

Technology module

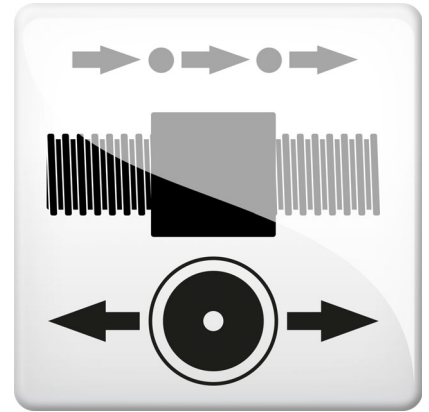


Table Positioning _____

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 overview of the TM	10
3.2	Important notes on how to operate the technology module	11
4	Application ranges	12
5	L_TF2P_TablePositioningBase function block	13
5.1	inputs	14
5.2	outputs	16
5.3	Parameter	18
6	State machine	19
7	Signal flow diagram	20
7.1	Structure of the signal flow	20
7.2	Structure of the access points	20
8	Description of the basic functions	21
8.1	Manual jog (jogging)	22
8.2	Homing	24
8.3	QSP application	27
8.4	Halt	28
8.5	Absolute positioning	29
8.6	Relative positioning	31
8.7	Continuous travel with a defined velocity	34
8.8	Aborting and completing a positioning process	37
8.9	Override for speed	39
9	Description of the technology function	41
9.1	Profile data management	41
9.2	Positioning modes	45
9.3	Perform positioning	47
9.4	Touch probe positioning	49
	Index	51
	Your opinion is important to us	52

1 About this document

This documentation ...

- contains detailed information on the functionalities of the "Table Positioning" 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	04/2020	TD06	Error corrections & supplements
2.0	12/2019	TD06	Error corrections & supplements
1.0	09/2019	TD06	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 overview of the TM

Apart from the basic functions of the L_TB2P_BasicMotionBase technology module, the technology module also offers the following functionalities to move an axis:

- ▶ [Profile data management](#) (📖 41)
- ▶ [Positioning modes](#) (📖 45)
- ▶ [Perform positioning](#) (📖 47)
- ▶ [Touch probe positioning](#) (📖 49)

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

**Tip!**

The basic functions of the technology module are also described in these instructions.

- ▶ [Description of the basic functions](#) (📖 21)

3.2

Important notes on how to operate the technology module

Start of the axes

The start of the axes after a locked controller depends on the type of movement commissioned.

A differentiation is made between the movement type with a level-activated function and the movement type with an edge-activated function.

- The movement type with a level-activated function is activated automatically after the controller release as long as no higher priority function is active. This movement type includes
 - [Manual jog \(jogging\)](#) (□ 22)
 - [QSP application](#) (□ 27)
- The movement type with an edge-activated function must be activated after the controller release with a FALSE→TRUE-edge. This movement type includes
 - [Homing](#) (□ 24)



Example of a movement type with a level-activated function: manual mode (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 of a movement type with an edge-activated function: reference run (homing)

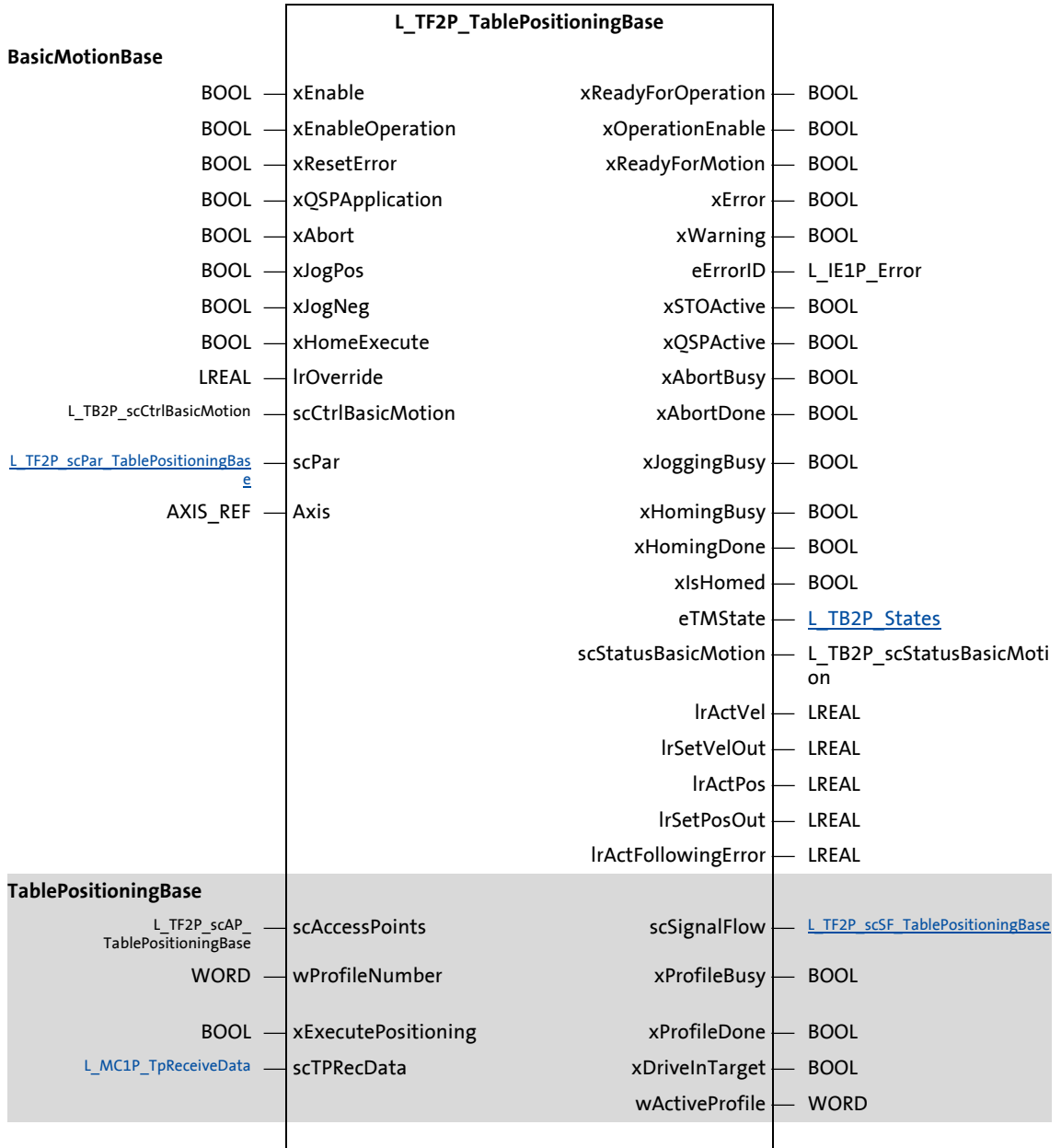
1. In the inhibited axis state ($xOperationEnabled = FALSE$), $xHomeExecute$ is set to TRUE.
 - $xEnableOperation = FALSE$ (axis is inhibited.)
==> **STANDBY** state ($xEnableOperation = FALSE$)
 - $xHomeExecute = TRUE$ (referencing is to be executed.)
2. Enable axis.
 - $xEnableOperation = TRUE$
==>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

4 Application ranges

- Positioning in limited and unlimited travel range (Modulo measuring system)
- Transport devices
- Rotary tables
- Storage and retrieval units
- Feed drives
- Dosers
- Hoists

5 L_TF2P_TablePositioningBase function block

The illustration shows that the inputs and outputs are part of the function block.



5.1

inputs

Identifier	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 the "ready for operation" state. Controller enable is issued.
xResetError	BOOL	TRUE Reset axis error or software error.
xQSPApplication	BOOL	TRUE The active movement is cancelled and the axis is brought to a standstill via the QSP application stop ramp. - The state changes to "Stop". - The technology module remains in the "STOP" status as long as <i>xQSPApplication</i> remains TRUE.
xAbort	BOOL	TRUE The active movement is cancelled and the axis is brought to a standstill via the active profile deceleration. The state changes to "Stop". The deceleration ramp can be aborted by commissioning of a new movement (e.g. <i>xMoveVelExecute</i>)
xJogPos	BOOL	TRUE Traverse axis in positive direction (manual jog). If <i>xJogNeg</i> is also TRUE, the <i>eErrorID</i> output displays the InvalidInputJogPosJogNeg error message and the axis stops.
xJogNeg	BOOL	TRUE Traverse axis in negative direction (manual jog). If <i>xJogPos</i> is also TRUE, the <i>eErrorID</i> output displays the InvalidInputJogPosJogNeg error message 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.
IrOverride	LREAL	With the <i>IrOverride</i> factor, the defined 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: Acceptance of the value is carried out irrespective of the status in <i>xContinuousUpdate</i>
scCtrlBasicMotion L_TB2P_scCtrlBasicMotion		• See description for the L_TB2P_BasicMotionBase technology module
Axis	AXIS_REF	Reference to the axis
scPar L_TF2P_scPar_TablePositioningBase		The parameter structure contains the parameters of the technology module.
scAccessPoints L_TF2P_scAP_TablePositioningBase		Structure of the access points Via the access points, the signal flow or the functions of the technology module can be influenced.
wProfileNumber	WORD	Selection of the active travel profile 15 different travel profiles (1 - 15) can be selected. No profile will be run if "0" is entered. A warning appears when starting

Identifier	Data type	Description
xExecutePositioning	BOOL	The input is edge-controlled and evaluates the rising edge. Initial value: FALSE
	FALSE ↗ TRUE	The travel profile selected at the <i>wProfileNumber</i> input will be executed. Restart by a renewed FALSE↗TRUE edge: During a running positioning process, another profile can be specified via the <i>scPar.ascProfiles</i> profile data structure that will be executed after a restart. ▶ L_TF2P_scPar_TablePositioningBase (18)
scTPRecData L_MC1P_TpReceiveData		Touch probe structure input

5.2

outputs

Identifier	Data type	Description
xReadyForOperation	BOOL	The axis is faultless. The DC bus voltage is available. STO is not active.
xOperationEnabled	BOOL	The axis is enabled. The brake is released.
xReadyForMotion	BOOL	The axis is enabled and faultless. 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 A Safe Torque Off (STO) has been activated and is active. The axis is torqueless, the controller is inhibited.
xQSPActive	BOOL	TRUE The <i>xQSPApplication</i> input has been 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 a standstill.
xAbortBusy	BOOL	TRUE The function activated is aborted by the <i>xAbsort</i> input.
xAbsortDone	BOOL	TRUE The function activated has been aborted by the <i>xAbsort</i> input.
xJoggingBusy	BOOL	TRUE The manual jog function is active. The technology module is in the JOGGING state.
xHomingBusy	BOOL	TRUE Homing is carried out. The technology module is in the HOMING state.
xHomingDone	BOOL	TRUE Homing has been completed successfully.
xIsHomed	BOOL	TRUE The axis has been referenced ("Reference known"). <i>xIsHomed</i> must 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 of the function block
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
scSignalFlow	L_TB2P_scSF_TablePositioningBase	Structure of the signal flow

Identifier	Data type	Description	
xProfileBusy	BOOL	TRUE	A profile is currently being actively executed.
xProfileDone	BOOL	TRUE	A profile has been completed.
xDriveInTarget	BOOL	TRUE	Target position has been reached.
wActiveProfile	WORD	Profile number of the currently active positioning command.	

5.3

Parameter

L_TF2P_scPar_TablePositioningBase

Identifier	Data type	Description
IrVel	LREAL	Speed for movements that are commissioned via the scCtrlBasicMotion structure. - Unit: unit/s - Initial value: 3600
IrAcc	LREAL	Acceleration for movements that are commissioned via the scCtrlBasicMotion structure. - Selection of the value for the maximum acceleration speed variation. - Unit: unit/s² - Initial value: 36000
IrDec	LREAL	Deceleration for movements that are commissioned via the scCtrlBasicMotion structure. - Selection of the value for the maximum deceleration speed variation until standstill. - Unit: unit/s² - Initial value: 100
IrJerk	LREAL	Jerk for movements that are commissioned via the scCtrlBasicMotion structure. - Einheit: 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.
xAbsoluteModuloCycle	BOOL	TRUE Execute Modulo cycles completely. Initial value: FALSE
ascProfiles	ARRAY[1-15] OF L_TF2P_scPositionProfile	Field with 15 travel profile datasets. Each data structure contains all of the data to generate a positioning profile.

7 Signal flow diagram



Tip!

The signal flow plan is not available in this technology module.

7.1 Structure of the signal flow

L_TF2P_scSF_TablePositioningBase

The content of the structure is read-only and offers a practical diagnostics option within the signal flow.

Identifier	Data type	Description	
MP_xTpReceived	BOOL	TRUE	A touch probe mark has been detected.
MP_xTpMissed	BOOL	TRUE	No touch probe mark was detected within the touch probe window.
MP_xTpOutOfWindow	BOOL	TRUE	A touch probe mark has been detected outside of the touch probe window.

7.2 Structure of the access points



Tip!

The structure of the access points is not available on this technology module.

8 Description of the basic functions

Overview of the basic functions

- [Manual jog \(jogging\)](#) (📖 22)
- [Homing](#) (📖 24)
- [QSP application](#) (📖 27)
- [Halt](#) (📖 28)
- [Absolute positioning](#) (📖 29)
- [Relative positioning](#) (📖 31)
- [Continuous travel with a defined velocity](#) (📖 34)
- [Aborting and completing a positioning process](#) (📖 37)
- [Override for speed](#) (📖 39)

8 Description of the basic functions

8.1 Manual jog (jogging)

8.1 Manual jog (jogging)

Prerequisite

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

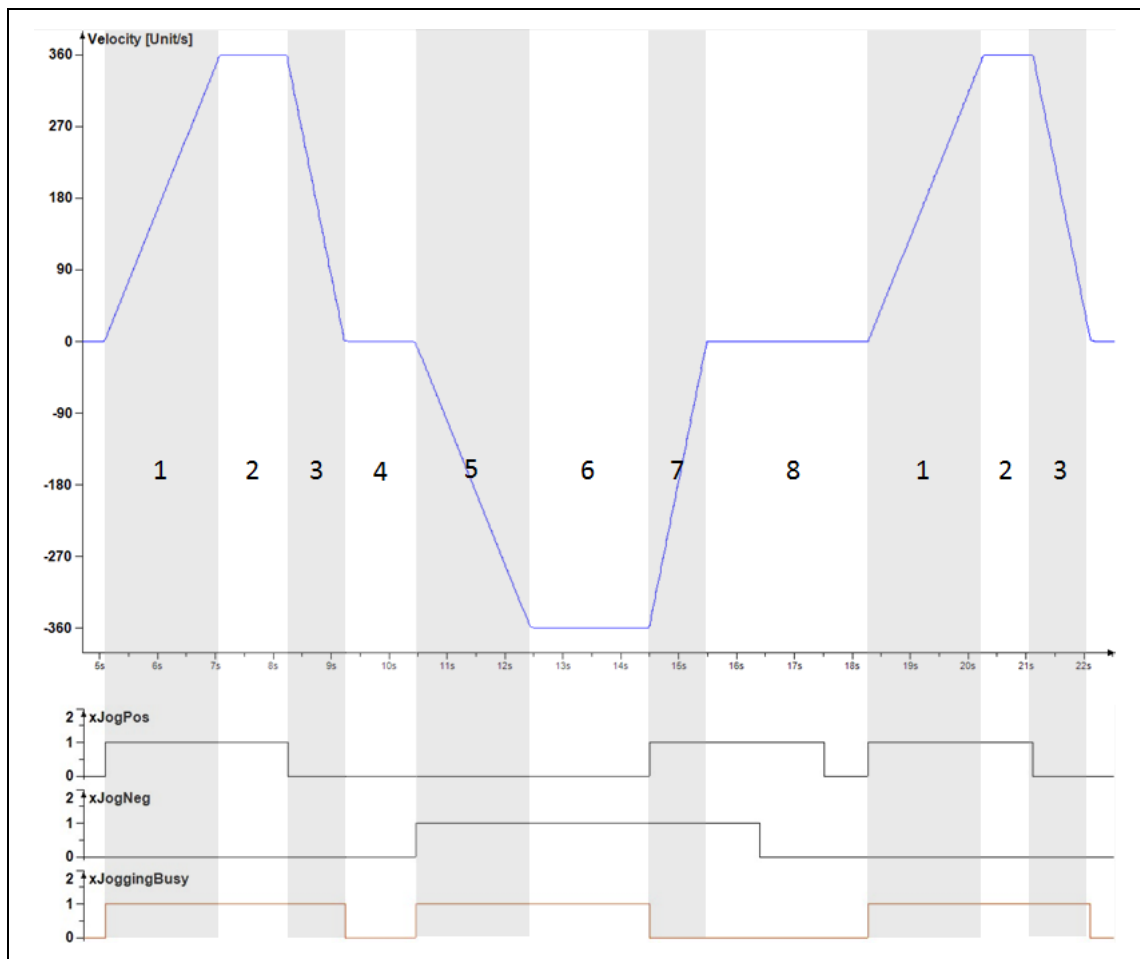
Execution

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.

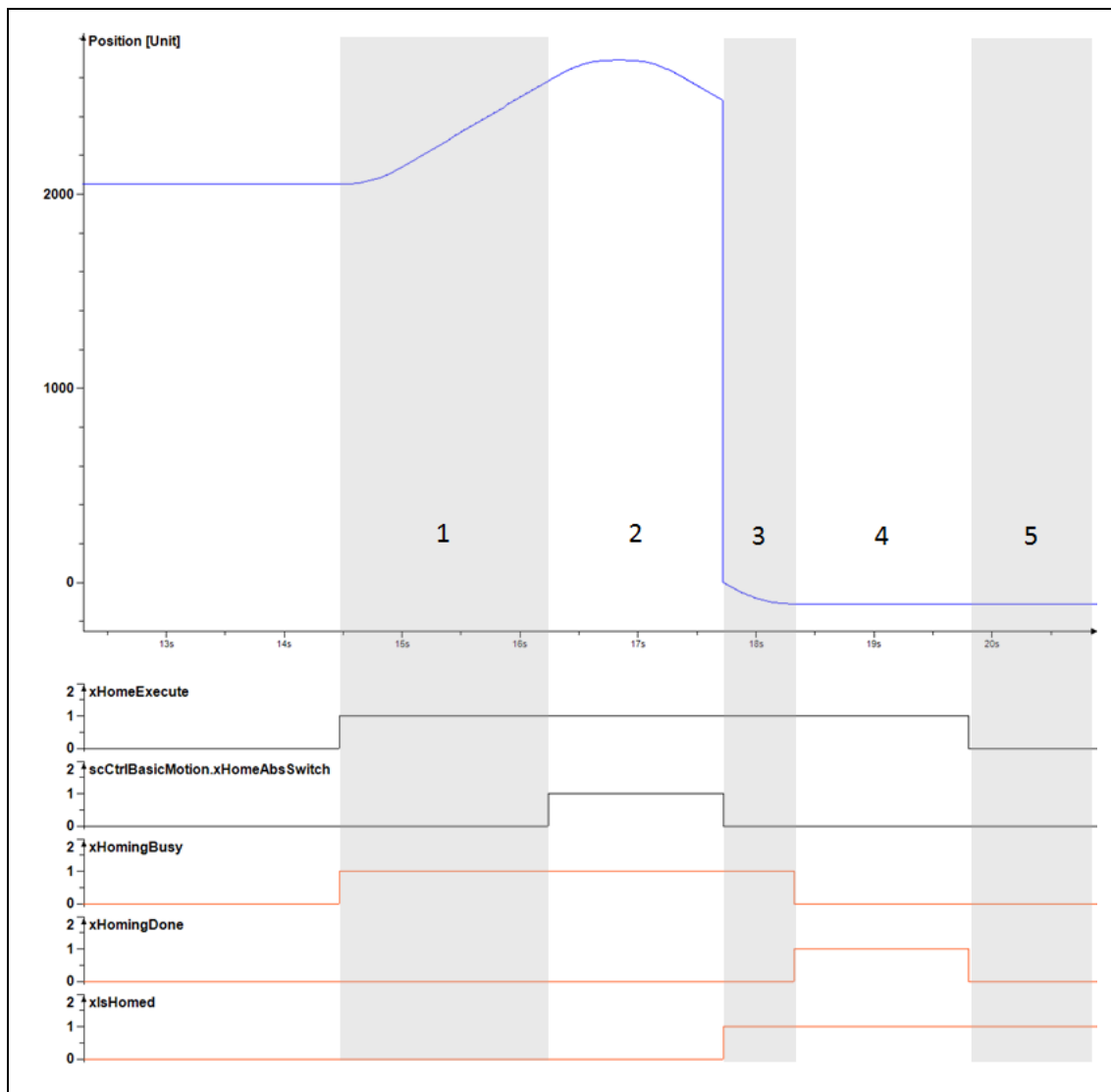
If *xHomeExecute* = FALSE, the state is changed to "STANDBY" and *xHomingDone* is set = FALSE.

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 <i>xHomeAbsSwitch</i> .
3	TRUE	FALSE → TRUE	TRUE	FALSE	FALSE → TRUE	HOMING	The negative edge of <i>xHomeAbsSwitch</i> 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, <i>xHomingBusy</i> is set = FALSE and <i>xHomingDone</i> is set = TRUE. <i>xHomingDone</i> remains TRUE for as long as <i>xHomeExecute</i> 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

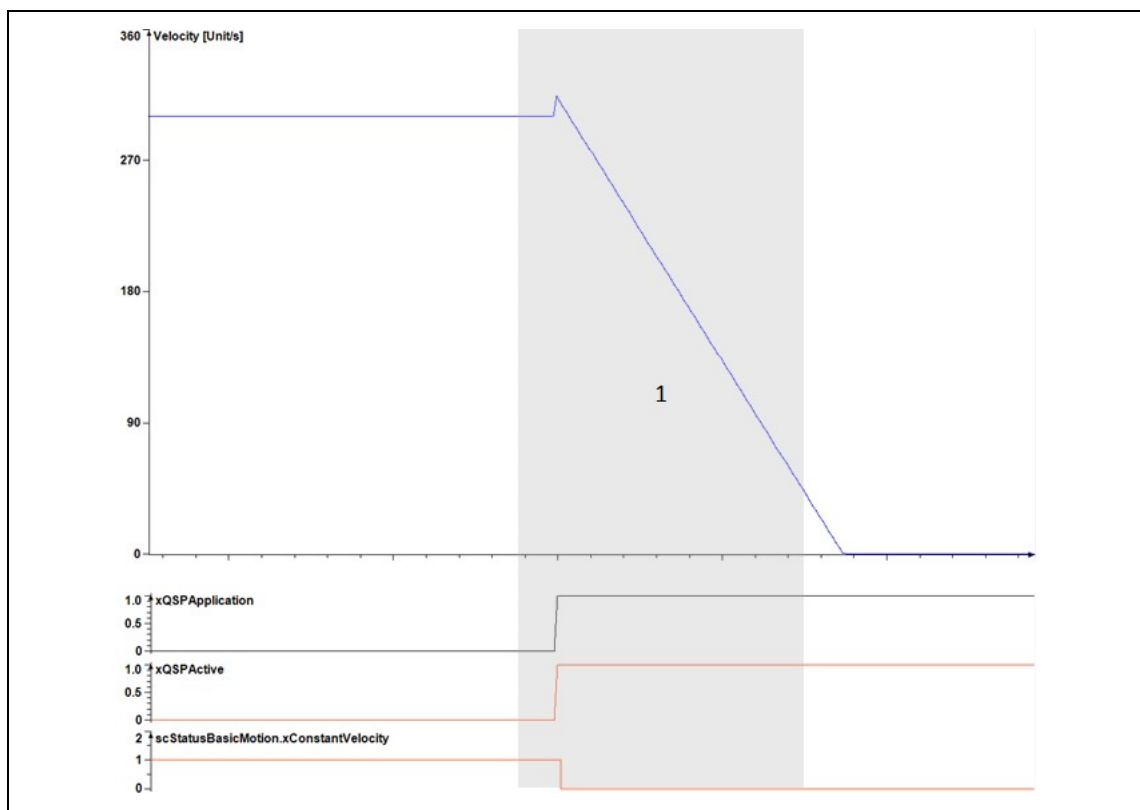
Prerequisite

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

Execution

With a rising edge (FALSE→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→TRUE	FALSE→TRUE	TRUE→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

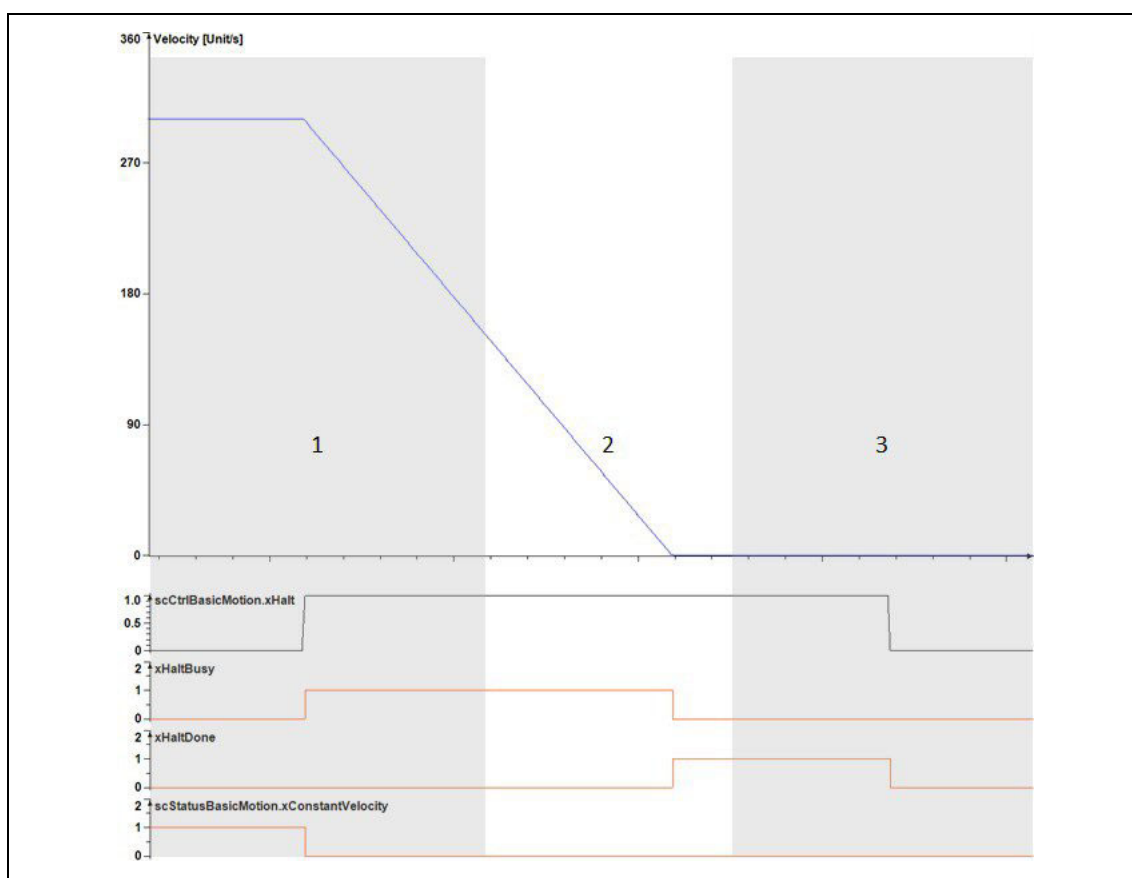
Prerequisite

- 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



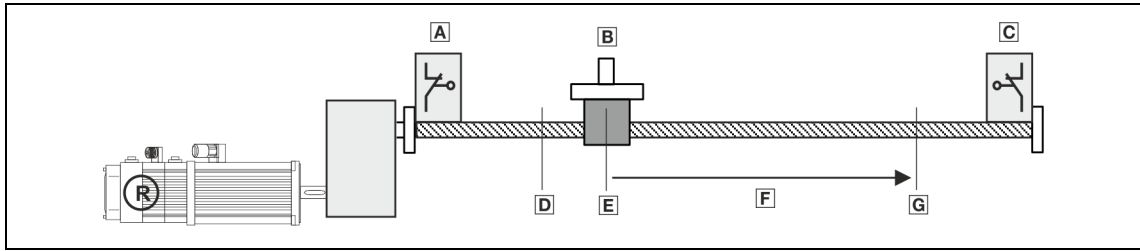
No	xHalt	xHaltBusy	xHaltDone	xConstant Velocity	eTMState	Info
1	$FALSE \rightarrow TRUE$	$FALSE \rightarrow TRUE$	$FALSE$	$TRUE \rightarrow FALSE$	$MOVEVEL \rightarrow STOP$	The axis decelerates with the halt deceleration ramp: $xHaltBusy = TRUE$ $xConstantVelocity = FALSE$.
2	$TRUE$	$TRUE \rightarrow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

8.5

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.

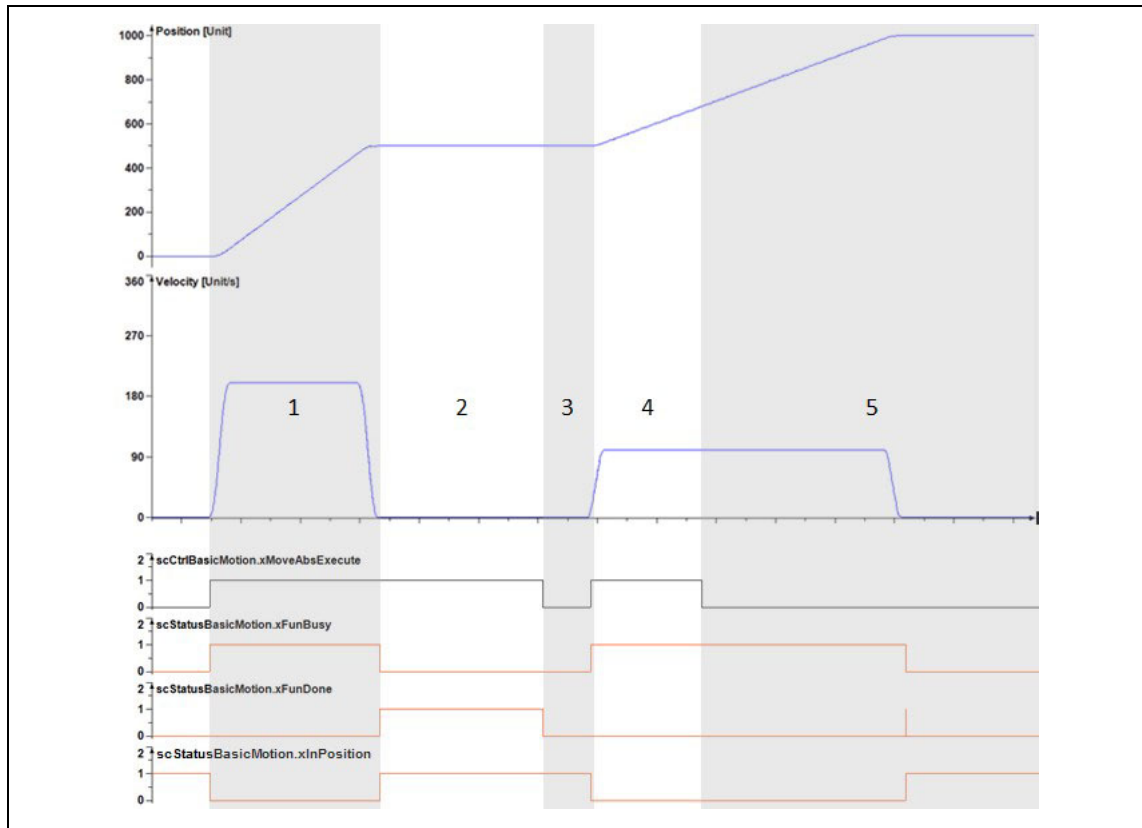
In order to reach the target position, the axis is brought to a standstill via the *lrDec* parameter. The axis reaches a standstill exactly in the target position.

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.

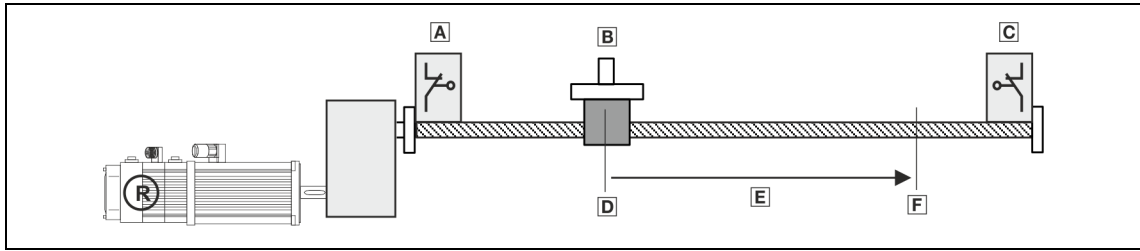
Parameters to be set

- *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.

8.6

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.

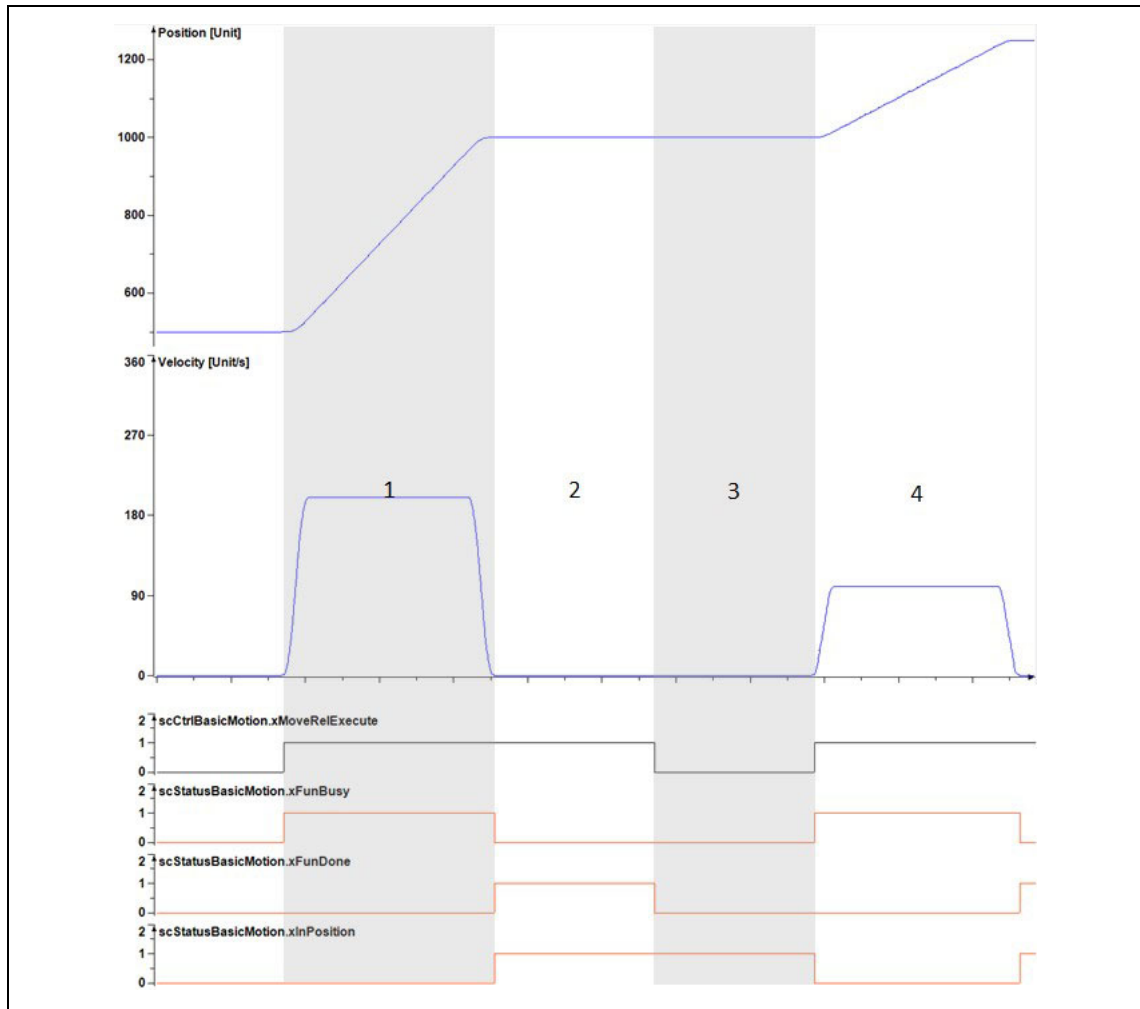
In order to reach the target position, the axis is brought to a standstill via the *lrDec* parameter. The axis reaches a standstill exactly in the target position.

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:

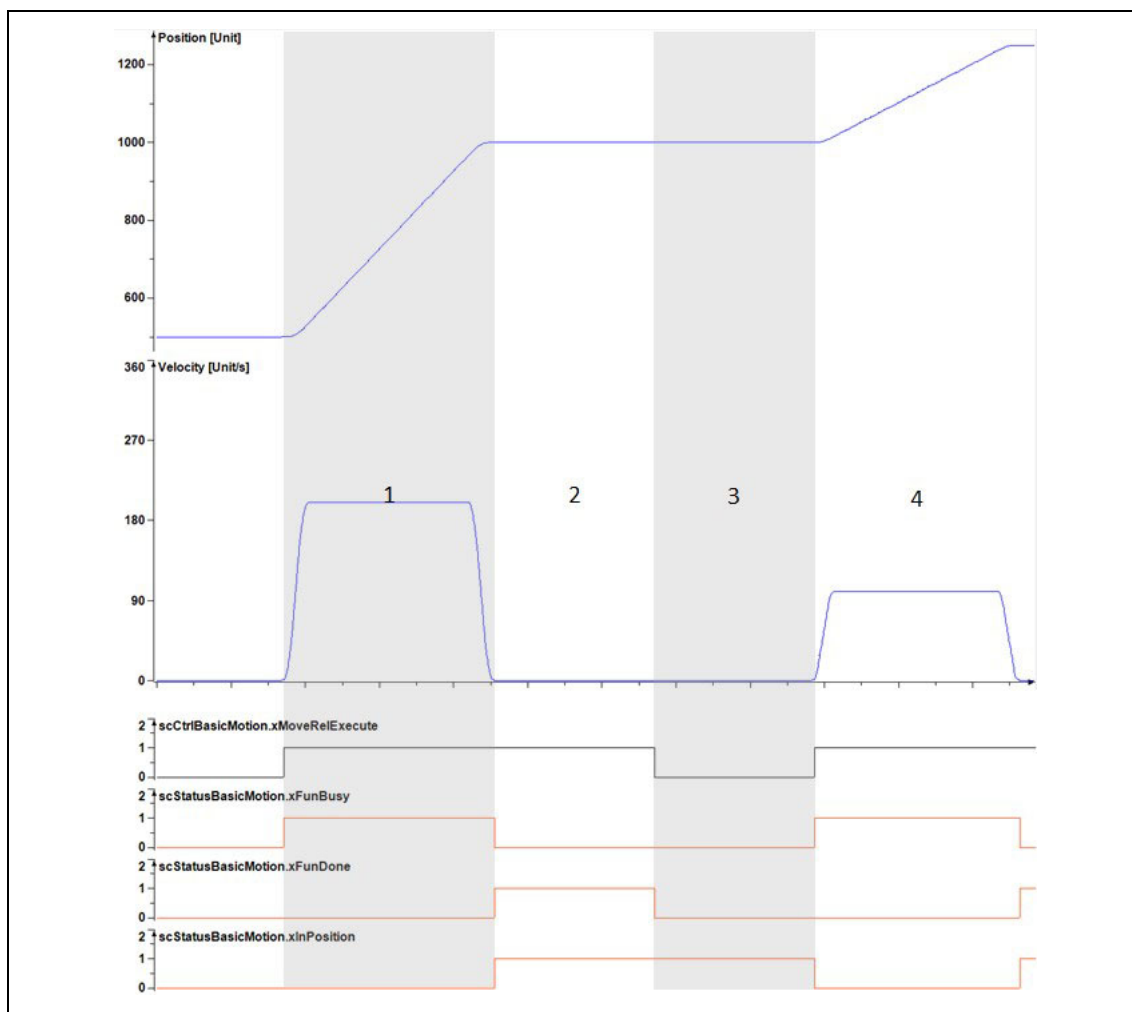
- Input *xDisableSWLimit* = FALSE
- Output *xSwLimitEnabled* = TRUE
- Output *xSwLimitSwitchPos/Neg* = TRUE
- Output *xHwLimitSwitchPos/Neg* = TRUE

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	xMoveVel Execute	xAbort	xFuncBusy	xFuncDone	xConstant Velocity	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 ($xReadyForMotion = TRUE$).

Execution

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

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

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

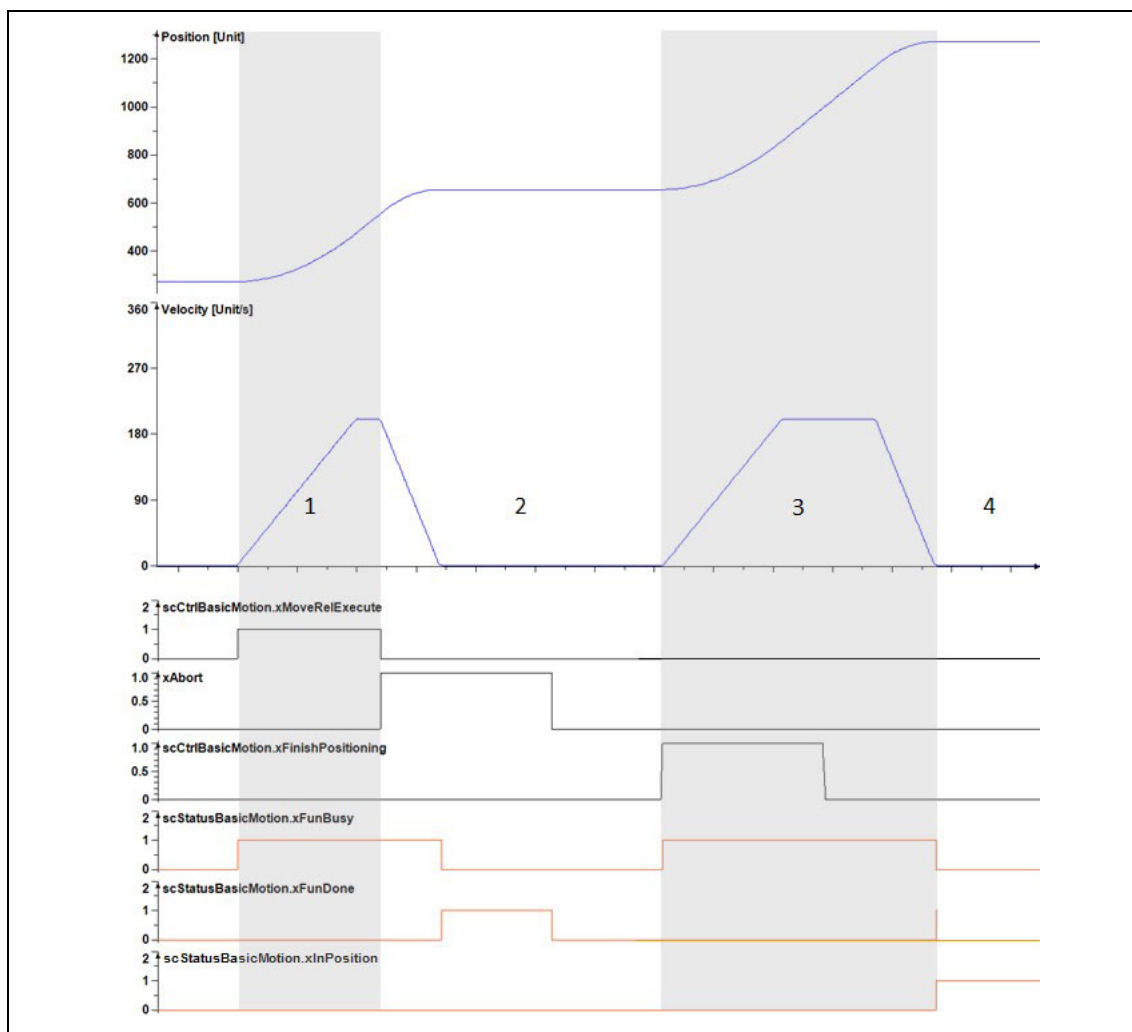
Both technology modules provide several profile data sets, one of which can be active.

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 ($FALSE \rightarrow TRUE$) 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* = TRUE).

Execution

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

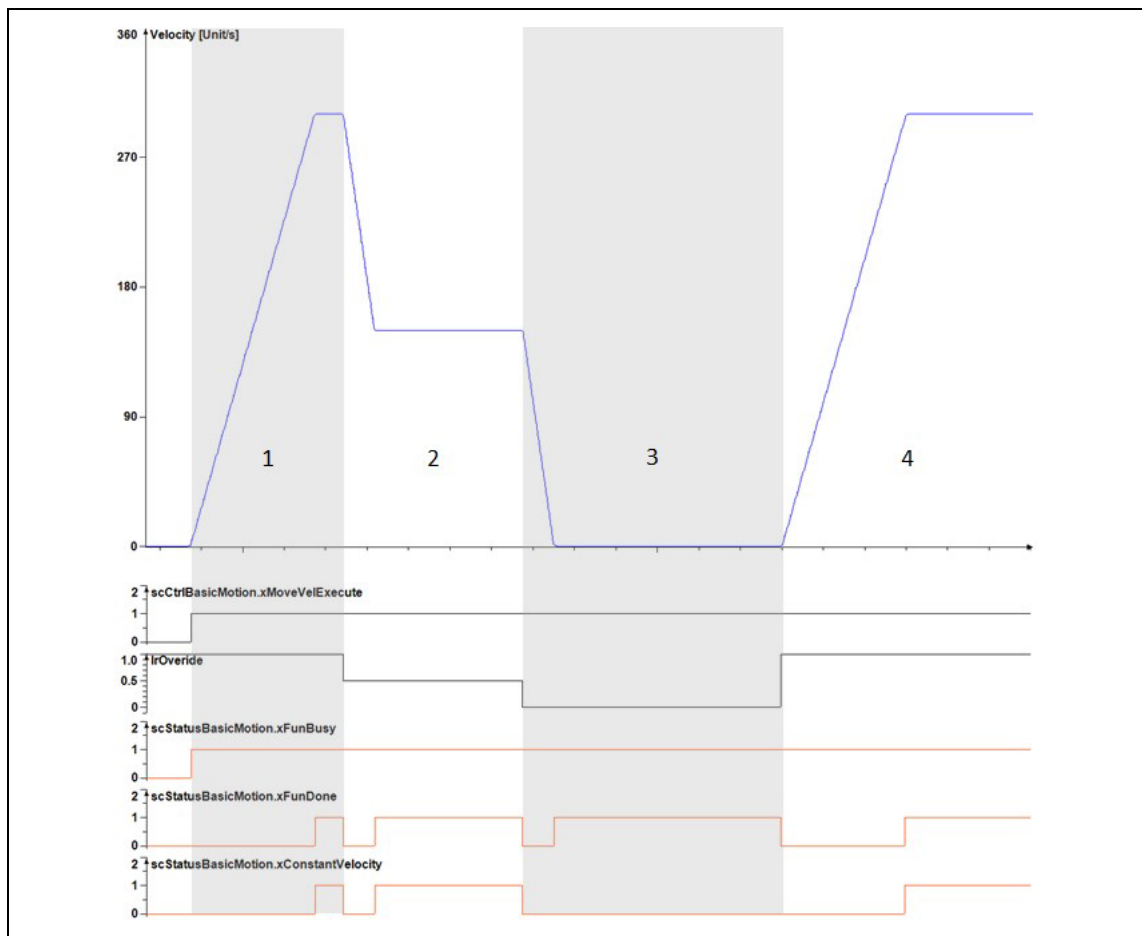
- *IrOverride*, initial value: 1.0
- *scCtrlBasicMotion.IrOverrideAcc*, initial value: 1.0 (100 %)
- *scCtrlBasicMotion.IrOverrideJerk*, initial value: 1.0 (100 %)

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 function

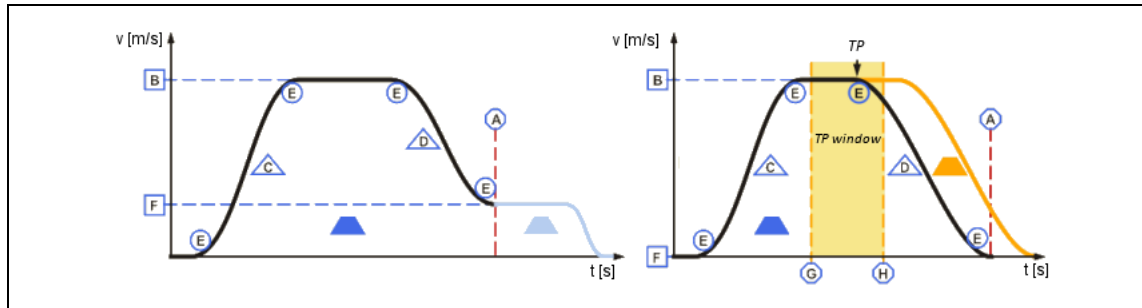
9.1 Profile data management

9 Description of the technology function

9.1 Profile data management

Profile data management manages the parameters of a maximum of 15 travel profiles of type *L_TF2P_scPositionProfile*.

A travel profile describes a motion task for positioning that is implemented by a rotary motion of the motor shaft.

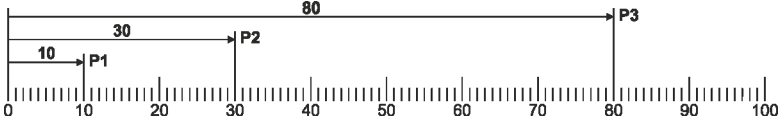
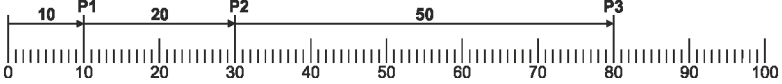
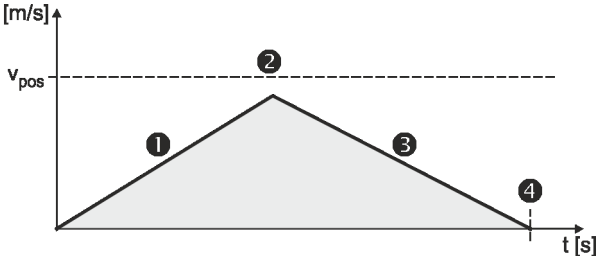


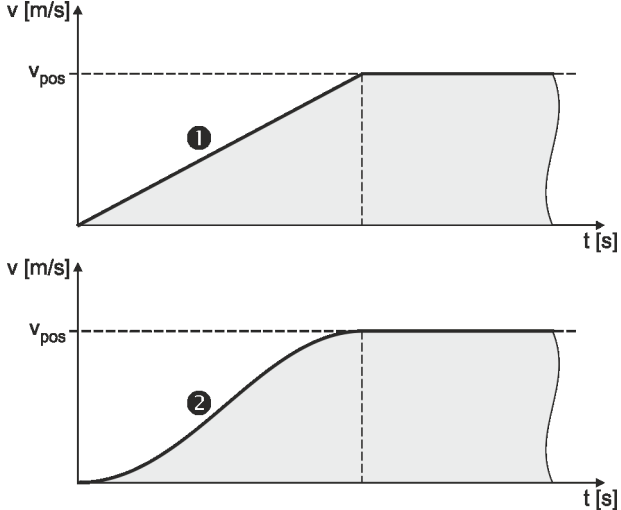
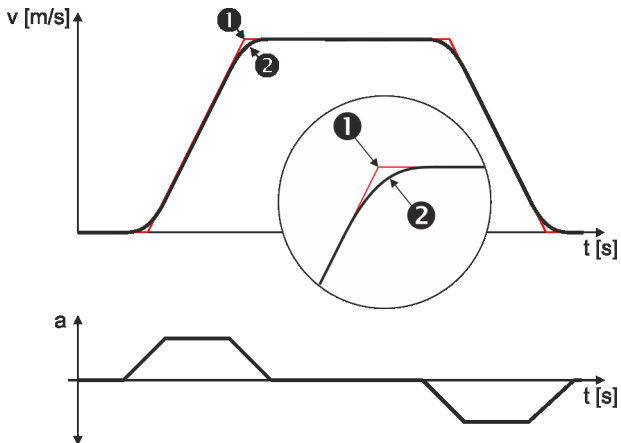
[9-1] Profile types

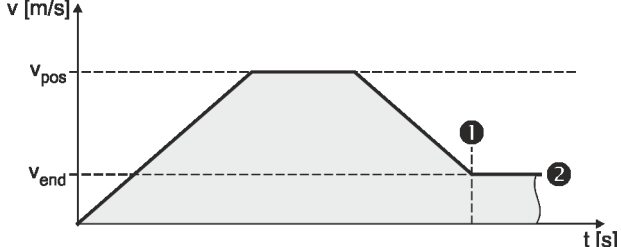
Symbol	Description
	wProfileNumber Number of the positioning command (1 - 15) performed first

The travel profile is described by the following parameters:

Symbol	Profile parameters
	ePositioningMode Selection of the positioning mode
	wTpProfile Number of the profile data set (1 - 15) that is to be executed after a detected touch probe. When "0" is set, no travel profile is executed after a touch probe has been detected.
	wSequenceProfile Number of the profile data set (1 - 15) that is to be executed after the current travel profile. When "0" is set, no further travel profile will be executed. The travel profile will only be executed if no touch probe has been detected within the touch probe window.

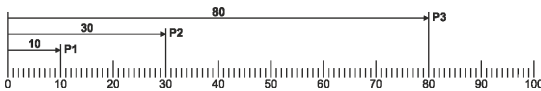
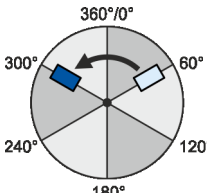
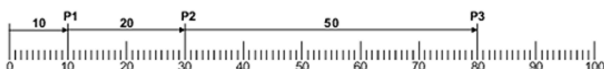
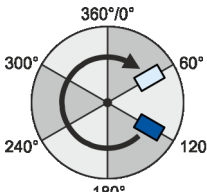
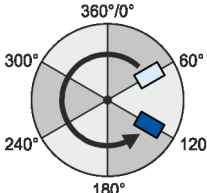
Symbol	Profile parameters
A	IrPosition Target position or distance to be traversed. IrPosition is either given "absolutely" or "relatively". The <u>absolute position</u> describes the distance from the zero position to the target position. (absolute position = target position) The home position for the absolute position is the zero position.  The <u>relative position</u> takes the current actual position at the starting time of the travel command into consideration. (relative position = target position - actual position) The home position for the starting point of the distance is the target position of the travel profile executed previously. 
B	IrVelocity Maximum velocity used for approaching the target position. Depending on the target position (IrPosition), acceleration (IrAcceleration) and deceleration (IrDeceleration). It is possible that the drive does not reach the maximum velocity. In this case, the graph is shown as a triangle instead of a trapezium.  1 Acceleration 2 Maximum speed (not reached here) 3 Deceleration 4 Target position

Symbol	Profile parameters
	IrAcceleration Acceleration to reach the maximum velocity (Velocity Profiles). v [m/s]  v_{pos} t [s] The following acceleration types are distinguished: ❶ Linear increase in speed ❷ S-shaped increasing speed
	IrDeceleration Deceleration to brake from the maximum velocity (Velocity Profiles) to a standstill.
	IrJerk Jerk limitation for acceleration (acceleration profiles) and deceleration (deceleration profiles). If a jerk limitation is specified for a travel profile, acceleration and deceleration are executed more smoothly via S-shaped ramps. This prevents, for instance, sensitive machine parts from damages. The S-shaped acceleration and deceleration extend the duration of positioning compared to linear acceleration and deceleration. v [m/s]  t [s] a t [s] ❶ Profile without jerk limitation ❷ Profile with jerk limitation

Symbol	Profile parameters
F	IrFinalVelocity Final speed the drive uses to execute the next travel profile after reaching the target position. When the target position has been reached, the next travel profile is executed immediately without braking the drive to a standstill at the target position. The final speed can be used to execute a velocity changeover.  ① Target position ② Final velocity
G	IrTpWindowNeg Lower limit value for the touch probe window in which a touch probe can be detected.
H	IrTpWindowPos Upper limit value for the touch probe window in which a touch probe can be detected.

9.2

Positioning modes

No.	Positioning mode	Description
0	absolute	The axis travels to an absolute position. - The absolute position describes the distance between the zero position and the target position. (absolute position = target position) - The home position for the absolute position is the zero position. - The home position must be known.  The rotary table positioning is basically an absolute positioning with target positions between 0 and 360 angular degree [°]. In this mode the zero point can also be overtravelled if this is the shortest way to the target position: 
1	relative	The axis travels a defined distance. - The relative position considers the distance to the starting position. (relative position = target position - starting position) - The home position for the starting point of the distance is the target position of the travel profile executed previously. 
2	velocity	Continuous constant travel/movement - This mode does not require a predefined position but follows the profile parameters. - Acceleration and deceleration are based on the profile parameters <i>IrAcceleration</i> and <i>IrDeceleration</i>. - The travel direction is defined by the sign of the travelling speed (+: CW / -: CCW).
3	absolute CW_modulo	The axis travels to an absolute position in CW direction. - Can only be used for Modulo measuring system (rotary table application). - The home position for the absolute position is the zero position. - The home position must be known. In CW direction, the zero position of the axis can be overtravelled. 
4	absolute CCW_modulo	The axis travels to an absolute position in CCW direction. - Can only be used for Modulo measuring system (rotary table application). - The home position for the absolute position is the zero position. - The home position must be known. In CCW direction, the zero position of the axis can be overtravelled. 

Notice

If a valid TP sequence profile has been set in the profile data set (TP Overchange Profile) the touch probe interface is also evaluated.

9.3

Perform positioning

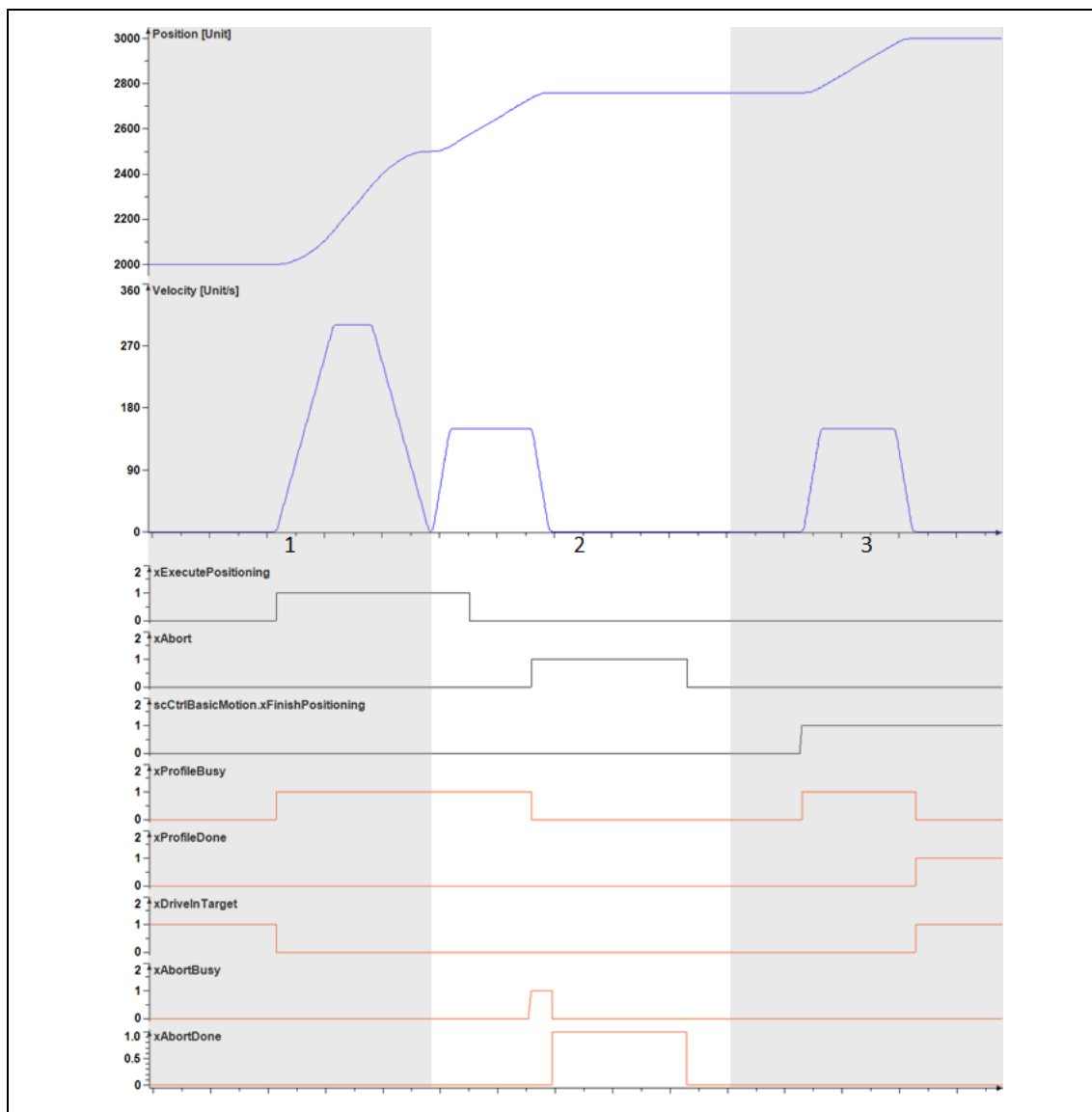
Condition

- The axis is enabled (*xReadyForMotion* = TRUE).
- A valid travel profile is defined in the profile data ▶ [Profile data management](#) (41)
- A valid travel profile (1 - 15) is selected in *wProfileNumber*.

Type

The following profile linkage is executed with a FALSE→TRUE edge at the *xExecutePositioning* input:

wProfileNumber	1	2
ePositioningMode	relative	relative
IrPosition	500	500
IrVelocity	300	150
IrAcceleration	300	500
IrDeceleration	300	500
IrJerk	7200	7200
IrFinalVelocity	0	0
IrTpWindowNeg	0	0
IrTpWindowPos	0	0
wTpProfile	0	0
wSequenceProfile	2	0



No	xExecutePositioning	xAabort	xFinishPositioning	xProfileBusy	xDriveInTarget	xAabortBusy	eTMState	INFO
1	FALSE → TRUE	FALSE	FALSE	FALSE → TRUE	TRUE → FALSE	FALSE	STANDBY → MOVEREL	The axis accelerates at 300 unit/s ² to the speed of travel profile 1 of 300 units/s and travels to the first target position. The profile speed is 0 units/s in the target position.
2	TRUE → FALSE	FALSE → TRUE → FALSE	FALSE	TRUE → FALSE	FALSE	FALSE → TRUE → FALSE	MOVEREL → STOP → STANDBY	The axis accelerates at 500 unit/s ² to the speed of travel profile 2 of 150 units/s. During the constant travel phase, the relative positioning profile is aborted. The axis runs at the profile deceleration of 500 unit/s ² to a standstill.
3	FALSE	FALSE	FALSE → TRUE TRUE → FALSE	FALSE → TRUE	FALSE → TRUE	FALSE	JOGGING	The profile is run to the end with the <i>scCtrlBasicMotion.xFinishPositioning</i> input. As soon as the target position is reached, <i>xDriveInTarget</i> output = TRUE appears.

9.4 Touch probe positioning

Condition

- The axis is enabled ($xReadyForMotion = TRUE$).
- A valid travel profile is defined in the profile data ▶ [Profile data management](#) (41)
- A valid travel profile (1 - 15) is selected in $wProfileNumber$.

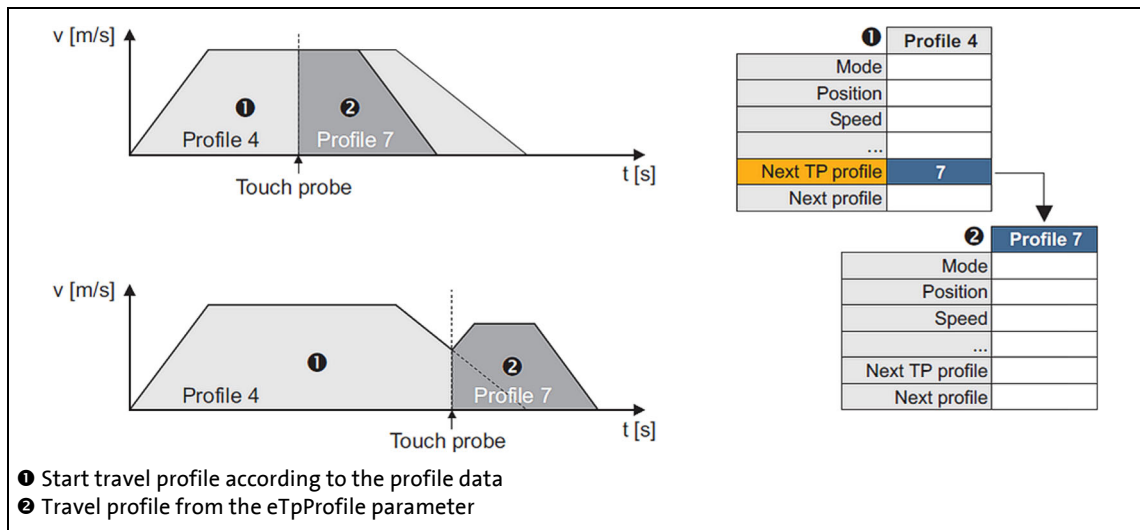
Type

The travel profile according to the profile data is initially executed.

▶ [Perform positioning](#) (47)

If a touch probe is detected during the positioning process, it is automatically changed to the profile defined in the $eTpProfile$ profile parameter.

The current position is saved at the time when the touch probe is detected by a sensor. Directly at this position the travel profile is now executed from the $eTpProfile$ profile parameter.



[9-2] Example: Diagrams for profile change after a touch probe

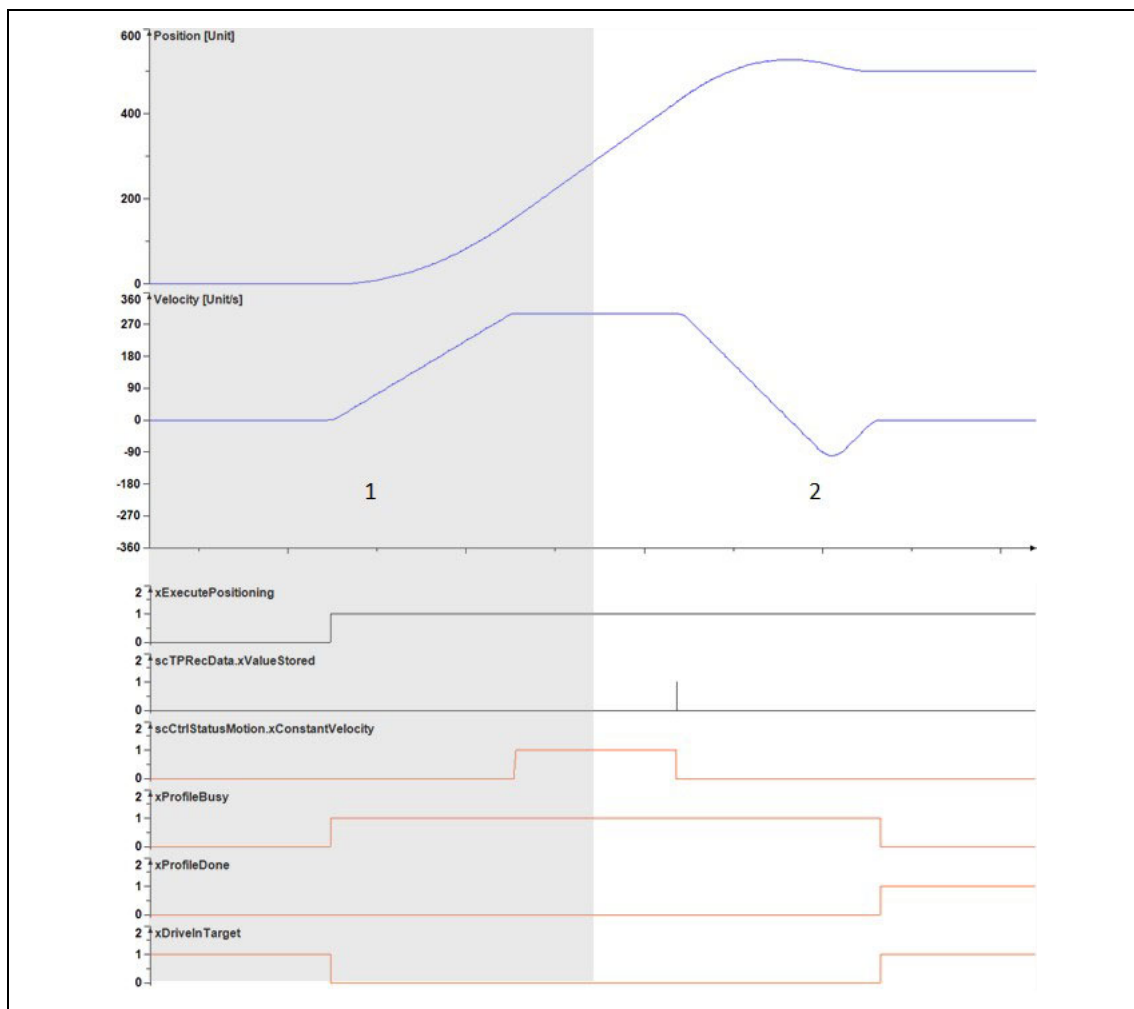
The profile parameters $IrTpWindowNeg$ and $IrTpWindowPos$ limit the area (touch probe window) in which a touch probe can be detected. If both parameters have the value '0', the touch probe detection is activated for the entire travel profile or travel range.

If no touch probe is detected and the travel profile has been executed completely, the profile set in the $eSequenceProfile$ profile parameter is executed.

The following profile linkage is executed with a FALSE→TRUE edge at the $xExecutePositioning$ input:

$wProfileNumber$	1	2
$ePositioningMode$	velocity	absolute
$IrPosition$	500	500
$IrVelocity$	300	150
$IrAcceleration$	300	500
$IrDeceleration$	300	500
$IrJerk$	7200	7200
$IrFinalVelocity$	0	0
$IrTpWindowNeg$	0	0

wProfileNumber	1	2
IrTpWindowPos	0	0
wTpProfile	2	0
wSequenceProfile	0	0



No	xExecute Positioning	xValue Stored	xConstant Velocity	xProfile Busy	xDriveIn Target	eTMState	INFO
1	FALSE → TRUE	FALSE	FALSE → TRUE	FALSE → TRUE	TRUE → FALSE	STANDBY → MOVEREL	The axis accelerates at 300 unit/s ² to the speed of travel profile 1 of 300 units/s.
2	TRUE	FALSE → TRUE → FALSE	TRUE → FALSE	TRUE → FALSE	FALSE → TRUE	MOVEREL → MOVEABS	Profile 2 is activated and the axis is positioned on the absolute target position 500 units as soon as a touch probe event triggers <i>scRecData.xValueStored</i> .

Index

A

Aborting a positioning process [37](#)

Absolute positioning [29](#)

Access points [20](#)

Application notes [7](#)

C

Competing a positioning process [37](#)

Continuous travel with a defined velocity [34](#)

Conventions used [6](#)

D

Document history [5](#)

E

E-mail to Lenze [52](#)

F

Feedback to Lenze [52](#)

H

Halt, [28](#)

Homing [24](#)

I

inputs [14](#)

L

L_TF2P_scPar_TablePositioningBase [18](#)

L_TF2P_scPar_TablePositioningBase parameter structure [18](#)

L_TF2P_TablePositioningBase [13](#)

L_TF2P_TablePositioningBase (functional description) [10](#)

Layout of the safety instructions [7](#)

M

Manual jog (jogging) [22](#)

N

Notes on how to operate the technology module [11](#)

O

outputs [16](#)

Override for speed [39](#)

P

Perform positioning [47](#)

Positioning modes [45](#)

Profile data management [41](#)

Q

QSP application [27](#)

R

Relative positioning [31](#)

S

Safety instructions [8](#)

Signal flow diagram [20](#)

Speed override [39](#)

Start of the axes [11](#)

State machine [19](#)

States [19](#)

Structure of the signal flow L_TF2P_scSF_TablePositioningBase [20](#)

T

TablePositioningBase (V2) - functional description [10](#)

Target group [4](#)

Technology module functions (overview) [10](#)

Touch probe positioning [49](#)

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

