a standstill via the QSP application deceleration ramp (33). • The state changes to "STOP". • The technology module remains in the "STOP" state as long as <i>xQSPApplication</i> = TRUE.
xAbort	BOOL	TRUE Active motion is aborted and axis is brought to a standstill via the active profile deceleration. • The state changes to "STOP". The deceleration ramp can be aborted by activating a new motion (e.g. <i>xMoveVelExecute</i>)
		FALSE The active motion is not aborted.
xJogPos	BOOL	The input is level-controlled and evaluates a high level.
		TRUE Traverse axis in positive direction (manual jog). If <i>xJogNeg</i> is also TRUE, the "InvalidInputJogPosJogNeg" error message is output via the <i>eErrorID</i> output and the axis stops.
xJogNeg	BOOL	The input is level-controlled and evaluates a high level.
		TRUE Traverse axis in negative direction (manual jog). If <i>xJogPos</i> is also TRUE, the "InvalidInputJogPosJogNeg" error message is output via the <i>eErrorID</i> output and the axis stops.
xHomeExecute	BOOL	The input is edge-controlled and evaluates the rising edge.
		FALSE Start homing. TRUE The function is aborted via the <i>scCtrlBasicMotion.xHalt</i> structure input.
lrOverride	LREAL	Using the <i>lrOverride</i> factor, the specified setpoint speed is multiplied (e.g. an override value of 0.5 generates half the setpoint speed) • Value range: 0 ... 1.0 • Initial value: 1.0 Note: The acceptance of the value is independent of the status in <i>xContinuousUpdate</i>
scCtrlBasicMotion L_TB2P_scCtrlBasicMotion		Structure of the status data You can find information on status data here: "Basic Motion Base V2" reference manual
Axis AXIS_REF		Reference to the axis
MasterValues L_MC1P_IAXIS_REF_MASTERVERVALUES		Connection for the master values (master axis)
scPar L_TS2P_scPar_ElectronicGearboxBase		The parameter structure contains the parameters of the technology module. ► L_TS2P_scPar_ElectronicGearboxBase (19)
scAccessPoints L_TS2P_scAP_ElectronicGearboxBase		Structure of the access points The access points can be used to influence the signal flow or technology module functions. ► L_TS2P_scAP_ElectronicGearboxBase (26)

Designator	Data type	Description
xSyncln	BOOL	The input is level-controlled and evaluates a high level.
		Synchronize slave axis to the master values of the master axis. During the process, the <i>xAccDecSync</i> output = TRUE. If there is synchronicity (clutch closed), the <i>xSynchronized</i> output = TRUE. • Initial value: FALSE
		TRUE Clutch-in in any position
		FALSE Declutching The slave axis decelerates at <i>scPar.lrSyncDec</i> to a standstill.
lrSetBaseVel	LREAL	Basic velocity ▶ Synchronism with basic speed (□ 61) • Initial value: 0 • Unit: unit/s
lrSetOffsetPos	LREAL	Position offset to the master axis The <i>scPar.xLoadOffset</i> parameter makes it possible to specify whether the position offset is cyclic or loaded via a profile generator. ▶ Position offset via profile generator (□ 57) • Initial value: 0 • Unit: unit
lrSetOffsetVel	LREAL	Velocity offset to the master axis ▶ Position offset via profile generator (□ 57) • Initial value: 0 • Unit: unit/s
xTrimPos	BOOL	Trimming in positive direction If <i>xTrimPos</i> AND <i>xTrimNeg</i> = TRUE, the first selected travel direction is maintained. The evaluation of this input depends on the set trimming mode (<i>scPar.eTrimMode</i> parameter). ▶ Trimming per velocity offset (□ 52) ▶ Trimming per position offset (□ 54) • Initial value: FALSE
		TRUE eTrimMode = 0 (trimming per velocity offset): The input is level-controlled and evaluates a high level.
		FALSE TRUE eTrimMode = 1 (trimming per position offset): The input is edge-controlled and evaluates the rising edge.
xTrimNeg	BOOL	Trimming in negative direction If <i>xTrimPos</i> AND <i>xTrimNeg</i> = TRUE, the first selected travel direction is maintained. The evaluation of this input depends on the set trimming mode. ▶ Trimming per velocity offset (□ 52) ▶ Trimming per position offset (□ 54) • Initial value: FALSE
		TRUE eTrimMode = 0 (trimming per velocity offset): The input is level-controlled and evaluates a high level.
		FALSE TRUE eTrimMode = 1 (trimming per position offset): The input is edge-controlled and evaluates the rising edge.
xTrimReset	BOOL	Reset the position offset resulting from the trimming. The input is edge-controlled and evaluates the rising edge. ▶ Trimming per velocity offset (□ 52) ▶ Trimming per position offset (□ 54) • Initial value: FALSE
		FALSE TRUE The position offset is moved to zero via the profile generator.
xInvertGearfactor	BOOL	Inversion of the gearbox factor. If the inversion is not allowed due to the current target speed, a warning (ErrorID 17191) is issued.

5.2

Outputs

Designator	Data type	Description
xReadyForOperation	BOOL	The axis is in correct working order. The DC-bus voltage is available. "Safe Torque Off" (STO) is not active.
xOperationEnabled	BOOL	The axis is enabled. The brake is released.
xReadyForMotion	BOOL	The axis is enabled and in correct working order. The brake is released.
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 <i>xError</i> = TRUE or <i>xWarning</i> = TRUE. "FAST technology modules" reference manual: Here you can find information on error or warning messages.
xSTOActive	BOOL	TRUE "Safe Torque Off" (STO) was triggered and is active. The axis is free of torque, the controller is inhibited.
xQSPActive	BOOL	TRUE The <i>xQSPApplication</i> input was set. The technology module is in the "STOP" state; the axis is brought to a standstill. The output is reset as soon as the axis is at standstill.
xAbortBusy	BOOL	TRUE The activated function is aborted via the <i>xAbsort</i> input.
xAbsortDone	BOOL	TRUE The activated function has been aborted via the <i>xAbsort</i> input.
xJoggingBusy	BOOL	TRUE The Manual jog (jogging) (□ 28) function is active. The technology module is in the "JOGGING" state.
xHomingBusy	BOOL	TRUE The Homing (□ 30) is executed. The technology module is in the "HOMING" state.
xHomingDone	BOOL	TRUE Homing has been successfully completed.
xisHomed	BOOL	TRUE The axis has been referenced ("Reference known"). <i>xisHomed</i> may only be evaluated when the <i>scStatusBasicMotion.xAxisAvailable</i> status is reported.
eTMState	L_TB2P_States	Current state of the technology module
scStatusBasicMotion	L_TB2P_scStatusBasicMotion	Structure of the status data You can find information on status data here: "Basic Motion Base V2" reference manual
IrActVel	LREAL	Current speed • Unit: unit/s
IrSetVelOut	LREAL	Setpoint speed • Unit: unit/s
IrActPos	LREAL	Current position • Unit: unit
IrSetPosOut	LREAL	Setpoint position • Unit: unit
IrActFollowingError	LREAL	Current following error • Unit: unit

Designator	Data type	Description
scSignalflow L_TS2P_scSF_ElectronicGearboxBase		Structure of the signal flow ▶ L_TS2P_scSF_ElectronicGearboxBase (25)
xSynchronised	BOOL	TRUE The velocity of the slave axis is synchronized to the master axis. The technology module is in the "SYNCHRONIZED" state. ▶ Clutching-in/declutching (50) ▶ Synchronism with basic speed (61)
xAccDecSync	BOOL	TRUE A synchronization function is active. The axis is synchronized or desynchronized (i.e. clutch opens or closes). The technology module is in the "ACC_DCC_SYNC" state. ▶ Clutching-in/declutching (50) ▶ Synchronism with basic speed (61)
xOffsetTrimActive	BOOL	TRUE An offset or trimming function is active. ▶ Trimming per velocity offset (52) ▶ Trimming per position offset (54) ▶ Velocity offset via profile generator (59) ▶ Position offset via profile generator (57)
xBaseVelocityActive	BOOL	TRUE The clutch is open and the slave axis moves at the basic speed set at the <i>IrSetBaseVel</i> input. The technology module is in the "BASEVEL" state. ▶ Synchronism with basic speed (61)
IrOffsetTotalVel	LREAL	Total velocity offset (sum of the measuring points <i>MP_IrVelOffset</i> and <i>MP_IrPosOffsetTotalVel</i>) • Unit: unit/s

5.3

Parameter

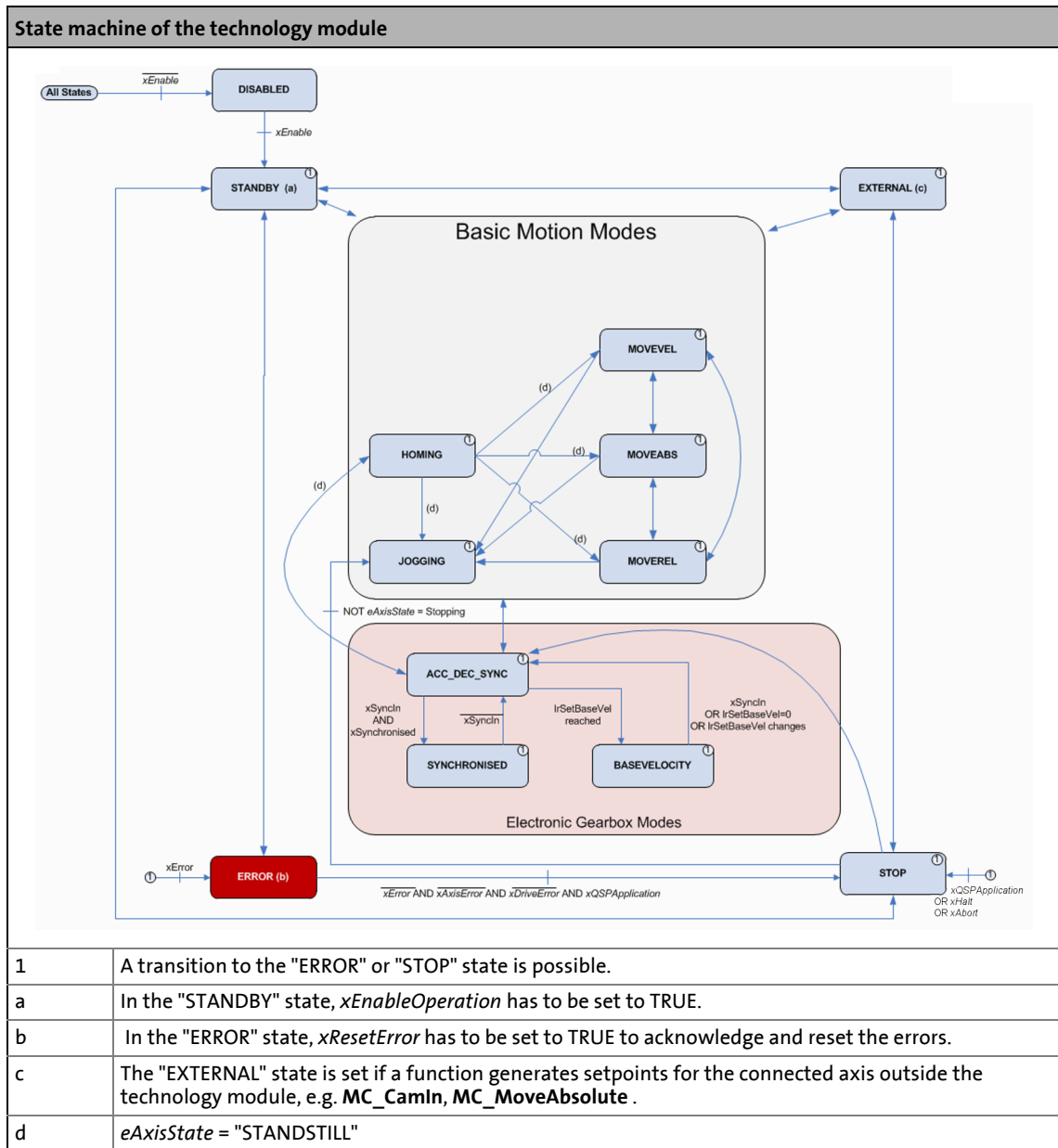
L_TS2P_scPar_ElectronicGearboxBase

Designator	Data type	Description
IrVel	LREAL	Speed for motions activated via the scCtrlBasicMotion structure. You can find information on status data here: "Basic Motion Base" reference manual - Unit: unit/s - Initial value: 3600
IrAcc	LREAL	Acceleration for motions activated via the scCtrlBasicMotion structure. Specification of the maximum velocity variation which is to be used for acceleration. - Unit: unit/s² - Initial value: 36000
IrDec	LREAL	Deceleration for motions activated via the scCtrlBasicMotion structure. Specification of the maximum speed variation which is to be used for deceleration to standstill. - Unit: unit/s² - Initial value: 100
IrJerk	LREAL	Jerk for motions activated via the scCtrlBasicMotion structure. - Unit: unit/s³ - Initial value: 0
eDirection	MC_DIRECTION	Traversing direction: <i>mcCurrentDirection</i>: Retain current direction (only in case of modulo axis) <i>mcPositiveDirection</i>: Positive direction <i>mcNegativeDirection</i>: Negative direction <i>mcShortestWay</i>: Shortest distance (only in case of modulo axis) - Initial value: <i>mcPositiveDirection</i>
scHomeExtTP	MC_TRIGGER_REF	Transfer of an external touch probe event: - Only relevant for "external encoder" touch probe configuration. - For describing the MC_TRIGGER_REF structure, see the MC_TouchProbe function block.
dwGearingNumerator	DWORD	Gearbox numerator The setting of the parameter is effective in the "unit coupling" mode (<i>dwCouplingMode</i> = 0). Parameter changes are accepted immediately. ▶ Set gearbox factor (□ 47) - Unit: - - Initial value: 1
dwGearingDenominator	DWORD	Gearbox factor denominator The setting of the parameter is effective in the "unit coupling" mode (<i>dwCouplingMode</i> = 0). Parameter changes are accepted immediately. ▶ Set gearbox factor (□ 47) - Unit: - - Initial value: 1
IrSyncAcc	LREAL	Acceleration for the clutch-in/declutch process. Selection of the maximum speed variation which is to be used for acceleration. ▶ Clutching-in/declutching (□ 50) ▶ Synchronism with basic speed (□ 61) - Unit: unit/s² - Initial value: 1 000
IrSyncDec	LREAL	Deceleration for the clutch-in/declutch process. Specification of the maximum speed variation which is to be used for deceleration. ▶ Clutching-in/declutching (□ 50) ▶ Synchronism with basic speed (□ 61) - Unit: unit/s² - Initial value: 1 000

Designator	Data type	Description						
IrSyncJerk	LREAL	Jerk for the clutch-in/declutch process. Specification of the maximum acceleration/deceleration variation which is to be used for the clutch-in/declutch process. ▶ Clutching-in/declutching (▢ 50) ▶ Synchronism with basic speed (▢ 61) • Unit: unit/s ³ • Initial value: 1 000 000						
eAllowedDirection L_TB2P_AllowedDirection		Allowed direction of rotation Limit setpoints to allowed travel directions (▢ 63) • Initial value: positive and negative direction (0) <table><tr><td>0</td><td>Positive and negative direction of rotation The slave axis may engage in both directions. In the case of offset or trim, it is allowed to move in both directions.</td></tr><tr><td>1</td><td>Only positive direction of rotation The slave axis may only engage in positive direction. An offset may only be moved in positive direction. A negative offset is only moved to the extent that the speed of the total conductance does not become less than '0'.</td></tr><tr><td>2</td><td>Only negative direction of rotation The slave axis may only engage in negative direction. An offset may only be moved in negative direction. A positive offset is only moved to the extent that the speed of the total conductance does not exceed '0'.</td></tr></table>	0	Positive and negative direction of rotation The slave axis may engage in both directions. In the case of offset or trim, it is allowed to move in both directions.	1	Only positive direction of rotation The slave axis may only engage in positive direction. An offset may only be moved in positive direction. A negative offset is only moved to the extent that the speed of the total conductance does not become less than '0'.	2	Only negative direction of rotation The slave axis may only engage in negative direction. An offset may only be moved in negative direction. A positive offset is only moved to the extent that the speed of the total conductance does not exceed '0'.
0	Positive and negative direction of rotation The slave axis may engage in both directions. In the case of offset or trim, it is allowed to move in both directions.							
1	Only positive direction of rotation The slave axis may only engage in positive direction. An offset may only be moved in positive direction. A negative offset is only moved to the extent that the speed of the total conductance does not become less than '0'.							
2	Only negative direction of rotation The slave axis may only engage in negative direction. An offset may only be moved in negative direction. A positive offset is only moved to the extent that the speed of the total conductance does not exceed '0'.							
xLimitAllowedDirection	BOOL	Activate total setpoint limitation to the allowed range (eAllowedDirection). <table><tr><td>TRUE</td><td>The speed of the total setpoint is limited to the allowed range (eAllowedDirection). For example, a negative total speed setpoint is not possible with eAllowedDirection = 1 (only positive direction of rotation).</td></tr><tr><td>FALSE</td><td>The speed of the total setpoint can be in a non-allowed range if the conductivity speed is negative.</td></tr></table>	TRUE	The speed of the total setpoint is limited to the allowed range (eAllowedDirection). For example, a negative total speed setpoint is not possible with eAllowedDirection = 1 (only positive direction of rotation).	FALSE	The speed of the total setpoint can be in a non-allowed range if the conductivity speed is negative.		
TRUE	The speed of the total setpoint is limited to the allowed range (eAllowedDirection). For example, a negative total speed setpoint is not possible with eAllowedDirection = 1 (only positive direction of rotation).							
FALSE	The speed of the total setpoint can be in a non-allowed range if the conductivity speed is negative.							
IrToleranceAllowedDirection	LREAL	Maximum setpoint difference in a blocked direction (eAllowedDirection) before an error is triggered. This tolerance window is only effective if xLimitAllowedDirection = TRUE. • Unit: units • Initial value: 5						
IrSetOffsetTrimVel	LREAL	Speed for the offset and trimming functions Specification of the maximum speed for the additional profile of the offset and trimming functions. ▶ Trimming per velocity offset (▢ 52) ▶ Trimming per position offset (▢ 54) ▶ Velocity offset via profile generator (▢ 59) ▶ Position offset via profile generator (▢ 57) • Unit: unit/s • Initial value: 10						
IrSetOffsetTrimAcc	LREAL	Acceleration for the offset and trimming functions Specification of the speed variation by which the slave axis should accelerate relative to the master axis. The acceleration acting on the drive is the sum of the master and slave accelerations. ▶ Trimming per velocity offset (▢ 52) ▶ Trimming per position offset (▢ 54) ▶ Velocity offset via profile generator (▢ 59) ▶ Position offset via profile generator (▢ 57) • Unit: unit/s ² • Initial value: 100						

Designator	Data type	Description
IrSetOffsetTrimDcc	LREAL	Deceleration for the offset and trimming functions Specification of the speed variation by which the slave axis should decelerate relative to the master axis. The deceleration acting on the drive is the sum of the master and slave decelerations. ▶ Trimming per velocity offset (45 52) ▶ Trimming per position offset (45 54) ▶ Velocity offset via profile generator (45 59) ▶ Position offset via profile generator (45 57) • Unit: unit/s² • Initial value: 100
IrSetOffsetTrimJerk	LREAL	Jerk for the offset and trimming functions ▶ Trimming per velocity offset (45 52) ▶ Trimming per position offset (45 54) ▶ Velocity offset via profile generator (45 59) ▶ Position offset via profile generator (45 57) • Unit: unit/s³ • Initial value: 10 000
xLoadOffset	BOOL	Mode to load the position offset to the master axis (<i>IrSetOffsetPos</i> input) ▶ Velocity offset via profile generator (45 59) • Initial value: FALSE
		TRUE The position offset is loaded cyclically. The profile generator is deactivated.
		FALSE The position offset is moved via the profile generator.
eTrimMode L_TS2P_TrimMode		Trimming mode A trimming can be triggered via the inputs <i>xTrimPos</i> or <i>xTrimNeg</i> = TRUE. ▶ Trimming per velocity offset (45 52) ▶ Trimming per position offset (45 54) • Initial value: 1
		0 Trimming per velocity offset
		1 Trimming per position offset (<i>scPar.IrTrimDist</i>)
IrTrimDist	LREAL	Only in case of <i>scPar.eTrimMode</i> = 1 (trimming per position offset): Setting of the position offset to be trimmed. The position offset set here will be trimmed in positive/negative direction for each rising edge at the <i>xTrimPos</i> / <i>xTrimNeg</i> input. ▶ Trimming per position offset (45 54) • Unit: unit • Initial value: 1
eCouplingMode L_TS2P_CouplingMode		Gearbox-coupling factor ▶ Set gearbox factor (45 47) • Initial value: 1
		0 Unit clutch
		1 Coupling via cycle lengths
IrToleranceInvertGearfactor	LREAL	Invert gearbox factor If the setpoint speed of the axis is within the parameterised window, the inversion of the gearbox factor is permitted via the input <i>xInvertGearfactor</i> . DANGER: The inversion takes place abruptly! ▶ Invert gearbox factor (45 49) • Unit: unit/s • Initial value: 0

6 State machine



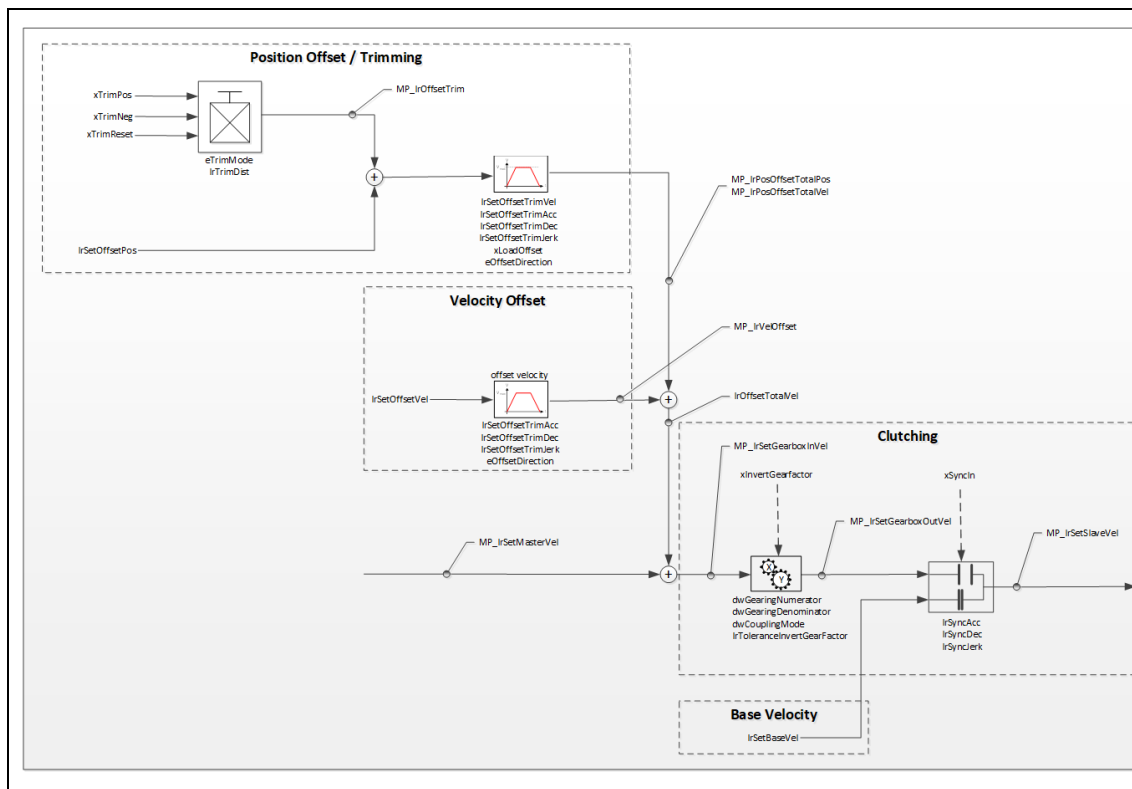
States of the eTMState output (L_TB2P_States)

No.	L_TB2P_States	Description
-1	DISABLED	Technology module deactivated
0	STANDBY	Technology module is on standby
1	STOP	Stop/Halt active
2	HOMING	Homing active
3	JOGGING	Manual jog active
4	MOVEVEL	Continuous travel with a defined velocity active
5	MOVEABS	Absolute positioning active
6	MOVEREL	Relative positioning active
7	EXTERNAL	An external function generates setpoints for the connected axis
8	ERROR	Error state
30	ACC_DEC_SYNC	Synchronization function active
31	SYNCHRONIZED	Axes are synchronized
51	BASEVELOCITY	Axis rotates at basic speed
1000	SYSTEMFAULT	System fault

7 Signal flow diagram

7 Signal flow diagram

The following figure shows the main signal flow of the function:



[7-1] Signal flow of the technology module

7.1 Structure of the signal flow

L_TS2P_scSF_ElectronicGearboxBase

The contents of the **L_TS2P_scSF_ElectronicGearboxBase** structure are read-only and offer a practical diagnostics option within the signal flow ([Signal flow diagram](#) (□ 24)).

Designator	Data type	Description
MP_IrSetMasterVel	LREAL	Setpoint velocity of the master axis • Unit: Units
MP_IrSetGearboxInVel	LREAL	Master speed before the measuring system adaptation • Unit: Units
MP_IrSetGearboxOutVel	LREAL	Master speed for the coupling after the measuring system adaptation • Unit: Units
MP_IrSetSlaveVel	LREAL	Setpoint velocity of the slave axis • Unit: Units
MP_IrVelOffset	LREAL	Current velocity offset Profile generator output for the velocity offset relating to the master axis as per the <i>IrSetOffsetVel</i> input. ▶ Velocity offset via profile generator (□ 59) • Unit: Units
MP_IrPosOffsetTotalPos	LREAL	Total position offset Position output of the profile generator, which moves the position offset set at input <i>IrSetOffsetPos</i> or a trim ▶ Position offset via profile generator (□ 57) • Unit: units
MP_IrPosOffsetTotalVel	LREAL	Profile generator speed output that moves the position offset set at the <i>IrSetOffsetPos</i> input or a trimming. ▶ Position offset via profile generator (□ 57) • Unit: Units
MP_IrOffsetTrim	LREAL	Trim position offset Already trimmed position. Note: The value is not updated until the trim is completed. There is no update during the trimming process. • Unit: units

7.2 Structure of the access points

L_TS2P_scAP_ElectronicGearboxBase

The access points (AP) can be used to influence signals. In the initial state, the access points do not have any effect.

Each access point acts as an alternative branch and is activated via an OR operation or a switch.

Designator	Data type	Description
AP_xGearingNumerator	BOOL	Enabling of the <i>AP_dwGearingNumerator</i> access point
	TRUE	The access point overwrites the values at the access point in the signal flow. Here the coupling factor in the <i>scPar.dwGearingNumerator</i> parameter becomes ineffective.
AP_dwGearingNumerator	LREAL	Active coupling factor for the Y axis The setting of the parameter is effective in the "unit coupling" mode (<i>scPar.eCouplingMode</i> = 0). The coupling factor can also be modified cyclically in the clutched-in state.
AP_xGearingDenominator	BOOL	Enabling of the <i>AP_dwGearingDenominator</i> access point
	TRUE	The access point overwrites the values at the access point in the signal flow. Here the coupling factor in the <i>scPar.dwGearingDenominator</i> parameter becomes ineffective.
AP_dwGearingDenominator	LREAL	Active coupling factor for the X axis The setting of the parameter is effective in the "unit coupling" mode (<i>scPar.eCouplingMode</i> = 0). The coupling factor can also be modified cyclically in the clutched-in state.

8 Description of the basic functions

Overview of the basic functions

- ▶ [Manual jog \(jogging\)](#) (📖 28)
- ▶ [Homing](#) (📖 30)
- ▶ [QSP application](#) (📖 33)
- ▶ [Halt](#) (📖 34)
- ▶ [Absolute positioning](#) (📖 35)
- ▶ [Relative positioning](#) (📖 37)
- ▶ [Continuous travel with a defined velocity](#) (📖 40)
- ▶ [Aborting and completing a positioning process](#) (📖 43)
- ▶ [Override for speed](#) (📖 45)

8 Description of the basic functions

8.1 Manual jog (jogging)

8.1 Manual jog (jogging)

Prerequisite

- The axis is enabled (*xReadyForMotion* = TRUE).

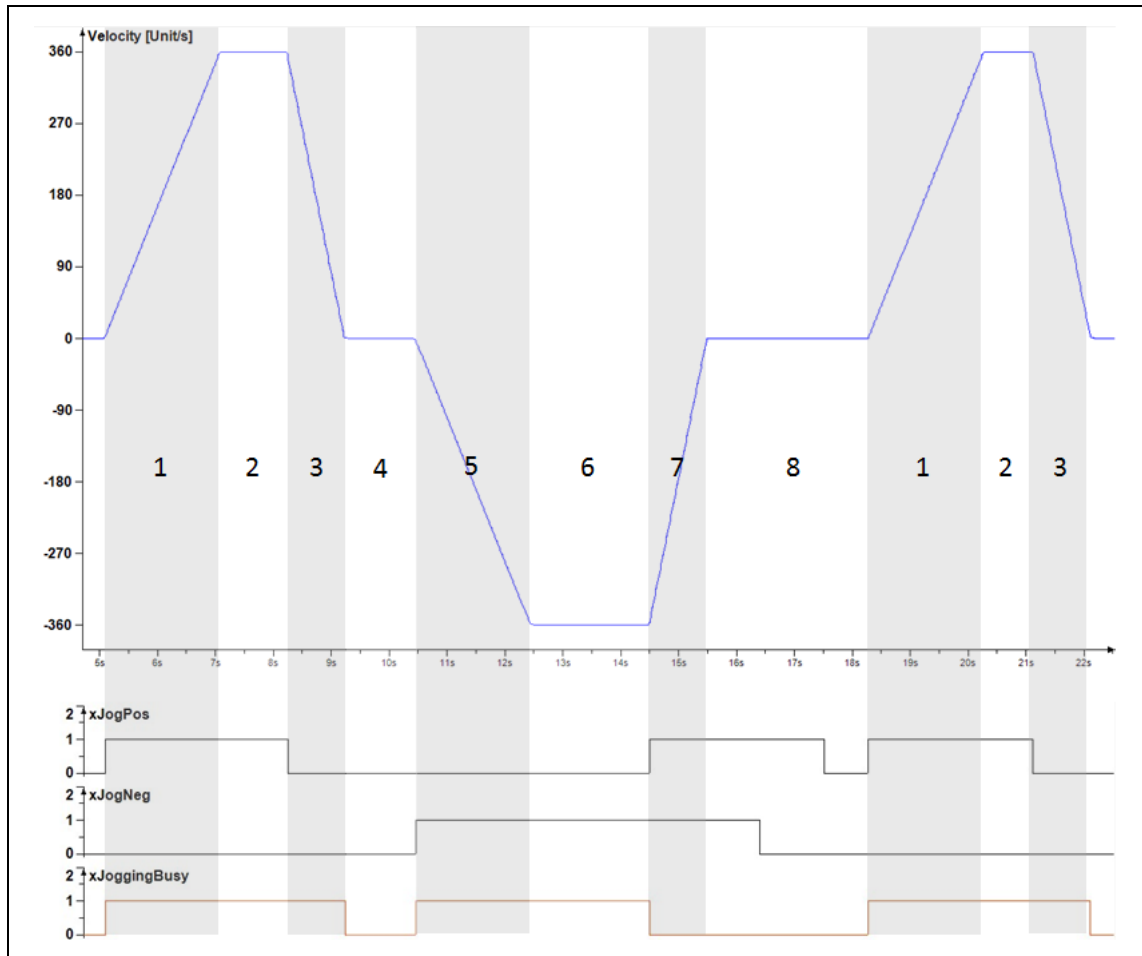
Execution

With the *xJogPos* input = TRUE, the axis is traversed in the positive direction and with the *xJogNeg* input = TRUE, it is traversed in the negative direction. The axis is traversed as long as *xJogPos* or *xJogNeg* remain set to TRUE.

The current travel command cannot be replaced by the other jog command. If *xJogPos* = TRUE and *xJogNegPos* = TRUE, the axis brakes into standstill and the *eTMState* status output changes to the "ERROR" state.

Only if both inputs have been reset, the technology module changes back to the "STANDBY" state.

Travel diagram



No	xJogPos	xJogNeg	xJogging Busy	eTMState	Info
1	FALSE → TRUE	FALSE	FALSE → TRUE	STANDBY → JOGGING	The axis accelerates to the positive manual speed 360 units/s with 180 unit/s².
2	TRUE	FALSE	TRUE	JOGGING	The axis travels constantly with 360 units/s.
3	FALSE → TRUE	FALSE	TRUE	JOGGING	The axis decelerates to a standstill with 360 unit/s².
4	FALSE	FALSE	TRUE → FALSE	JOGGING → STANDBY	The axis has reached a standstill.
5	FALSE	FALSE → TRUE	FALSE → TRUE	STANDBY → JOGGING	The axis accelerates to the negative manual speed -360 units/s with 180 unit/s².
6	FALSE	TRUE	TRUE	JOGGING	The axis travels constantly with -360 units/s.
7	TRUE	TRUE	TRUE → FALSE	ERROR	The axis decelerates to a standstill with 360 unit/s².
8	FALSE	FALSE	FALSE	ERROR → STANDBY	The axis is at a standstill and changes to "STANDBY" if both inputs xJogPos and xJogNeg have the value FALSE.

Parameters to be set

The parameters for manual jog can be found in the AXIS_REF in subindex 181 ... 184.

The parameter values can be changed during operation. They are accepted and become effective immediately when setting the inputs xJogPos = TRUE or xJogNeg = TRUE.

The manual jog parameters from the PLC program can also be written via the L_MC1P_SetBasicParameter block.

8.2 Homing

Prerequisites

- The technology module is in "STANDBY" mode.
- The axis is enabled (*xReadyForMotion* = TRUE).

Execution

With a rising edge (FALSE→TRUE) at the *xHomeExecute* input, homing is started (output *xHomingBusy* = TRUE). The axis travels until the home position is reached.

When the reference switch has been reached, *xisHomed* switches to TRUE.

When the homing process has been completed successfully, *xHomingDone* changes to TRUE.

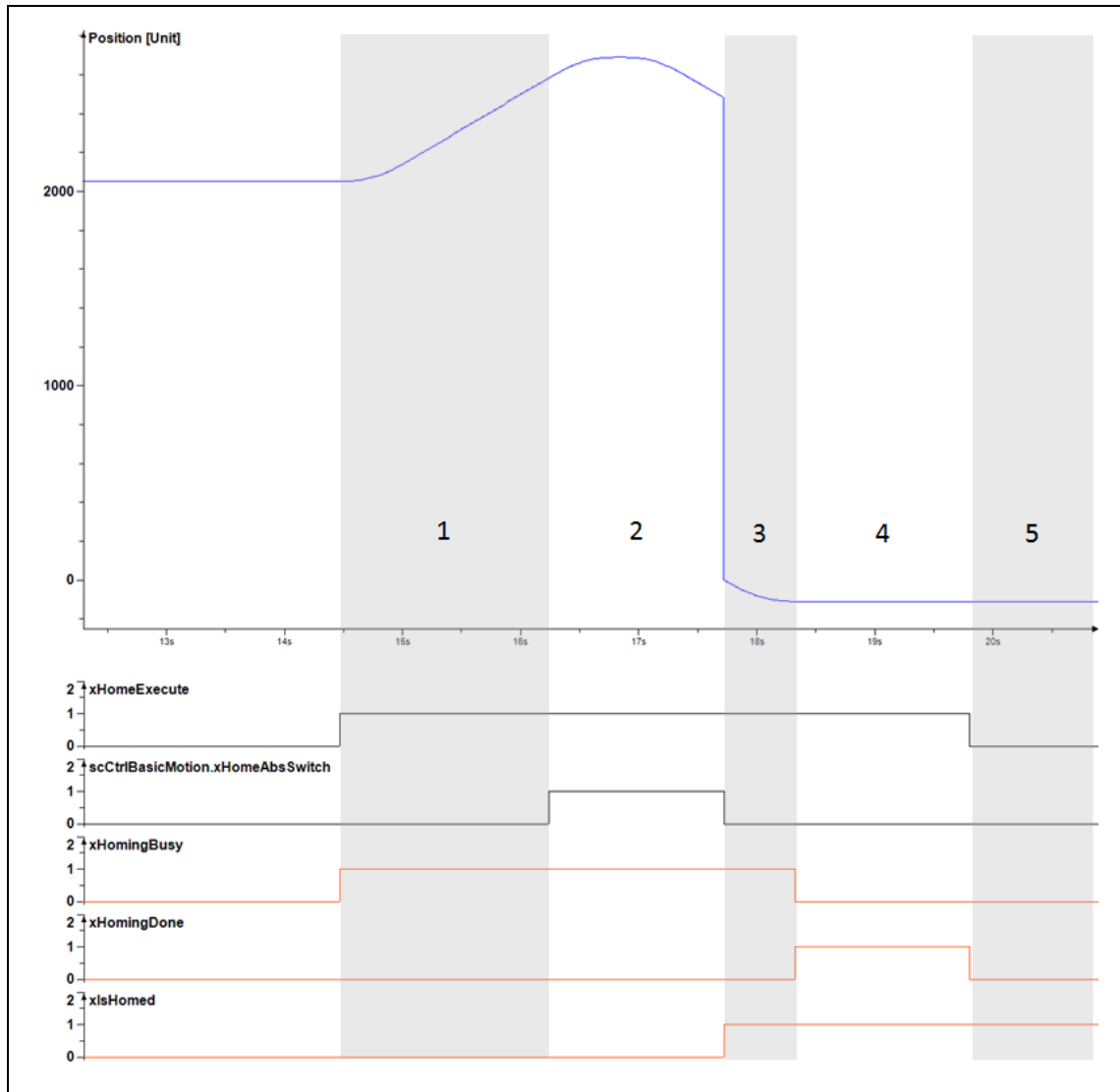
If *xHomeExecute* = FALSE, the state is changed to "STANDBY" and *xHomingDone* is set = FALSE.

Abort of the homing process

The function is aborted via the *xHalt* input.

Homing is not interrupted if *xHomeExecute* is set to = FALSE early.

Travel diagram



[8-1] Homing in the CwRpCwRn module with the home position "0"

No	xHome Execute	xHome AbsSwitch	xHoming Busy	xHoming Done	xIsHomed	eTMState	Info
1	FALSE → TRUE	FALSE	FALSE → TRUE	FALSE	FALSE	STANDBY → HOMING	The axis starts homing in positive direction.
2	TRUE	TRUE → FALSE	TRUE	FALSE	FALSE	HOMING	The axis changes the direction of rotation due to the positive edge of xHomeAbsSwitch.
3	TRUE	FALSE → TRUE	TRUE	FALSE	FALSE → TRUE	HOMING	The negative edge of xHomeAbsSwitch sets the position of the axis to the home position 0 units and the axis is brought to a standstill.
4	TRUE	FALSE	TRUE → FALSE	FALSE → TRUE	TRUE	HOMING	As soon as the axis has reached the standstill, xHomingBusy is set = FALSE and xHomingDone is set = TRUE. xHomingDone remains TRUE for as long as xHomeExecute is set = TRUE, but always at least for one clock cycle.
5	TRUE → FALSE	FALSE	FALSE	TRUE → FALSE	TRUE	HOMING → STANDBY	The axis is ready for operation and can be positioned in an absolute fashion.

Parameters to be set

The parameters for homing can be found in the `AXIS_REF` in subindex 70 ... 83.

The manual jog parameters from the PLC program can also be written via the `L_MC1P_SetBasicParameter` block.

8.3 QSP application

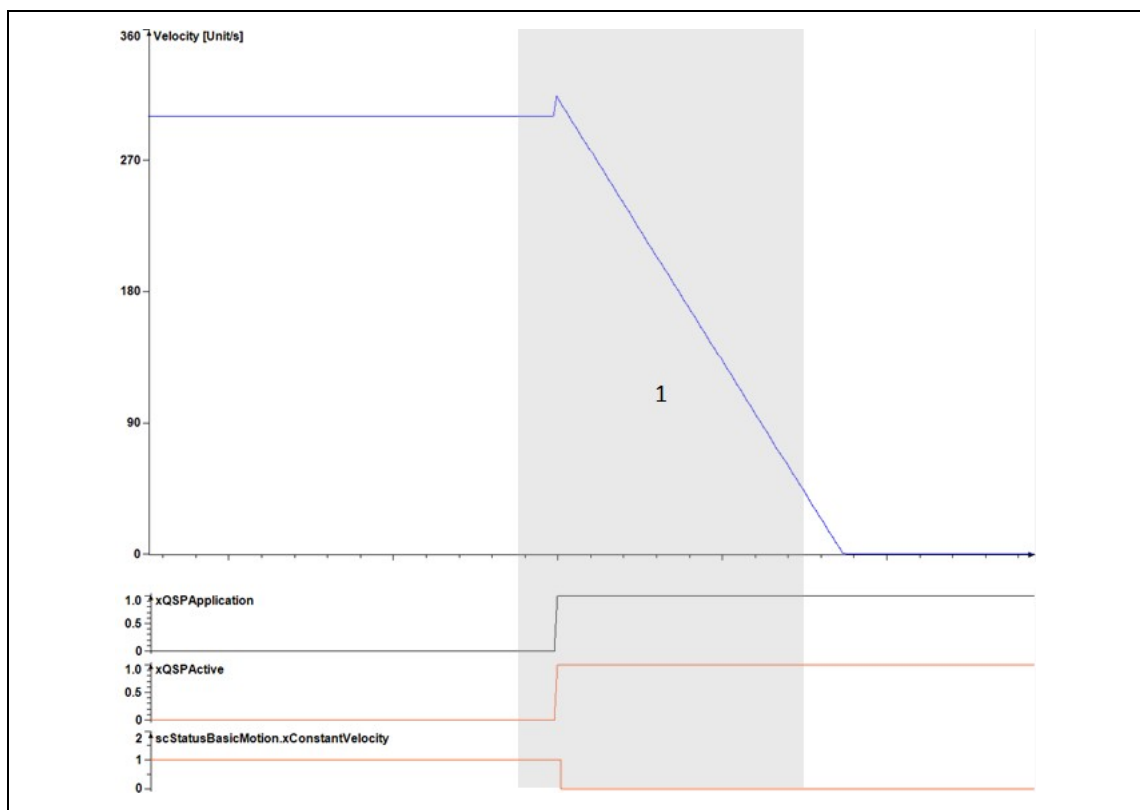
Prerequisite

- The axis is enabled ($xReadyForMotion = TRUE$).

Execution

With a rising edge ($FALSE \rightarrow TRUE$) at the $xQSPApplication$ input, the current movement of the axis is interrupted and the axis is brought to a standstill on the basis of its last current speed value on the QSP application deceleration ramp. The parameter for the QSP application deceleration ramp is set in the AXIS_REF.

Travel diagram



No	xQSP Application	xQSPActive	xConstant Velocity	eTMState	Info
1	$FALSE \rightarrow TRUE$	$FALSE \rightarrow TRUE$	$TRUE \rightarrow FALSE$	MOVEVEL → STOP	The speed setpoint of the axis is set to the last current speed and brought to a standstill via the QSP deceleration ramp.

Parameters to be set

The parameters for "QSP Application" can be found in the AXIS_REF in subindex 48 and 49.

The manual jog parameters from the PLC program can also be written via the **L_MC1P_SetBasicParameter** block.

8.4 Halt

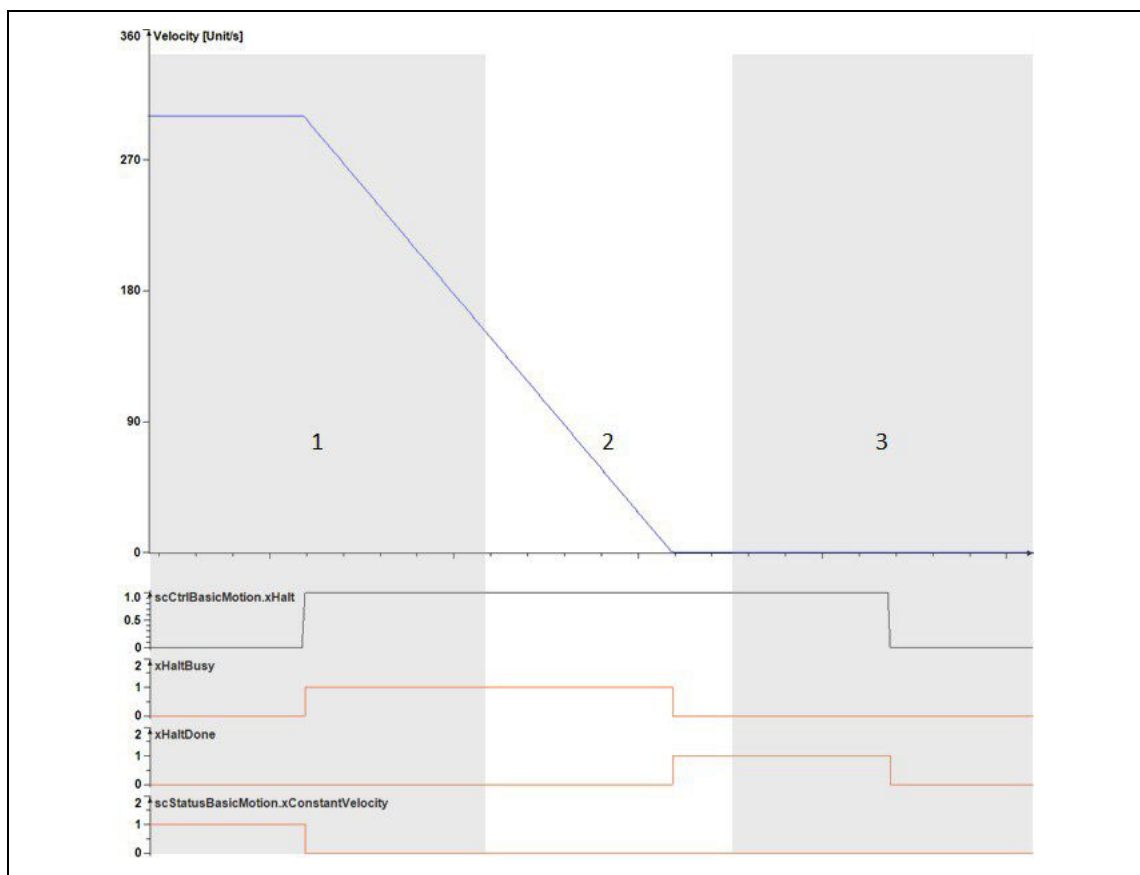
Prerequisite

- The axis is enabled ($xReadyForMotion = TRUE$).

Execution

With a rising edge ($FALSE \rightarrow TRUE$) at the `scCtrlBasicMotion.xHalt` structure input, the current movement of the axis is interrupted and the axis is brought to a standstill with a halt deceleration ramp.

Travel diagram



No	xHalt	xHaltBusy	xHaltDone	xConstant Velocity	eTMState	Info
1	$FALSE \rightarrow TRUE$	$FALSE \rightarrow TRUE$	$FALSE$	$TRUE \rightarrow FALSE$	$MOVEVEL \rightarrow STOP$	The axis decelerates with the halt deceleration ramp: $xHaltBusy = TRUE$ $xConstantVelocity = FALSE$.
2	$TRUE$	$TRUE \rightarrow FALSE$	$FALSE \rightarrow TRUE$	$FALSE$	$STOP$	The axis is brought to a standstill: $xHaltBusy = FALSE$ $xHaltDone = TRUE$
3	$TRUE \rightarrow FALSE$	$FALSE \rightarrow TRUE$	$TRUE \rightarrow FALSE$	$FALSE$	$STOP \rightarrow STANDBY$	$xHalt = FALSE$ $xHaltDone = TRUE \rightarrow FALSE$.

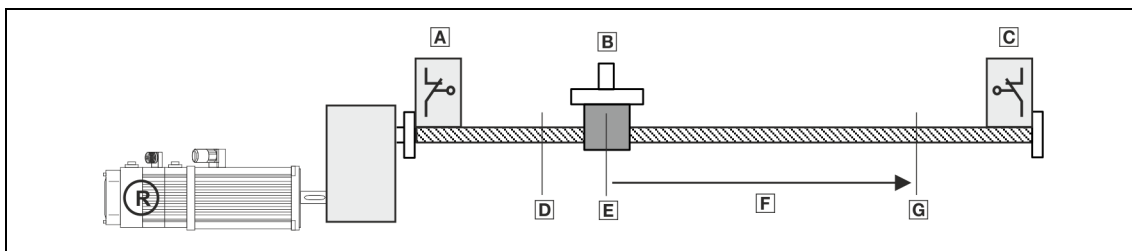
Parameters to be set

The parameter for the halt can be found in the `AXIS_REF` in subindex 186 and 187.

8 Description of the basic functions

8.5 Absolute positioning

8.5 Absolute positioning



[8-2] 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 (<i>lrPos_Dist</i> input)

With absolute positioning, the target position in the *lrPos_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).

Prerequisites

- The axis is enabled (*xReadyForMotion* = TRUE).
- The axis is referenced (*xisHomed* = 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 "MOVEABS" state.

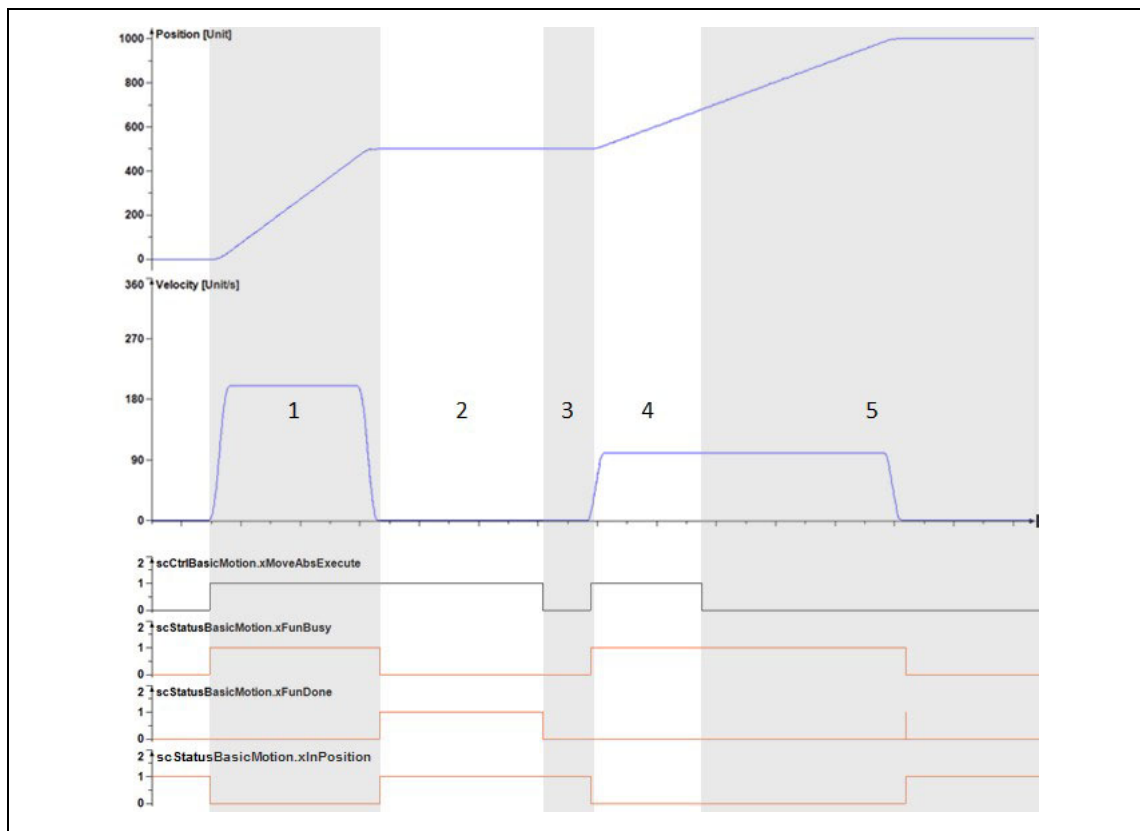
In order to reach the target position, the axis is brought to a standstill via the *lrDec* parameter. The axis reaches a standstill exactly in the target position.

After reaching the target position, the state changes back to STANDBY (*xMoveAbsExecute* = FALSE).

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

Positioning can be interrupted via *xAbort*, *xHalt* or *xQSPApplication*. Via *xFinishPositioning*, positioning can be continued again and completed.

Travel diagram



No	xMoveAbsExecute	xFuncBusy	xFuncDone	xInPosition	eTMState	Info
1	FALSE → TRUE	FALSE → TRUE	FALSE	TRUE → FALSE	STANDBY → MOVEABS	The axis starts the positioning run to the absolute target position 500 units.
2	TRUE	TRUE → FALSE	FALSE → TRUE	FALSE → TRUE	MOVEABS → STANDBY	When the target position has been reached, <i>xInPosition</i> is set to TRUE.
3	TRUE → FALSE	FALSE	TRUE → FALSE	TRUE	STANDBY	When <i>xMoveAbsExecute</i> is set to FALSE, the <i>xFuncDone</i> status signal changes from TRUE → FALSE.
4	FALSE → TRUE	FALSE → TRUE	FALSE	TRUE → FALSE	STANDBY → MOVEABS	The axis starts the positioning run to the absolute target position 1000 units.
5	TRUE → FALSE	TRUE → FALSE	FALSE → TRUE → FALSE	FALSE → TRUE	MOVEABS → STANDBY	After the target position has been reached, <i>xFuncDone</i> remains TRUE for exactly one cycle, because <i>xMoveAbsExecute</i> = FALSE has been requested during the positioning process.

Parameters to be set

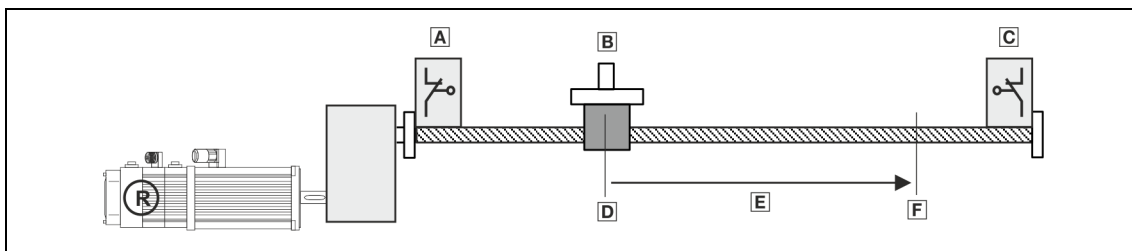
- *IrVel*, initial value: 3600 [units/s]
- *IrAcc*, initial value: 36000 [units/s²]
- *IrDcc*, initial value: 36000 [units/s²]
- *IrJerk*, initial value: 0 [units/s³]

Further parameters for absolute positioning can be found in the *AXIS_REF* in subindex 130 ... 137.

8 Description of the basic functions

8.6 Relative positioning

8.6 Relative positioning



[8-3] 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 (= <i>lrPos_Dist</i> input)
F	Target position (input current actual position + <i>lrPos_Dist</i>)

With relative positioning, the target position defined in the *lrPos_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).

Prerequisite

- The axis is enabled (*xReadyForMotion* = TRUE).

Execution

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

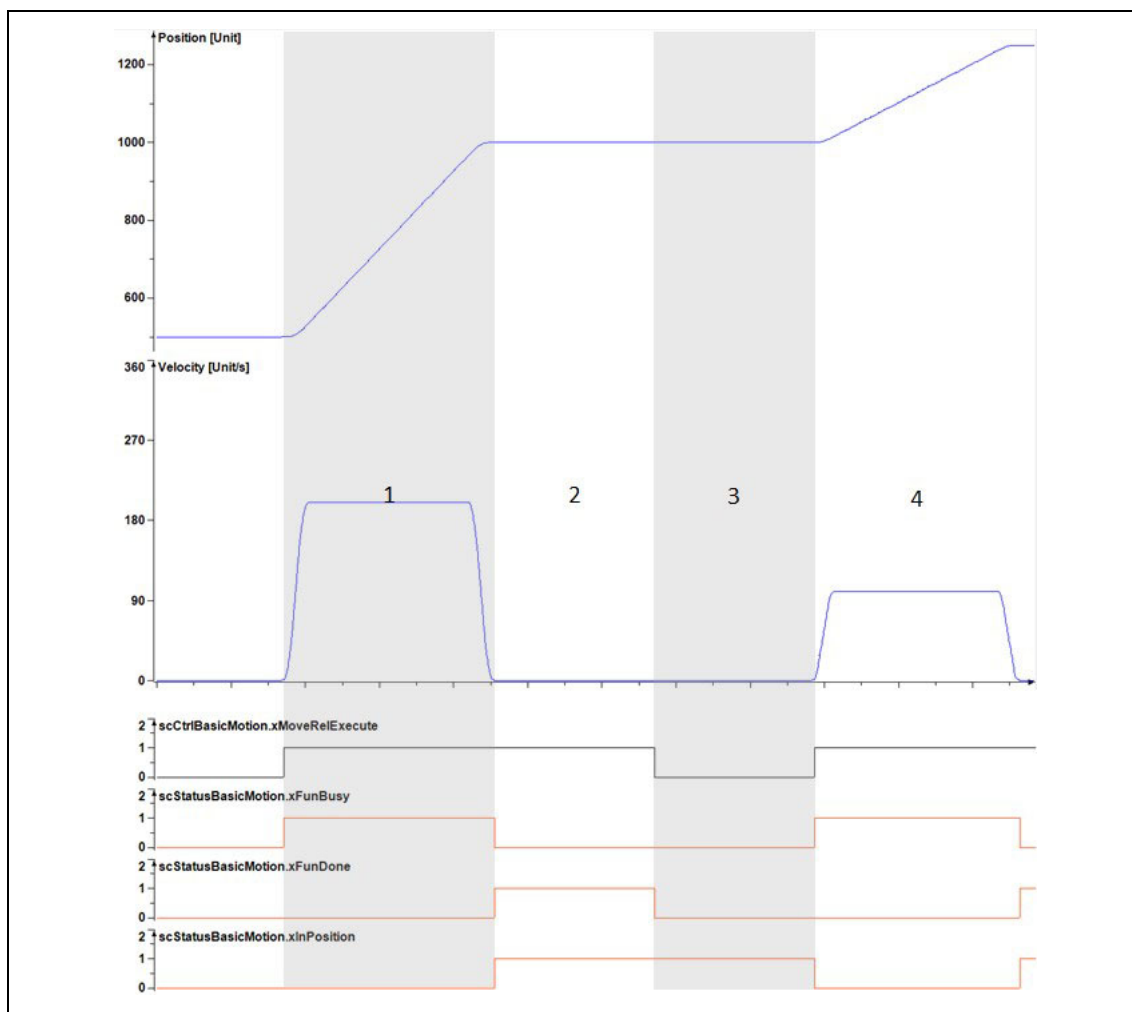
In order to reach the target position, the axis is brought to a standstill via the *lrDec* parameter. The axis reaches a standstill exactly in the target position.

After reaching the target position, the state changes back to STANDBY (*xMoveRelExecute* = FALSE).

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

Positioning can be interrupted via *xAbort*, *xHalt* or *xQSPApplication*. Via *xFinishPositioning*, positioning can be continued again and completed.

Travel diagram



No	xMoveRelExecute	xFuncBusy	xFuncDone	xInPosition	eTMState	Info
1	FALSE ↗ TRUE	FALSE ↗ TRUE	FALSE	FALSE	STANDBY → MOVEREL	The axis starts the positioning run to the relative target position 500 units.
2	TRUE	TRUE ↘ FALSE	FALSE ↗ TRUE	FALSE ↗ TRUE	MOVEREL → STANDBY	As soon as the relative target position has been reached, <i>xInPosition</i> and <i>xFuncDone</i> are set to TRUE.
3	FALSE ↗ TRUE	FALSE-	TRUE ↘ FALSE	TRUE	STANDBY	When <i>xMoveRelExecute</i> is set = FALSE, <i>xFuncDone</i> changes from TRUE ↘ FALSE.
4	FALSE ↗ TRUE	FALSE ↗ TRUE ↘ FALSE	TRUE ↘ FALSE ↗ TRUE	TRUE ↘ FALSE ↗ TRUE	STANDBY → MOVEREL → STANDBY	The axis starts the positioning run to the relative target position 250 units. After reaching the relative target position, <i>xInPosition</i> and <i>xFuncDone</i> are set to TRUE.

Parameters to be set

- *IrVel*, initial value: 3600 [units/s]
- *IrAcc*, initial value: 36000 [units/s²]
- *IrDcc*, initial value: 36000 [units/s²]
- *IrJerk*, initial value: 0 [units/s³]

Further parameters for relative positioning can be found in the AXIS_REF in subindex 130 ... 137.

8.7 Continuous travel with a defined velocity

Prerequisite

- The axis is enabled (*xReadyForMotion* = 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 "MOVEVEL" state.

The axis travels until a software limit position has been reached or a hardware limit switch has been approached:

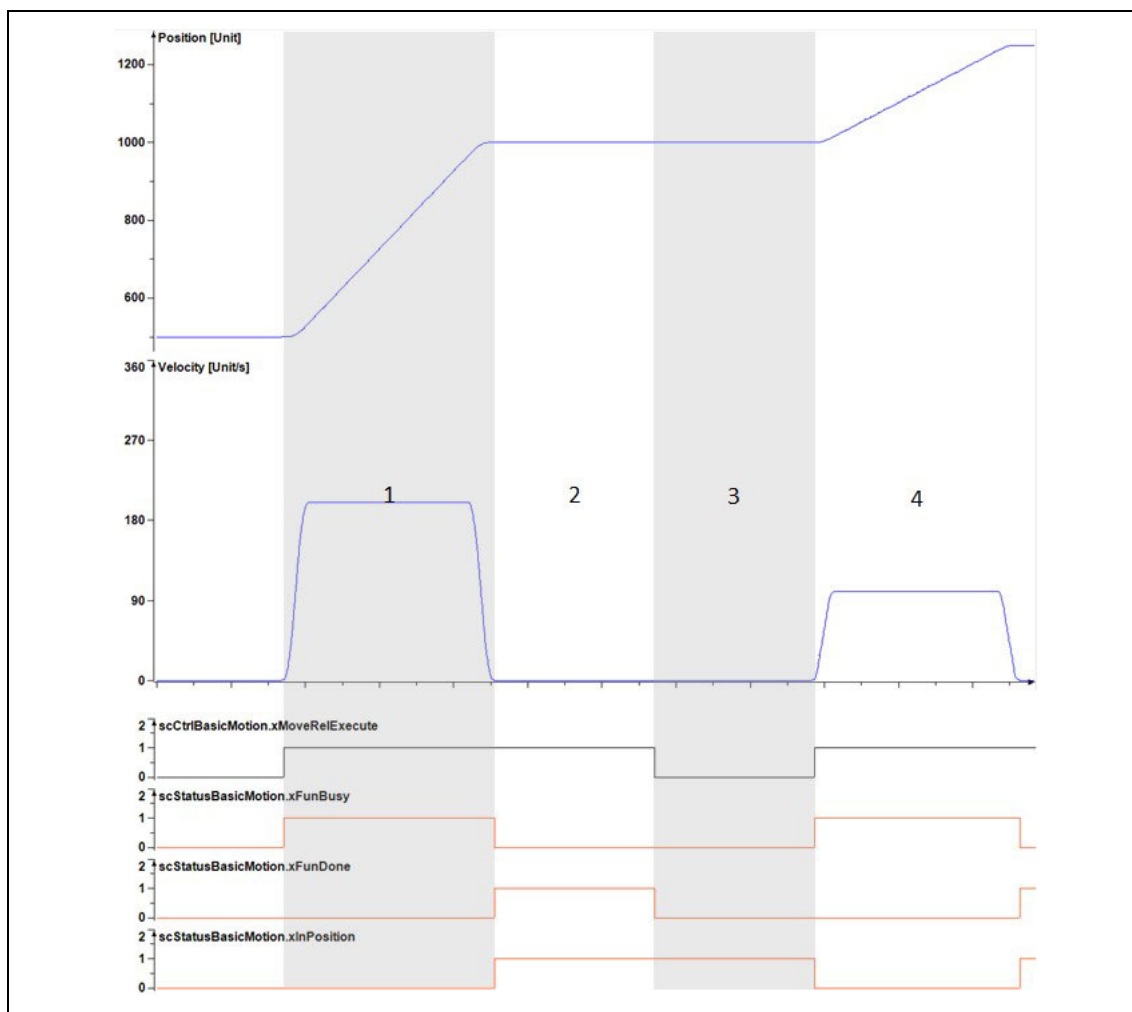
- Input *xDisableSWLimit* = FALSE
- Output *xSwLimitEnabled* = TRUE
- Output *xSwLimitSwitchPos/Neg* = TRUE
- Output *xHwLimitSwitchPos/Neg* = TRUE

When a software limit position has been reached, the axis is brought to a standstill via the *IrDec* parameter.

When a hardware limit switch has been reached, the axis is brought to a standstill via the QSP application deceleration ramp.

Then the state is changed back to "STANDBY".

Travel diagram



No	xMoveVelExecute	xAbsort	xFuncBusy	xFuncDone	xConstantVelocity	eTMState	Info
1	FALSE ↗ TRUE ↘ FALSE	FALSE	FALSE ↗ TRUE	FALSE	FALSE	STANDBY → MOVEVEL	The axis accelerates to the target speed of 200 units/s.
2	FALSE	FALSE	TRUE	FALSE ↗ TRUE	FALSE ↗ TRUE	MOVEVEL	After reaching the target speed, xFuncDone changes from FALSE ↗ TRUE and xConstantVelocity changes from FALSE ↗ TRUE.
3	FALSE	FALSE ↗ TRUE	TRUE ↘ FALSE	TRUE ↘ FALSE ↗ TRUE	TRUE ↘ FALSE	MOVEVEL → STANDBY	The speed run is terminated by xAbort = TRUE and brought to a standstill with the profile deceleration. When standstill has been reached, xFuncBusy is set = FALSE.

Parameters to be set

- *IrVel*, initial value: 3600 [units/s]
- *IrAcc*, initial value: 36000 [units/s²]
- *IrDcc*, initial value: 36000 [units/s²]
- *IrJerk*, initial value: 0 [units/s³]

Further parameters for continuous travel can be found in the AXIS_REF in subindex 50 ... 53 and subindex 104 ... 105.

8.8 Aborting and completing a positioning process

Prerequisite

- The axis is enabled ($xReadyForMotion = \text{TRUE}$).

Execution

An active positioning process is aborted with a rising edge ($\text{FALSE} \rightarrow \text{TRUE}$) at the $xAbort$ input.

The axis travels to a standstill with the deceleration ramp $scPar.IrDec$.

- Exception: **TM SpeedControl** and **TM TablePositioning**

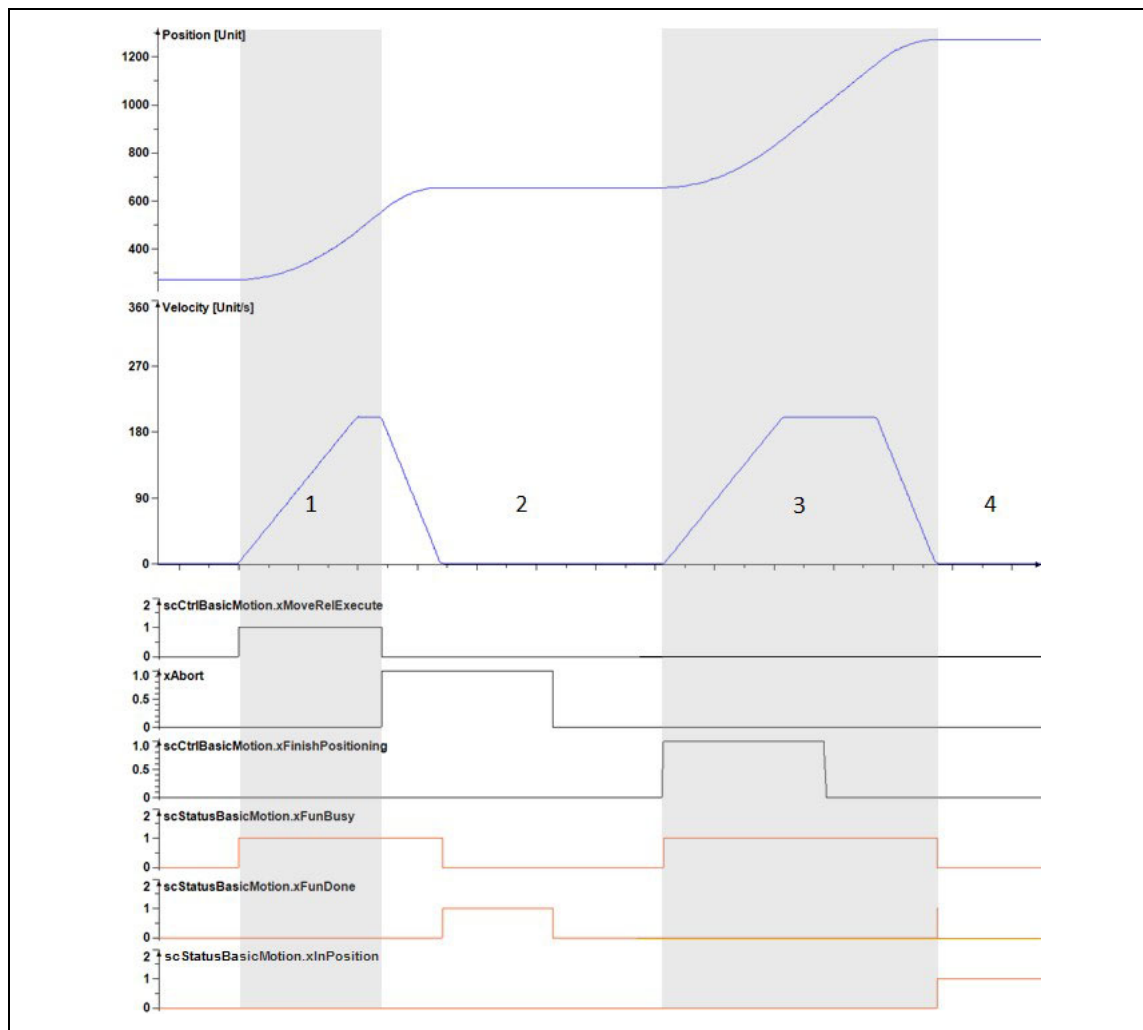
Both technology modules provide several profile data sets, one of which can be active.

If one of these profile data sets is active, the axis moves to standstill with the deceleration ramp of this active profile data set.

If no profile data set is active, the default deceleration ramp $scPar.IrDec$ is used.

The positioning process is completed with a rising edge ($\text{FALSE} \rightarrow \text{TRUE}$) at the $xFinishPositioning$ input.

Travel diagram



No	xMoveRel Execute	xAbort	xFinish Positioning	xInPosition	eTMState	Info
1	FALSE ↗ TRUE	FALSE	FALSE	FALSE	STANDBY→ MOVEREL	The axis starts the positioning process with the relative target position 1000 units.
2	TRUE↘ FALSE	TRUE↘ FALSE	FALSE	FALSE	MOVEREL→ STOP→ STANDBY	The positioning process is aborted with <i>xAbsort</i> = TRUE. The axis is brought to a standstill with the profile deceleration.
3	FALSE	FALSE ↗ TRUE	FALSE↗ TRUE↘ FALSE	FALSE	STANDBY→ MOVEREL	With <i>xFinishPositioning</i> = TRUE, the positioning process is continued.
4	FALSE	FALSE	FALSE	FALSE ↗ TRUE	MOVEREL→ STANDBY	As soon as the relative target position has been reached, <i>xInPosition</i> and <i>xFunDone</i> are set to TRUE.
	FALSE ↗ TRUE	FALSE↗ TRUE↘ FALSE	TRUE↘ FALSE↗ TRUE	TRUE↘ FALSE↗ TRUE	STANDBY→ MOVEREL→ STANDBY	The axis starts the positioning process with the relative target position 250 units. After reaching the relative target position, <i>sxInPosition</i> and <i>xFunDone</i> are set to TRUE.

8.9

Override for speed

Prerequisite

- The axis is enabled ($xReadyForMotion = \text{TRUE}$).

Execution

In order to be able to set overrides for the active profile data, the following inputs are available at the technology module:

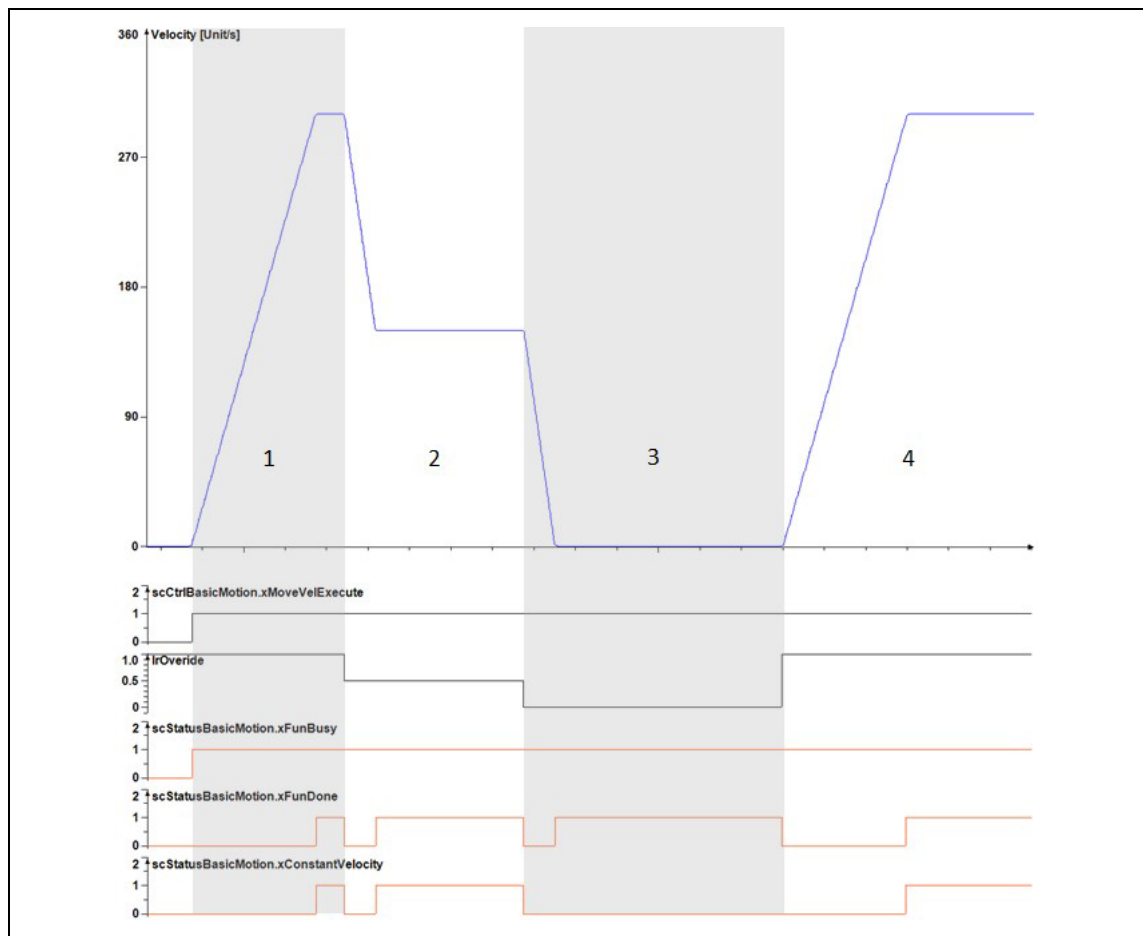
- *IrOverride*, initial value: 1.0
- *scCtrlBasicMotion.IrOverrideAcc*, initial value: 1.0 (100 %)
- *scCtrlBasicMotion.IrOverrideJerk*, initial value: 1.0 (100 %)

For profile-generating functions, separate overrides for speed, acceleration, deceleration and jerk can be set.

The values set act as a factor and are multiplied with the value active in each case.

Example

In this example, the use of the *IrOverride* input is shown.



No	xMoveVel Execute	IrOverride	xFuncBusy	xFunDone	xConstant Velocity	eTMState	Info
1	FALSE ↗ TRUE	1	FALSE ↗ TRUE	FALSE ↗ TRUE	FALSE ↗ TRUE	STANDBY→ MOVEVEL	The axis accelerates to the target speed of 300 units/s.
2	TRUE	0.5	TRUE	TRUE↘ FALSE↗ TRUE	TRUE↘ FALSE↗ TRUE	MOVEVEL	The override value is set to 0.5 and the axis decelerates to 150 unit/s with the profile deceleration
3	TRUE	0	TRUE	TRUE↘ FALSE↗ TRUE	TRUE↘ FALSE	MOVEVEL	The override value is set to 0 and the axis is brought to a standstill to 0 unit/s with the profile deceleration
4	TRUE	1	TRUE	TRUE↘ FALSE↗ TRUE	FALSE ↗ TRUE	MOVEVEL	The override value is set to 1 and the axis accelerates to 300 unit/s with the profile deceleration

9 Description of the technology functions

9.1 Speed synchronism clutching-in/declutching mechanism

9 Description of the technology functions

9.1 Speed synchronism clutching-in/declutching mechanism

9.1.1 Set gearbox factor

For synchronising the slave axis and master axis, the slave velocity is directly coupled to the master velocity by means of the coupling factor.

Execution

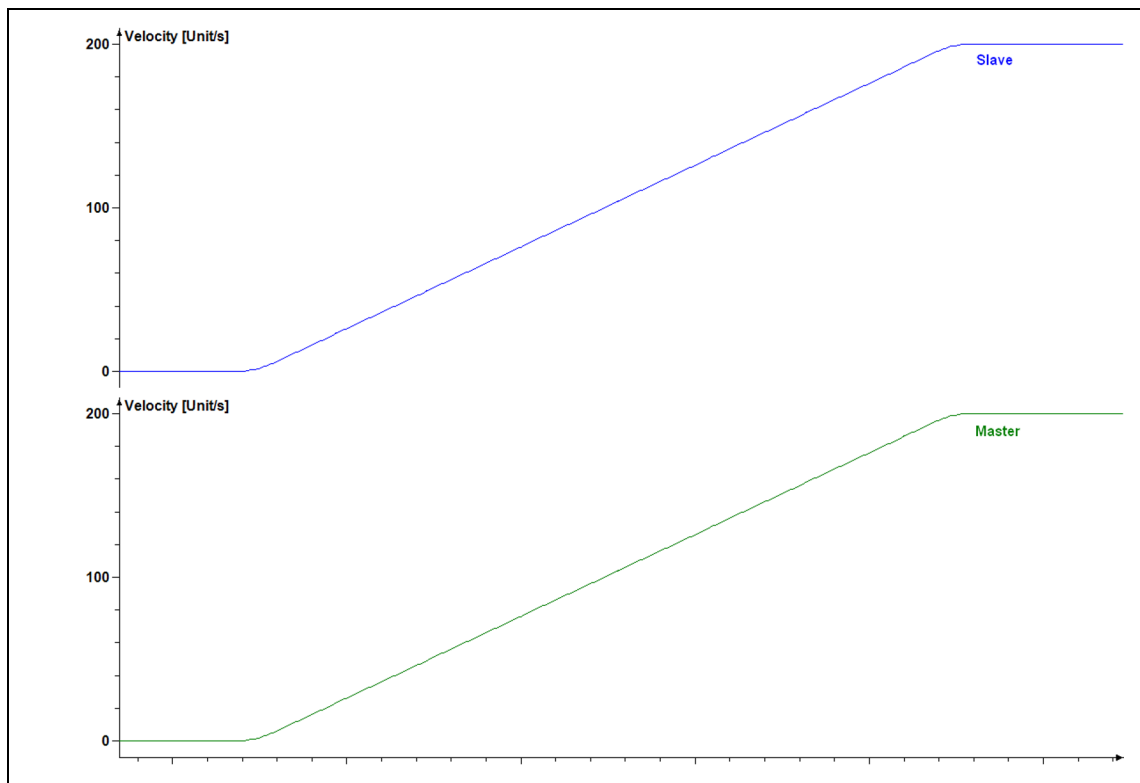
The coupling factor can either be defined by unit coupling or set cycle lengths.

- *scPar.dwCouplingMode* = **0 (unit coupling)**
Coupling factor = $\text{scPar.dwGearingNumerator} / \text{scPar.dwGearingDenominator}$
- *scPar.dwCouplingMode* = **1 (coupling via cycle lengths)**
Coupling factor = cycle length of the slave axis / cycle length of the master axis
The cycle lengths are set in the axes.

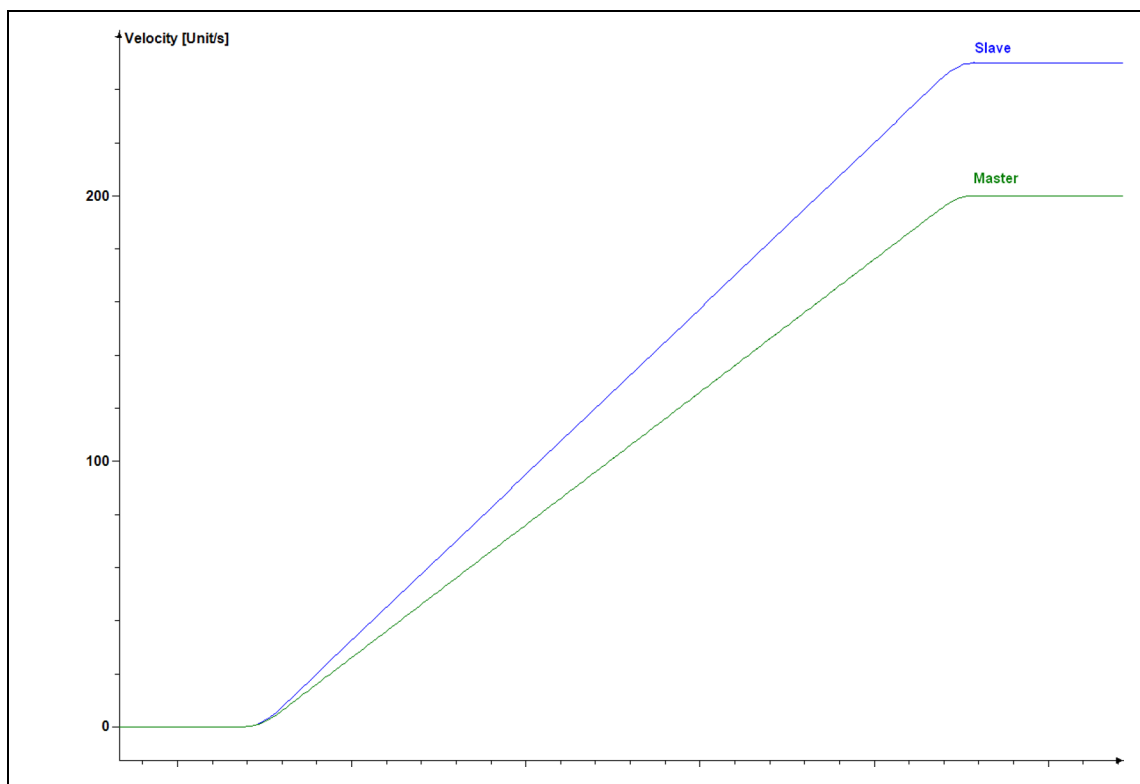
Parameters relevant for the function

- *scPar.dwCouplingMode*, initial value: 1 (coupling via cycle lengths)
- *scPar.dwGearingNumerator*, initial value: 1
- *scPar.dwGearingDenominator*, initial value: 1

Examples



[9-1] Synchronism with identical cycle lengths (360/360) at master speed = 200 unit/s



[9-2] Synchronism with different cycle lengths (450/360) at master speed = 200 unit/s

9.1.2 Invert gearbox factor

The set gearbox factor (coupling factor) can be inverted.

► [Set gearbox factor](#) (📖 47)

DANGER!

The inversion of the gearbox factor takes place with a jump in speed!

Execution

To invert the gearbox factor, the input *xInvertGearfactor* must be set = TRUE.

Precondition for inversion in the "ACC_DEC_SYNC" and "SYNCHRONISED" states

The Inversion in these states is only possible if the target speed of the axis is within a parameterisable speed window (*IrToleranceInvertGearfactor*). If the set speed of the axis is outside of the mentioned states, the inversion is inhibited and a warning is issued (*eErrorID* = 17191).

In all other states, inversion is possible at any time.

Functionally relevant inputs

- *xInvertGearfactor*, Initial value: FALSE

Functionally relevant parameters

- *IrToleranceInvertGearfactor*, Initial value: 0

9.1.3 Clutching-in/declutching

The clutch function synchronizes the speed of the slave axis to the speed of the master axis without a step change.

Preconditions

- The technology module is in an error-free state.
- The drive is enabled (*xReadyForMotion* = TRUE).

Execution

For the clutch-in/declutch process, the acceleration, deceleration and clutch jerk are specified via parameters (*scPar.lrsyncAcc*, *scPar.lrsyncDec*, *scPar.lrsyncJerk*).

The slave axis can start clutching in to the master axis at any position when the *xSyncIn* input is set to TRUE.

During the coupling process, the *xAccDecSync* output = TRUE. If there is synchronicity (clutch closed), the *xSynchronized* output = TRUE.

When declutching by means of *xSyncIn* = FALSE, the slave axis moves to a standstill at the parameterized deceleration *scPar.lrsyncDec* (outputs *xAccDecSync* and *xSynchronized* = FALSE).

Inputs relevant for the function

- *xSyncIn*, initial value: FALSE

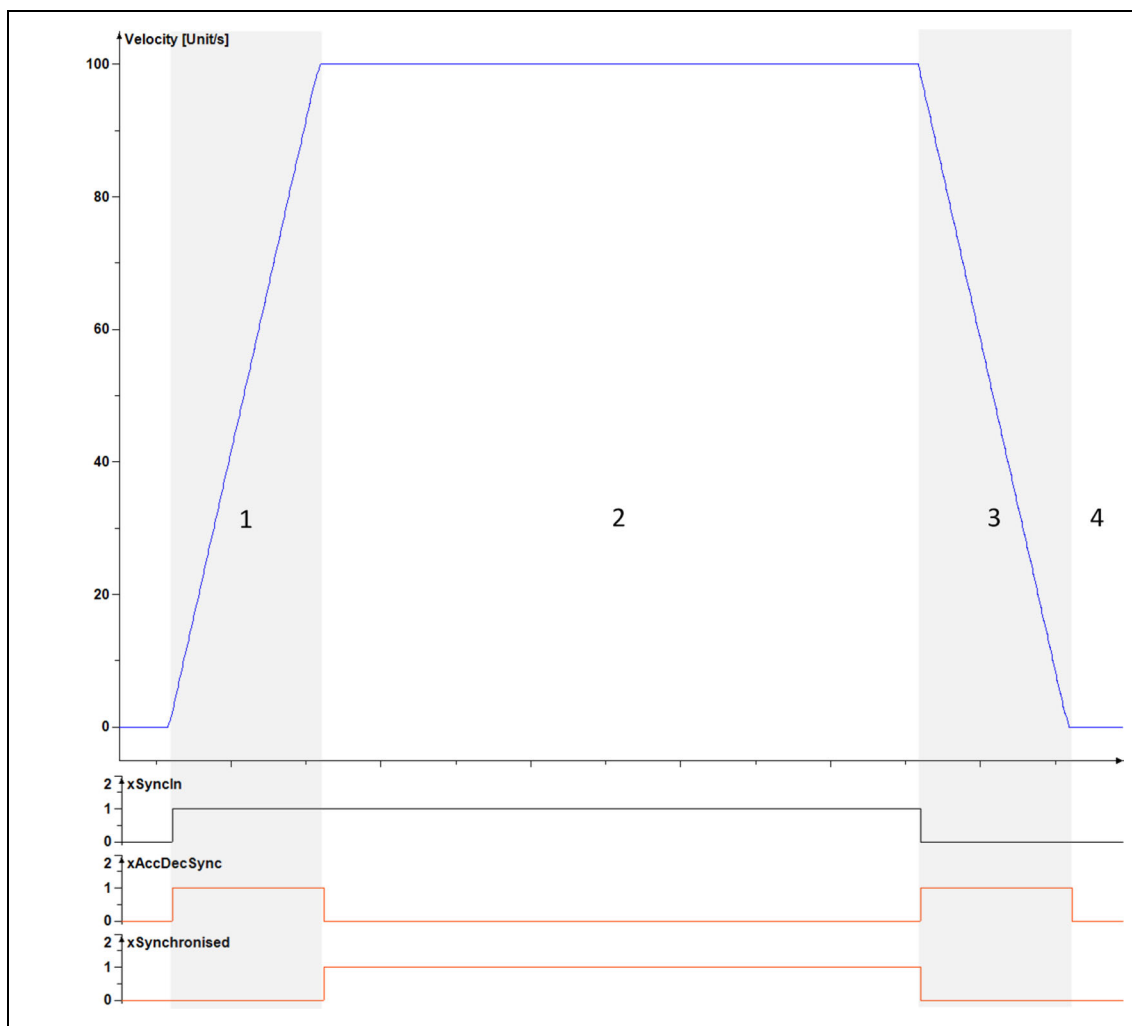
Outputs relevant for the function

- *xAccDecSync*
- *xSynchronised*

Parameters relevant for the function

- *scPar.lrsyncAcc*, initial value: 1 000 unit/s²
- *scPar.lrsyncDcc*, initial value: 1 000 unit/s²
- *scPar.lrsyncJerk*, initial value: 1 000 000 unit/s³

Example



[9-3] Clutch-in and declutch process at slave axis basic speed = 0

No.	xSynIn	xAccDecSync	xSynchronised	eTMState	Info
1	FALSE → TRUE	FALSE → TRUE	FALSE	STANDBY → ACC_DEC_SYNC	The slave axis accelerates up to the speed of the master axis (master value). The coupling to the master value is established.
2	TRUE	TRUE → FALSE	FALSE → TRUE	ACC_DEC_SYNC → SYNCHRONIZED	The slave axis is synchronous with the master axis and follows the master value.
3	TRUE → FALSE	FALSE → TRUE	TRUE → FALSE	SYNCHRONIZED → ACC_DEC_SYNC	The coupling to the master value is canceled and the slave axis decelerates to standstill.
4	FALSE	TRUE → FALSE	FALSE	ACC_DEC_SYNC → STANDBY	Standstill

9.2

Position trimming

9.2.1

Trimming per velocity offset

When trimming per velocity offset, the velocity of the slave axis is adjusted in relation to the master axis.

Preconditions

- The technology module is in an error-free state.
- The drive is enabled (*xReadyForMotion* = TRUE).
- *scPar.eTrimMode* = 0 (trimming per velocity offset)

Execution

For trimming, a profile is specified via the parameters speed, acceleration, deceleration and jerk (*scPar.lfSetOffsetTrimVel*, *scPar.lfSetOffsetTrimAcc*, *scPar.lfSetOffsetTrimDcc*, *scPar.lfSetOffsetTrimJerk*). Each parameter acts in addition to the corresponding master value.

When the *xTrimPos* input = TRUE, the speed of the slave axis is increased in relation to the master axis and when the *xTrimNeg* input = TRUE, its speed is decreased.

During the trimming process, the *xOffsetTrimActive* output = TRUE.

The slave axis is moved with an offset to the master axis while the respective input remains set to TRUE. When *xTrimPos* or *xTrimNeg* = FALSE, the slave axis decelerates until the velocity offset to the master axis is equal to zero.

NOTICE

- A running trimming command cannot be released by another trimming command.
- If both inputs = TRUE, the axis decelerates to standstill and the *eTMState* status output changes to the "ERROR" state. The "ERROR" state is exited if both inputs = FALSE.

Inputs relevant for the function

- *xTrimPos*, *xTrimNeg*

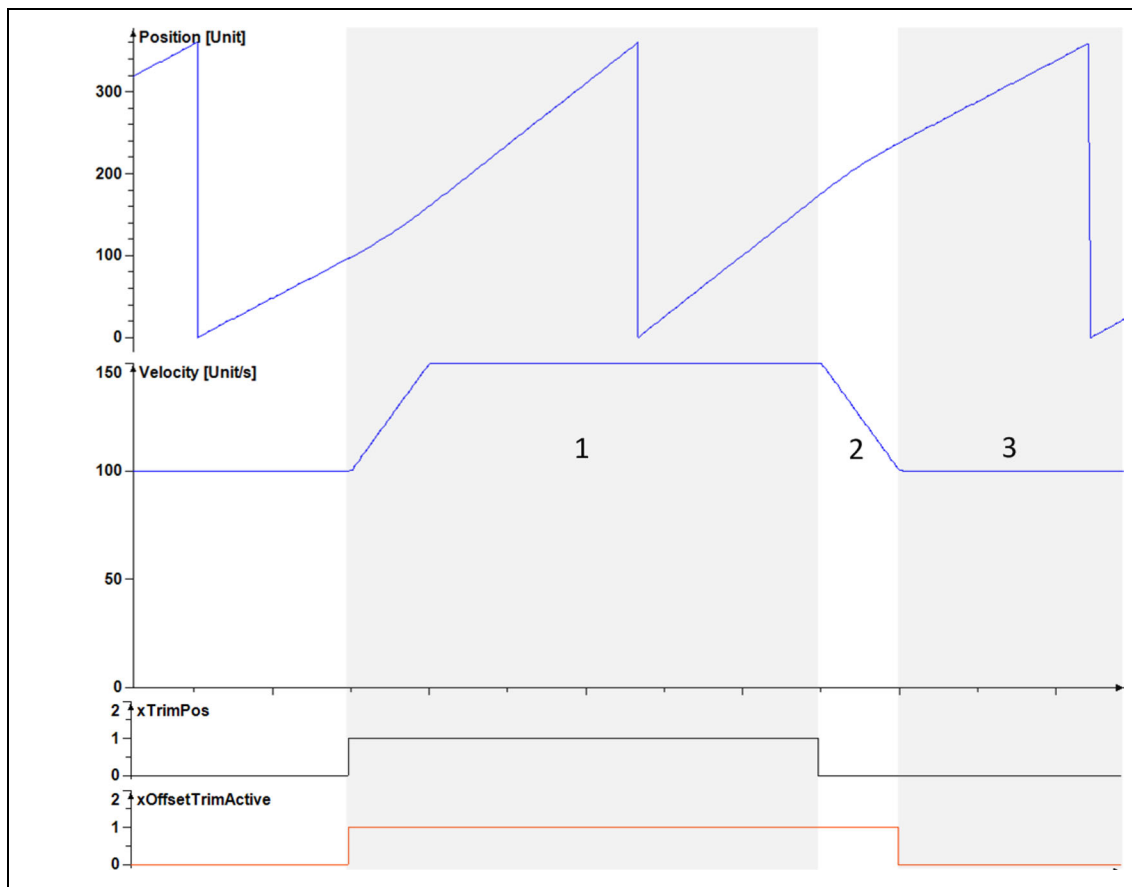
Outputs relevant for the function

- *xOffsetTrimActive*

Parameters relevant for the function

- *scPar.eTrimMode*, initial value: 1 (trimming per position offset)
- *scPar.lfSetOffsetTrimVel*, initial value: 10 unit/s
- *scPar.lfSetOffsetTrimAcc*, initial value: 100 unit/s²
- *scPar.lfSetOffsetTrimDcc*, initial value: 100 unit/s²
- *scPar.lfSetOffsetTrimJerk*, initial value: 10 000 unit/s³

Example



[9-4] Single cycle operation

No.	xTrimPos	xOffsetTrimActive	eTMState	Info
1	FALSE $\rightarrow$ TRUE	FALSE $\rightarrow$ TRUE	SYNCHRONIZED	Trimming per velocity offset is activated. The slave axis accelerates until it reaches a velocity offset to the master axis of 50 unit/s and constantly maintains this offset.
2	TRUE $\rightarrow$ FALSE	TRUE	SYNCHRONIZED	Trimming per velocity offset is deactivated. The slave axis decelerates until the velocity offset to the master axis is equal to zero.
3	FALSE	TRUE $\rightarrow$ FALSE	SYNCHRONIZED	The trimming function is not active. The slave axis moves without offset synchronously to the master speed of the master axis.

9.2.2 Trimming per position offset

When trimming per position offset, the position of the slave axis is adjusted in relation to the master axis.

Preconditions

- The technology module is in an error-free state.
- The drive is enabled (*xReadyForMotion* = TRUE).
- *scPar.eTrimMode* = 1 (trimming per position offset)

Execution

For trimming, a profile is specified via the parameters speed, acceleration, deceleration and jerk (*scPar.lfSetOffsetTrimVel*, *scPar.lfSetOffsetTrimAcc*, *scPar.lfSetOffsetTrimDcc*, *scPar.lfSetOffsetTrimJerk*).

The position difference for trimming is specified in the *scPar.lfTrimDist* parameter.

When the *xTrimPos* input = TRUE, the slave axis is adjusted in the positive direction (+ *scPar.lfTrimDist*) in relation to the master axis and when the *xTrimNeg* input = TRUE, it is adjusted in the negative direction (- *scPar.lfTrimDist*).

The acceleration and velocity of the trimming superimpose the ones of the master axis. Hence, the results for the axis to be trimmed are as follows:

- resulting speed of: $v_{\text{AxisRes}} = v_{\text{MasterAxis}} + \text{scPar.lfSetOffsetTrimVel}$
- resulting acceleration of: $a_{\text{AxisRes}} = a_{\text{MasterAxis}} + \text{scPar.lfSetOffsetTrimAcc}$

During the trimming process, the *xOffsetTrimActive* output = TRUE.

If there is a rising edge at the *xTrimReset* input = FALSE → TRUE, the position offset resulting from the trimming will be moved to zero via the profile generator.

NOTICE

- A running trimming command cannot be released by another trimming command.
- If both inputs = TRUE, the axis decelerates to standstill and the *eTMState* status output changes to the "ERROR" state. The "ERROR" state is exited if both inputs = FALSE.

Inputs relevant for the function

- *xTrimPos*, *xTrimNeg*
- *xTrimReset*

Outputs relevant for the function

- *xOffsetTrimActive*

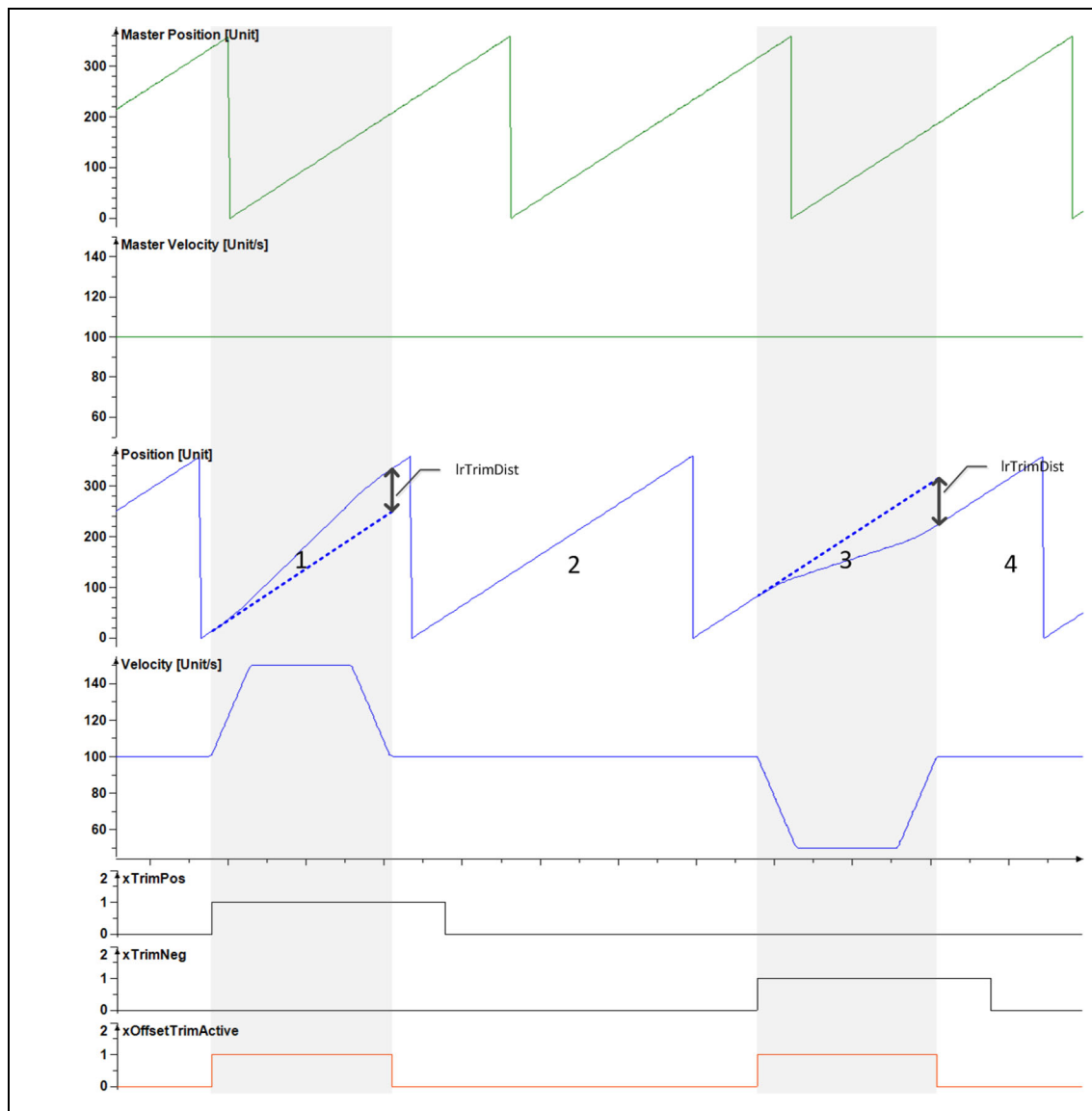
Parameters relevant for the function

- *scPar.eTrimMode*, initial value: 1 (trimming per position offset)
- *scPar.lrTrimDist*, initial value: 1 unit
- *scPar.lrSetOffsetTrimVel*, initial value: 10 unit/s
- *scPar.lrSetOffsetTrimAcc*, initial value: 100 unit/s²
- *scPar.lrSetOffsetTrimDcc*, initial value: 100 unit/s²
- *scPar.lrSetOffsetTrimJerk*, initial value: 10 000 unit/s³

Measuring points relevant for the function

- *scSignalflow.MP_lrPosOffsetTotalVel*

Example



[9-5] Trimming in positive/negative direction by 90 units at a maximum trimming speed of 50 units/s

No.	xTrimPos	xTrimNeg	xOffsetTrimActive	eTMState	Info
1	FALSE $\rightarrow$ TRUE	FALSE	FALSE $\rightarrow$ TRUE	SYNCHRONIZED	Trimming per position offset is activated. The slave axis positions itself in relation to the master axis in the <u>positive</u> direction by <i>scPar.IrTrimDist</i> = 90 units.
2	TRUE $\Rightarrow$ FALSE	FALSE	TRUE $\Rightarrow$ FALSE	SYNCHRONIZED	Synchronous travel
3	FALSE	FALSE $\rightarrow$ TRUE	FALSE $\rightarrow$ TRUE	SYNCHRONIZED	Trimming per position offset is activated. The slave axis positions itself in relation to the master axis in the <u>negative</u> direction by <i>scPar.IrTrimDist</i> = 90 units.
4	FALSE	TRUE $\Rightarrow$ FALSE	TRUE $\Rightarrow$ FALSE	SYNCHRONIZED	Synchronous travel

9.3 Position offset via profile generator

The slave axis is moved with a variable position offset to the master axis. The offset is specified by a profile generator.

Preconditions

- The technology module is in an error-free state.
- The drive is enabled (*xReadyForMotion* = TRUE).

Execution

For positioning, a profile is specified via the parameters speed, acceleration, deceleration and jerk (*scPar.lfSetOffsetTrimVel*, *scPar.lfSetOffsetTrimAcc*, *scPar.lfSetOffsetTrimDcc*, *scPar.lfSetOffsetTrimJerk*).

The *lfSetOffsetPos* input specifies the position offset to the master axis. The *scPar.xLoadOffset* parameter determines whether the position offset is cyclic or loaded via a profile generator.

The allowed direction of rotation of the slave axis is determined by the *eAllowedDirection* parameter:

Setting <i>eAllowedDirection</i>	Direction setting for the clutch and profile generator of the offset and trim functions
0 (both directions)	The slave axis may engage in <u>both</u> directions. An offset or trim may be moved in both directions.
1 (positive direction)	The slave axis may only engage in positive direction. An offset may only be moved in positive direction. A negative offset is only moved to the extent that the speed of the total conductance does not become less than '0'.
2 (negative direction)	The slave axis may only engage in negative direction. An offset may only be moved in negative direction. A positive offset is only moved to the extent that the speed of the total conductance does not exceed '0'.

If the offset function is active, the *xOffsetTrimActive* output = TRUE.

Inputs relevant for the function

- *lfSetOffsetPos*, initial value: 0 units

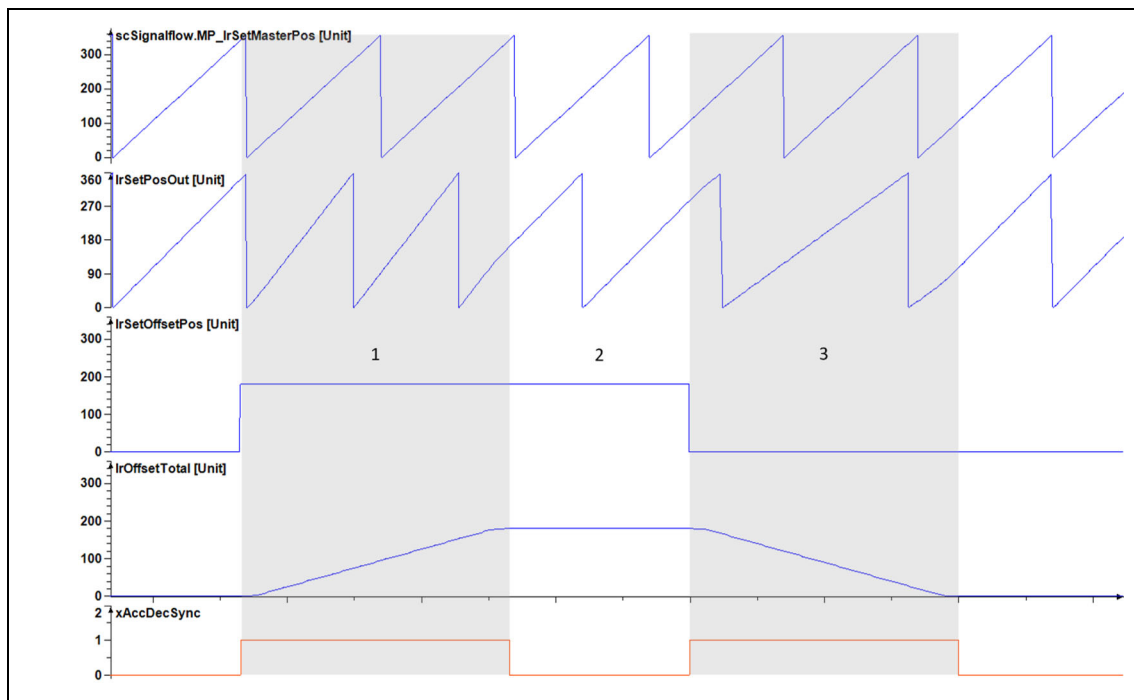
Outputs relevant for the function

- *lfOffsetTotalVel*
- *xOffsetTrimActive*

Parameters relevant for the function

- *scPar.xLoadOffset*, initial value: FALSE
- *eAllowedDirection*, initial value: 0 (both directions)
- *scPar.lfSetOffsetTrimVel*, initial value: 10 [unit/s]
- *scPar.lfSetOffsetTrimAcc*, initial value: 100 [unit/s²]
- *scPar.lfSetOffsetTrimDcc*, initial value: 100 [unit/s²]
- *scPar.lfSetOffsetTrimJerk*, initial value: 10 000 [unit/s³]

Example



[9-6] Single cycle operation

No.	IrSetOffsetPos	IrOffsetTotal	xOffsetTrimActive	Info
1	0 units → 200 units	0 units	FALSE → TRUE	The offset is corrected to the target value <i>IrSetOffsetPos</i> = 200 units via a profile generator.
2	200 units	200 units	TRUE → FALSE	The offset is corrected.
3	200 units → 0 units	200 units	FALSE → TRUE	The offset is corrected to the target value <i>IrSetOffsetPos</i> = 0 units via a profile generator.

9.4 Velocity offset via profile generator

The slave axis is moved with a variable velocity offset to the master axis. The offset is specified by a profile generator.

Preconditions

- The technology module is in an error-free state.
- The drive is enabled (*xReadyForMotion* = TRUE).

Execution

For positioning, a profile is specified via the parameters speed, acceleration, deceleration and jerk (*scPar.IrSetOffsetTrimVel*, *scPar.IrSetOffsetTrimAcc*, *scPar.IrSetOffsetTrimDcc*, *scPar.IrSetOffsetTrimJerk*).

The *IrSetOffsetVel* input specifies the velocity offset to the master axis. It is transmitted to the axis without speed step change by means of S-rounding via a profile generator and can be read at the *scSignalflow.MP_IrVelOffset* measuring point.

If the offset function is active, the *xOffsetTrimActive* output = TRUE.

Inputs relevant for the function

- *IrSetOffsetVel*, initial value: 0 units/s

Outputs relevant for the function

- *IrOffsetTotalVel*
- *xOffsetTrimActive*

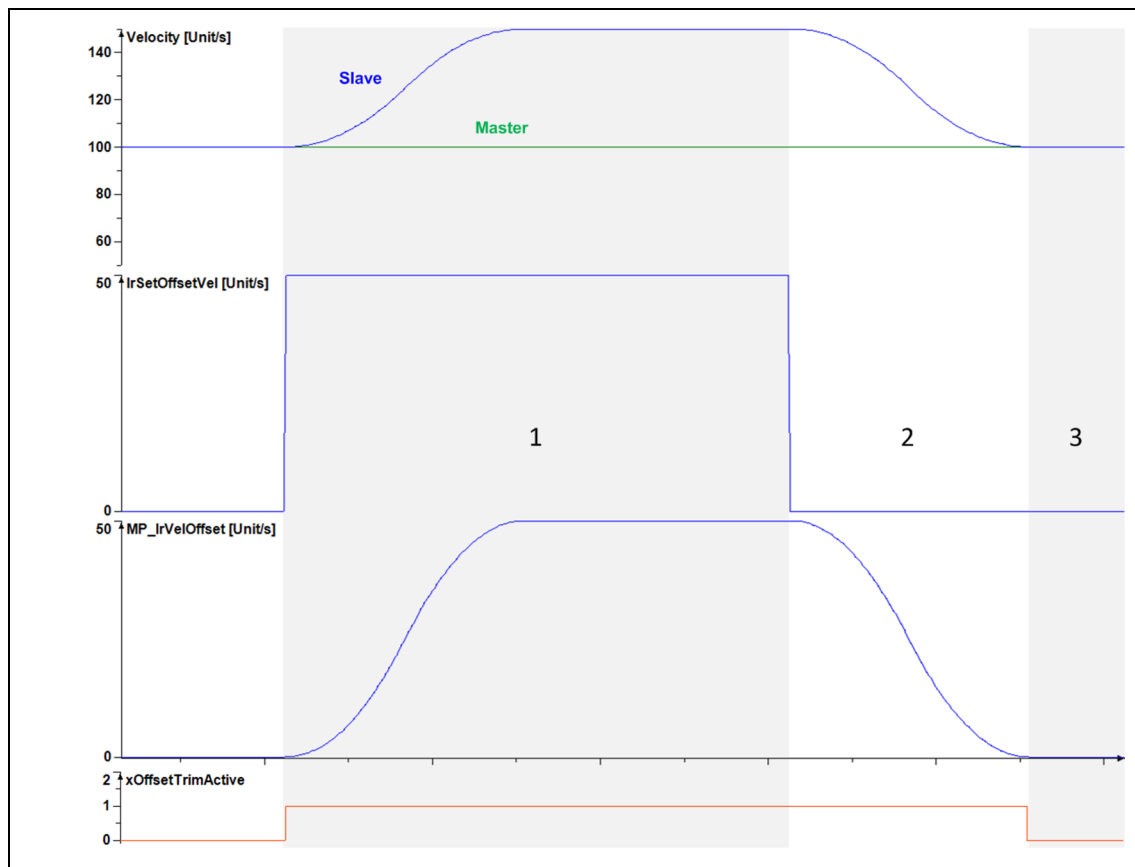
Parameters relevant for the function

- *scPar.IrSetOffsetTrimAcc*, initial value: 100 unit/s²
- *scPar.IrSetOffsetTrimDcc*, initial value: 100 unit/s²
- *scPar.IrSetOffsetTrimJerk*, initial value: 10 000 unit/s³

Measuring points relevant for the function

- *scSignalflow.MP_IrVelOffset*, initial value: 0 units/s

Example



[9-7] Velocity offset ($IrSetOffsetVel = 50 \text{ units/s}$) by means of S-rounding ($IrSetOffsetTrimJerk = 100 \text{ units/s}^3$)

No.	IrSetOffsetVel	MP_IrVelOffset	xOffsetTrimActive	eTMState	Info
1	0 units → 50 units/s		FALSE → TRUE	SYNCHRONIZED	A velocity offset of 50 unit/s is set at the <i>IrSetOffsetVel</i> input. The slave axis accelerates and then moves at a constant velocity offset of 50 unit/s in relation to the master speed of the master axis.
2	50 units → 0 units/s		TRUE	SYNCHRONIZED	The velocity offset is set equal to zero at the <i>IrSetOffsetVel</i> input. The slave axis decelerates until it reaches the master speed of the master and the velocity offset is equal to zero.
3	0 units/s		TRUE → FALSE	SYNCHRONIZED	The slave axis moves without velocity offset synchronously to the master axis.

9.5 Synchronism with basic speed

Starting from a basic speed, the clutch function synchronizes the speed of the slave axis to the speed of the master axis without a step change.

Preconditions

- The technology module is in an error-free state.
- The drive is enabled (*xReadyForMotion* = TRUE).
- The slave and master axes are uncoupled (*xSyncIn* = FALSE).

Execution

Synchronism with basic speed is active as soon as the *IrSetBaseVel* input takes on a non-zero value.

Immediately after the controller has been enabled, the slave axis moves by means of S-rounding at the basic speed specified as per the *IrSetBaseVel* input. The technology module changes to the "BASEVELOCITY" state. From this state, the speed of the slave axis can be synchronized to the master speed of the master axis via the clutch function.

For the clutch-in/declutch process, the acceleration, deceleration and clutch jerk are specified via parameters (*scPar.IrSyncAcc*, *scPar.IrSyncDec*, *scPar.IrSyncJerk*).

Clutch-in starts at any position when the *xSyncIn* input is set to TRUE.

During the coupling process, the *xAccDecSync* output = TRUE. If there is synchronicity (clutch closed), the *xSynchronised* output = TRUE.

When declutching by means of *xSyncIn* = FALSE, the slave axis decelerates to the basic speed *IrSetBaseVel* as per the *scPar.IrSyncDec* parameter. When the basic speed is activated, the *xBaseVelocityActive* output is set to TRUE.

Inputs relevant for the function

- *IrSetBaseVel*, initial value: 0 unit/s
- *xSyncIn*, initial value: FALSE

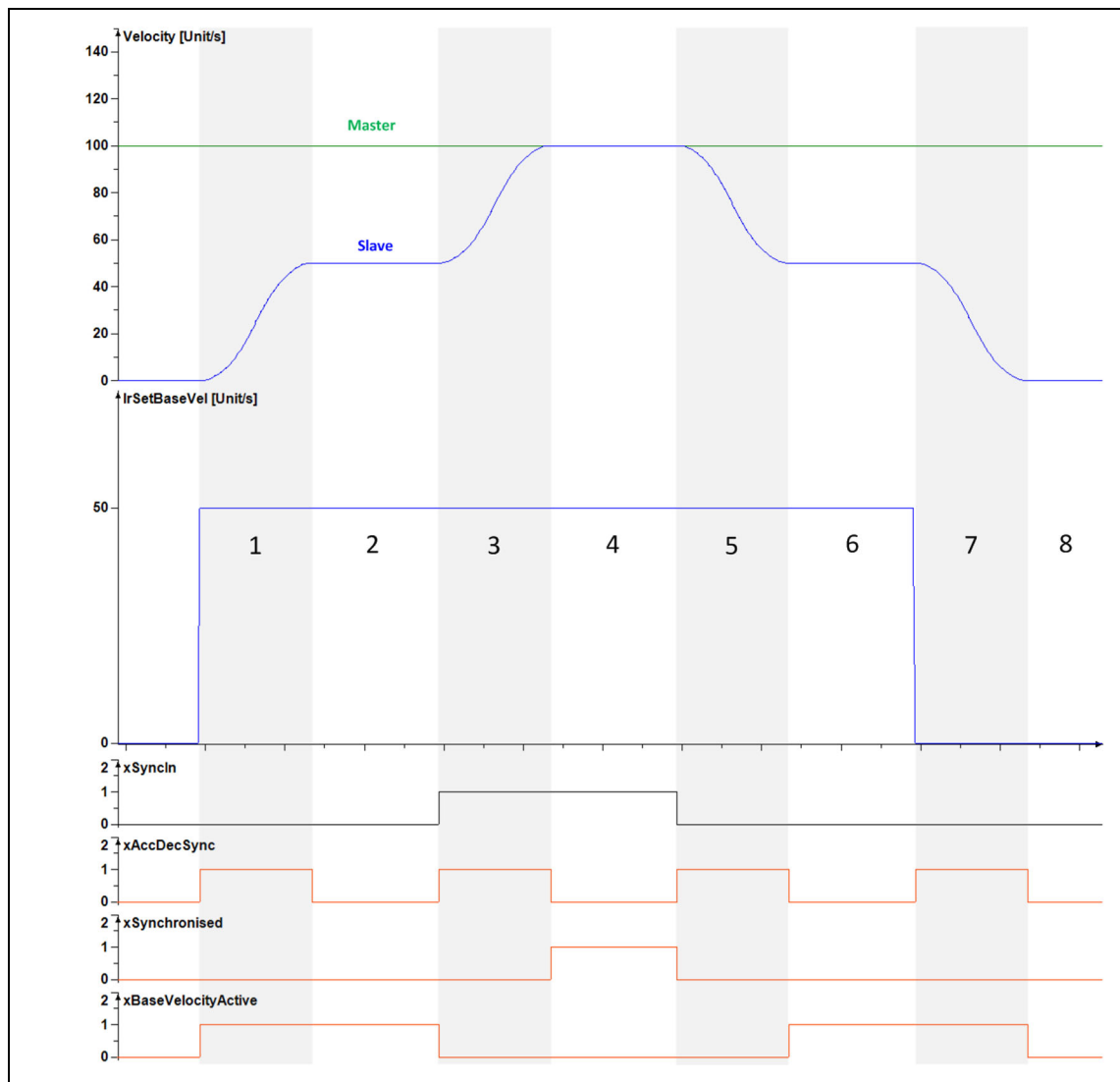
Outputs relevant for the function

- *xAccDecSync*
- *xSynchronised*
- *xBaseVelocityActive*

Parameters relevant for the function

- *scPar.IrSyncAcc*, initial value: 1 000 unit/s²
- *scPar.IrSyncDcc*, initial value: 1 000 unit/s²
- *scPar.IrSyncJerk*, initial value: 1 000 000 unit/s³

Example

[9-8] Clutch-in and declutch process with slave axis basic speed ($IrSetBaseVel = 50$)

No.	xSyncln	xAccDecSync	xSynchronised	xBaseVelocityActive	eTMState	Info
1	FALSE	FALSE→TRUE	FALSE	FALSE→TRUE	STANDBY → ACC_DEC_SYNC	The slave axis accelerates from standstill to a basic speed of 50 unit/s.
2	FALSE	TRUE↔FALSE	FALSE	TRUE	ACC_DEC_SYNC → BASEVELOCITY	The slave axis moves at a constant basic speed.
3	FALSE→TRUE	FALSE→TRUE	FALSE	TRUE↔FALSE	BASEVELOCITY → ACC_DEC_SYNC	The axis synchronizes to the speed of the master axis.
4	TRUE↔FALSE	TRUE↔FALSE	FALSE→TRUE	FALSE	ACC_DEC_SYNC → SYNCHRONIZED	The axis moves synchronously to the master axis.
5	TRUE	FALSE→TRUE	TRUE↔FALSE	FALSE	SYNCHRONIZED → ACC_DEC_SYNC	The coupling to the master axis is released and the slave axis decelerates to the basic speed.
6	FALSE	TRUE↔FALSE	FALSE	FALSE→TRUE	ACC_DEC_SYNC → BASEVELOCITY	The slave axis moves at a constant basic speed.
7	FALSE	FALSE→TRUE	FALSE	TRUE	BASEVELOCITY → ACC_DEC_SYNC	The slave axis decelerates to standstill.
8	FALSE	TRUE↔FALSE	FALSE	TRUE↔FALSE	ACC_DEC_SYNC → STANDBY	Standstill

9.6 Limit setpoints to allowed travel directions

In some applications, the tool axis must not move in a certain direction. In this case, the `eAllowedDirection` parameter can be used to block profile generation in the unauthorized direction. The setting prevents an impermissible position offset or engagement.

WARNING!

Injury to persons and damage or destruction of machine parts

Damage or destruction of machine parts can be caused by unpredictable motor movements.

Setting the `eAllowedDirection` parameter does not prevent the total setpoint of the slave axis from acting in an inhibited direction. This could be the case, for example, if the master speed or a speed offset acts in the blocked direction. These setpoint components are not actively limited by this function.

Possible consequences

Death or severe injuries

Protective measures

Check and adjust limits and direction of action for the master speed or speed offset.

To limit the total setpoint of a tool axis to the allowed range, the parameter `xLimitAllowedDirection` = TRUE must be set. In this way, only setpoints that act in the allowed direction are transferred to the slave axis.

The parameter `IrToleranceAllowedDirection` can be used to specify a tolerance range. If the setpoint moves in a blocked direction, the setpoint of the tool axis is "frozen" from that moment.

If the total setpoint then changes in the blocked direction and the tolerance range is violated, the axis changes to the "FailQSP" state.

If this does not happen and the setpoint leaves the tolerance window in the allowed direction, it is passed back to the tool axis.

Parameters relevant for the function

- `eAllowedDirection`, initial value: 0 (positive and negative rotating direction)
- `xLimitAllowedDirection`, initial value: FALSE
- `IrToleranceAllowedDirection`, initial value: 5 units

Symbols

"Electronic Gearbox Base" functional description [10](#)

"Electronic Gearbox Base" TM parameters [19](#)

A

Aborting a positioning process [43](#)

Absolute positioning [35](#)

Access points [26](#)

Application notes [7](#)

C

Competing a positioning process [43](#)

Continuous travel with a defined velocity [40](#)

Conventions used [6](#)

D

Document history [5](#)

E

Electronic Gearbox Base (functional description) [10](#)

E-mail to Lenze [65](#)

eTMState [23](#)

F

Feedback to Lenze [65](#)

G

Getriebefaktor invertieren [49](#)

H

Halt, [34](#)

Homing [30](#)

I

inputs [15](#)

Invertieren des Getriebefaktors [49](#)

L

L_TB2P_States [23](#)

L_TF2P_Electronic Gearbox Base function block [14](#)

L_TS2P_ElectronicGearboxBase [14](#)

L_TS2P_scAP_ElectronicGearboxBase [26](#)

L_TS2P_scPar_ElectronicGearboxBase [19](#)

L_TS2P_scPar_ElectronicGearboxBase parameter structure [19](#)

L_TS2P_scSF_ElectronicGearboxBase [25](#)

Layout of the safety instructions [7](#)

Limit setpoints to permitted travel range [63](#)

M

Manual jog (jogging) [28](#)

N

Notes on how to operate the technology module [12](#)

O

outputs [17](#)

Override for speed [45](#)

Q

QSP application [33](#)

R

Relative positioning [37](#)

S

Safety instructions [8](#)

Signal flow diagram [24](#)

Signal flow of the "Electrical Gearbox Base" technology module [24](#)

Speed override [45](#)

Start of the axes [12](#)

State machine [22](#)

States [22](#)

States of the eTMState output [23](#)

Structure of the access points

L_TS2P_scAP_ElectronicGearboxBase [26](#)

Structure of the signal flow

L_TS2P_scSF_ElectronicGearboxBase [25](#)

T

Target group [4](#)

Technology module functions (overview) [11](#)

V

Variable identifiers [6](#)



Your opinion is important to us

We have compiled these instructions to the best of our knowledge with the objective of supporting you optimally when you handle 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

