

Technology module



Cross Cutter

Reference Manual

EN



Contents

1	About this documentation	3
1.1	Document history	5
1.2	Conventions used	6
1.3	Definition of the notes used	7
2	Safety instructions	8
3	Functional description of "Cross Cutter"	10
3.1	Overview of the functions	12
3.2	Important notes on how to operate the technology module	13
3.3	Function block L_TT1P_CrossCutter[Base/State]	15
3.3.1	Inputs and outputs	16
3.3.2	Inputs	17
3.3.3	Outputs	20
3.3.4	Parameters	23
3.4	State machine	28
3.5	Signal flow diagrams	31
3.5.1	Structure of the signal flow	33
3.5.2	Structure of the access points	37
3.6	Parameterisation of the sealing roller	38
3.7	Positioning to the parking position	39
3.8	Manual jog (jogging)	41
3.9	Homing	41
3.10	"Length cutting" operating mode	42
3.11	Top cut	44
3.12	"Flow packer" operating mode	45
3.13	Positioning the cross cutter to static material	49
3.14	Clutching the cross cutter to moving material	51
3.15	Declutching the cross cutter	53
3.16	Positive opening operation / Emergency opening operation	55
3.17	Trimming and offset of the cutting position	57
3.17.1	Position cut trimming	58
3.17.2	Speed cut trimming (Tip trimming)	60
3.17.3	Absolute positioning of the cutting position via master offset	62
3.18	Method of elimination	64
3.19	Increasing/decreasing the speed (overspeed)	66
3.20	Calculating extreme values of a cam	68
3.21	Calculating the cutting length	68
3.22	"Length cutting with mark correction" operating mode	69
3.23	Partial cutting lengths in product length (print/cut format)	71
3.24	Cutting mark register	72
3.25	Selecting the touch probe source	74
3.26	Referencing the mark position	75
3.27	Speed synchronism of master and register axis	76
3.28	Gearbox factor correction	77
3.29	"Cutting on a mark (event-controlled)" operating mode	80
3.30	Torque feedforward control	83
3.31	CPU utilisation (example Controller 3231 C)	84
	Index	85
	Your opinion is important to us	87

1 About this documentation

This documentation ...

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

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

More technical documentation for Lenze components

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

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

Symbols:

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



Tip!

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

www.lenze.com

Target group

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

1

About this documentation

1.1

Document history

1.1

Document history

Version			Description
5.0	12/2019	TD06	Error corrections & supplements
4.0	06/2019	TD29	• General revisions • New: parameter <code>lrSetOffsetMarkWindow</code>
3.3	07/2018	TD29	• General revisions
3.2	05/2017	TD17	• Content structure has been changed. • General revisions
3.1	04/2016	TD17	• General revisions • New: Structure of the access points L TT1P_scAP_CrossCutter[Base/State] (37)
3.0	11/2015	TD17	• Corrections and additions • Content structure has been changed.
2.1	05/2015	TD17	General revisions
2.0	01/2015	TD17	• General editorial revision • Modularisation of the contents for the »PLC Designer« online help
1.0	04/2014	TD00	First edition

1 About this documentation

1.2 Conventions used

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

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Text		
Program name	» «	»PLC Designer« ...
Variable names	<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; ...
Icons		
Page reference	 6	Reference to further information: Page number in PDF file.

Variable names

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

1.3

Definition of the notes used

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

Safety instructions

Layout of the safety instructions:

**Pictograph and signal word!**

(characterise the type and severity of danger)

Note

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

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

Application notes

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

2 Safety instructions

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



The device documentation contains safety instructions which must be observed!

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



Danger!

High electrical voltage

Injury to persons caused by dangerous electrical voltage

Possible consequences

Death or severe injuries

Protective measures

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

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

Observe the corresponding information plates on the device.



Danger!

Injury to persons

Risk of injury is caused by ...

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

Possible consequences

Death or severe injuries

Protective measures

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



Stop!

Damage or destruction of machine parts

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

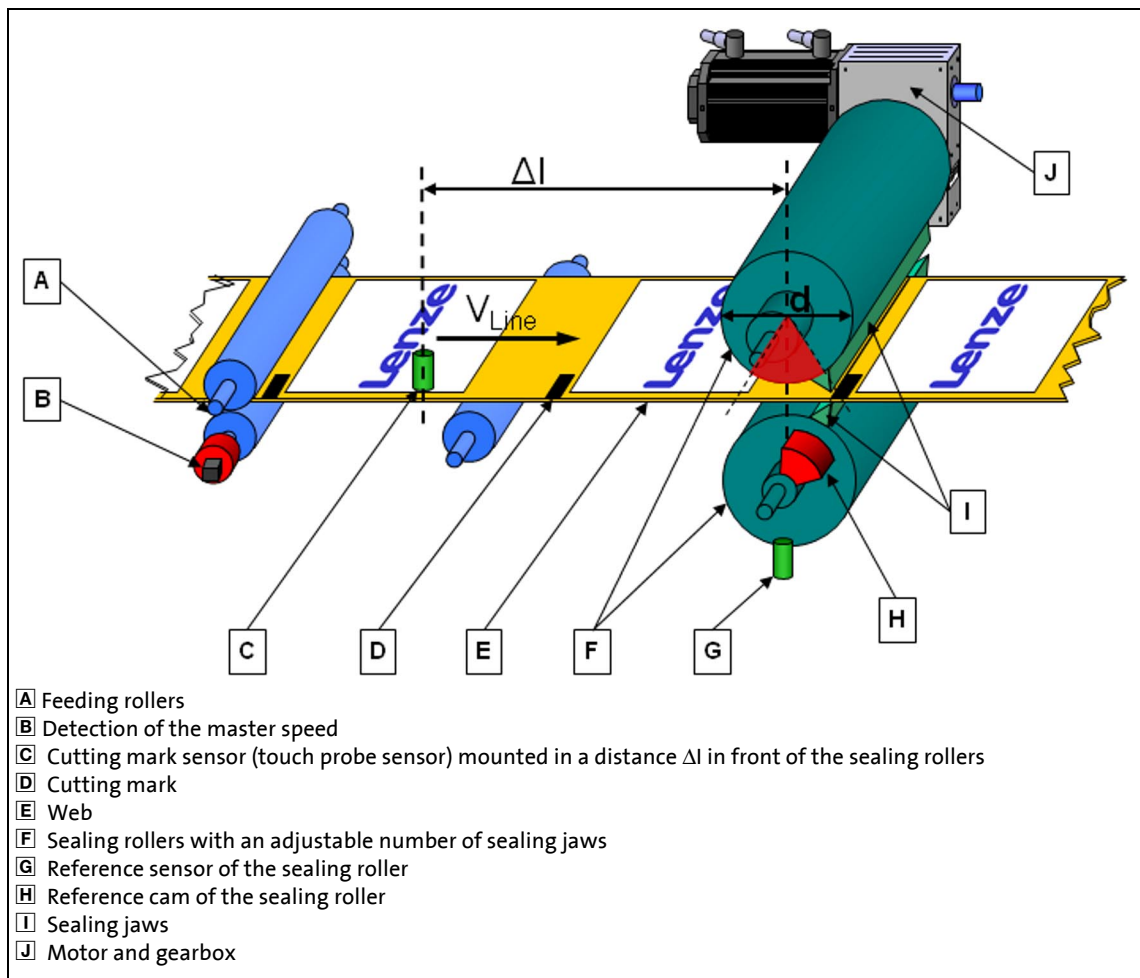
- Short circuit or static discharges (ESD);
- unpredictable motor movements (e.g. unintended direction of rotation, too high velocities or jerky movement);
- impermissible operating states during the parameterisation while there is an active online connection to the device.

Protective measures

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

3 Functional description of "Cross Cutter"

3 Functional description of "Cross Cutter"



[3-1] Typical mechanics of the technology module

The "Cross Cutter" technology module calculates the cam for the synchronous motion of a cross sealing roller and synchronises it with a master axis.

- In the "Base" version, the technology module works in a purely length-controlled way. The profiles are calculated in real time allowing to flexibly respond to product gaps or process data changes (cutting lengths, cutting asynchronicities and sealing area). The motion profile is calculated via a polynomial of the fifth degree.
- The "State" version provides an extended function range of the "Base" version:
The State version provides process modes in addition to length-cutting by allowing for cuts via the detection of marks. In this way constant lengths are cut and corrected by detected marks. In another mode, the cut is effected on a detected mark (event-controlled cut).

► [Overview of the functions](#) (12)

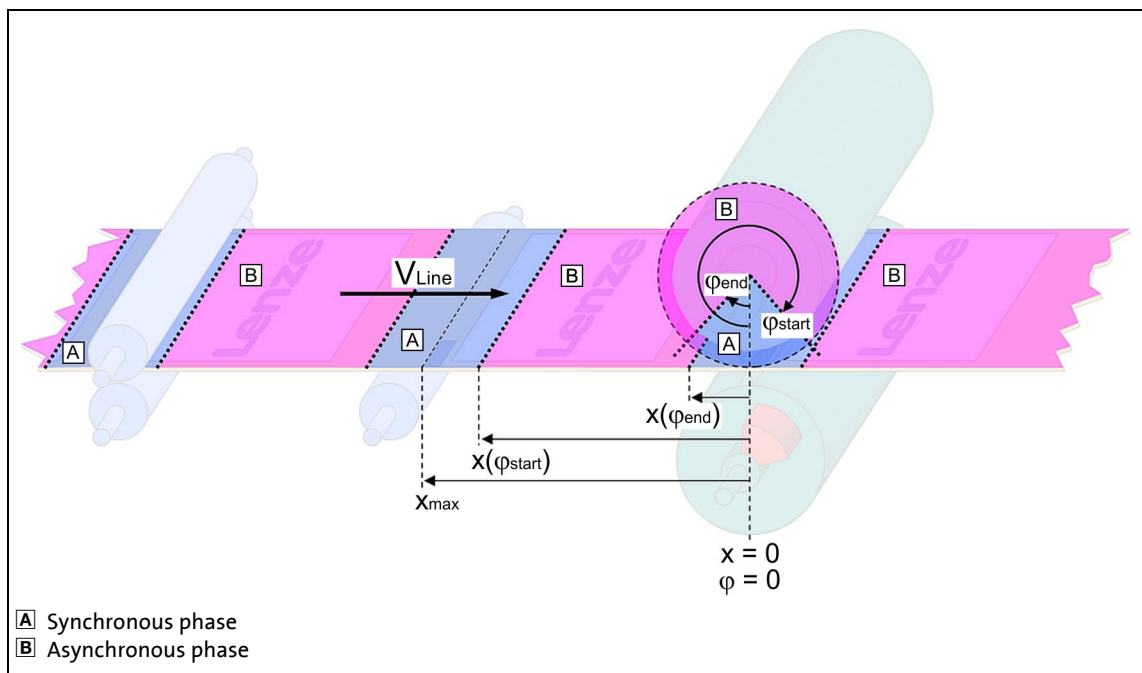
3 Functional description of "Cross Cutter"

Different parameters and process data are required for calculating the seal motion:

- Number of tools (mechanic parameter)
- Diameter of the sealing roller (mechanical parameter)
- Sealing area (synchronous area)
- Product length (process factor)
- Cutting asynchronicity (process factor)

The motion profile of the technology module is determined by these factors and is thus divided into two phases:

- Synchronous phase (cutting phase)
- Asynchronous phase



[3-2] Phases of the motion profile

Synchronous phase (cutting phase)

In this phase of motion, the material processing or the cut is effected. The sealing jaw of the sealing roller follows the motion of the material in order to ensure a high cutting accuracy. In this phase, the speed of the sealing roller mainly depends on the cutting circle diameter of the sealing roller and the material web speed. An asynchronous motion of the sealing jaws to the material would result in damage to the material web.

Asynchronous phase

In this section, the sealing roller is asynchronous to the material flow. Depending on the required seal length, the sealing roller must be undersynchronous or oversynchronous to the material speed in order to let pass the respective material length below the sealing roller or to make the sealing jaw position catch up with the material, respectively. Depending on the speed difference between material and sealing roller and the duration of the asynchronous section, the required seal length is achieved.

3.1 Overview of the functions

In addition to the basic functions for operating the **L_MC1P_AxisBasicControl** function block, the **stop function** and the **holding function**, the technology module offers the following functionalities which are assigned to the "Base" and "State" versions:

Functionality	Versions	
	Base	State
Parameterisation of the sealing roller (¶ 38)	●	●
Positioning to the parking position (¶ 39)	●	●
Manual jog (jogging) (¶ 41)	●	●
Homing (¶ 41)	●	●
"Length cutting" operating mode (¶ 42) • Top cut (¶ 44)	●	●
"Flow packer" operating mode (¶ 45)	●	●
Positioning the cross cutter to static material (¶ 49)	●	●
Clutching the cross cutter to moving material (¶ 51)	●	●
Declutching the cross cutter (¶ 53)	●	●
Positive opening operation / Emergency opening operation (¶ 55)	●	●
Trimming and offset of the cutting position (¶ 57) • Position cut trimming (¶ 58) • Speed cut trimming (Tip trimming) (¶ 60) • Absolute positioning of the cutting position via master offset (¶ 62)	●	●
Method of elimination (¶ 64)	●	●
Increasing/decreasing the speed (overspeed) (¶ 66)	●	●
Calculating extreme values of a cam (¶ 68)	●	●
Calculating the cutting length (¶ 68)	●	●
"Length cutting with mark correction" operating mode (¶ 69)		●
Partial cutting lengths in product length (print/cut format) (¶ 71)		●
Cutting mark register (¶ 72)		●
Selecting the touch probe source (¶ 74)		●
Referencing the mark position (¶ 75)		●
Speed synchronism of master and register axis (¶ 76)		●
Gearbox factor correction (¶ 77)		●
"Cutting on a mark (event-controlled)" operating mode (¶ 80)		●
Torque feedforward control (¶ 83)		●



»PLC Designer« Online help

Here you will find detailed information on the **L_MC1P_AxisBasicControl** function block, the **stop function** and the **holding function**.

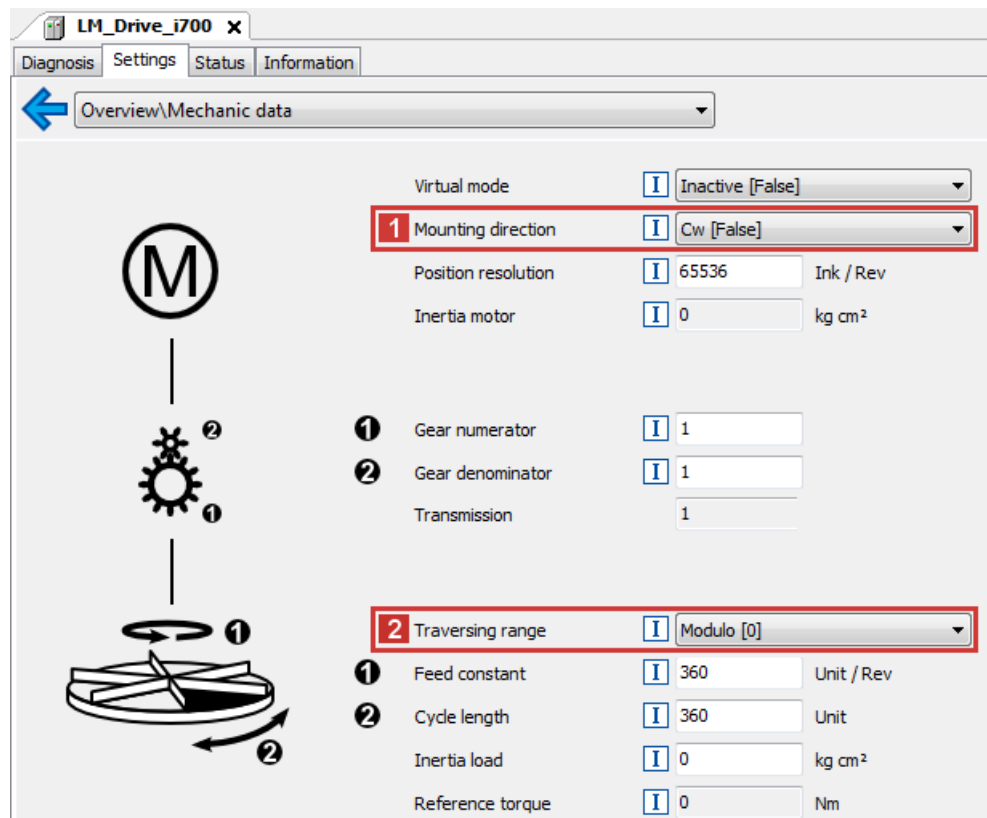
3.2

Important notes on how to operate the technology module

The "Cross Cutter" technology module only supports rotary axes:

- The master axis has to be a rotary axis and
- the slave axis has to be a rotary axis.

Go to the »PLC Designer« and set the "Modulo" machine measuring system for each axis under the **Settings** tab **2** :



During the cutting process the master axis must only move in positive direction.

In the application, the mounting direction **1** must also be set so that the master axis only moves in positive direction.

Setting of the operating mode

The operating mode for the cross cutter axis has to be set to "cyclically synchronous position" (csp) because the axis is led via the master position value.

Controlled start of the axes

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

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



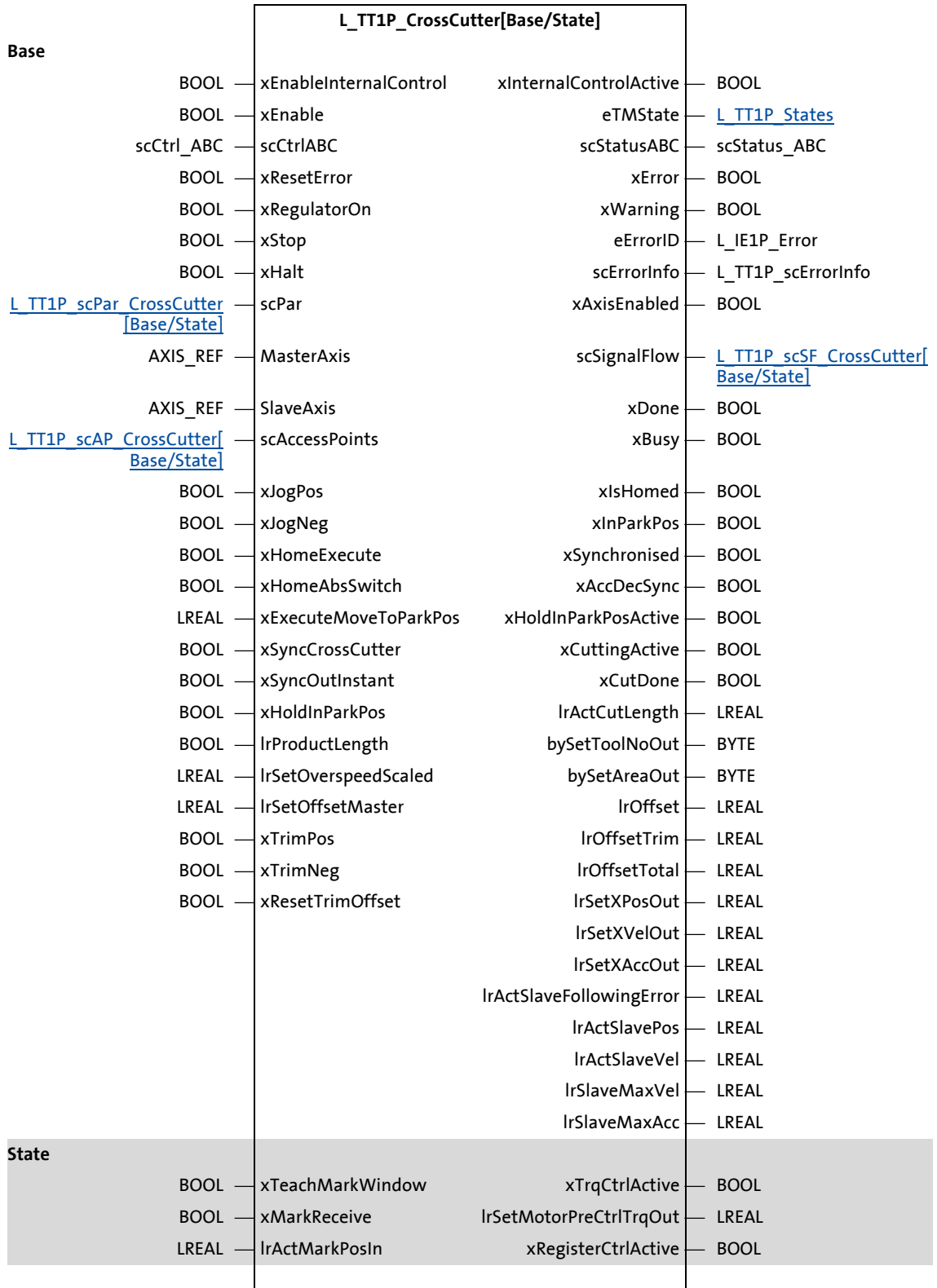
Example [Manual jog \(jogging\)](#) (41):

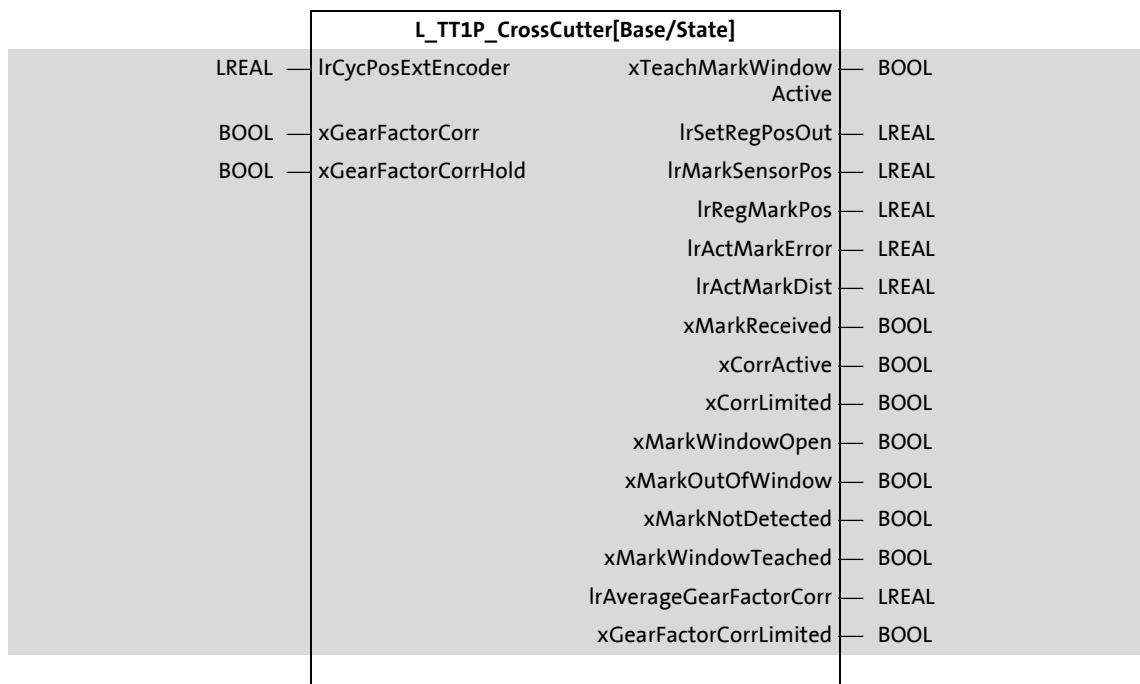
1. In the inhibited axis state ($xAxisEnabled = FALSE$), $xJogPos$ is set to TRUE.
 - $xRegulatorOn = FALSE$ (axis is inhibited.)
==> "READY" state ($xAxisEnabled = FALSE$)
 - $xJogPos = TRUE$ (manual jog is to be executed.)
2. Enable axis.
 - $xRegulatorOn = TRUE$
==> "READY" state ($xAxisEnabled =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

3.3 Function block L_TT1P_CrossCutter[Base/State]

The figure shows the relation of the inputs and outputs to the "Base" and "State" versions.

The additional inputs and outputs of the "State" version are shaded.





3.3.1 Inputs and outputs

Designator	Data type	Description	Available in version	
			Base	State
MasterAxis	AXIS_REF	Reference to the master axis (master axis)	●	●
SlaveAxis	AXIS_REF	Reference to the slave axis	●	●

3.3.2 Inputs

Designator	Data type	Description		Available in version	
				Base	State
xEnableInternalControl	BOOL	TRUE	In the visualisation, the internal control of the axis can be selected via the "Internal Control" axis.	●	●
xEnable	BOOL	Execution of the function block		●	●
		TRUE	The function block is executed.		
		FALSE	The function block is not executed.		
scCtrlABC	scCtrl_ABC	Input structure for the L_MC1P_AxisBasicControl function block • <i>scCtrlABC</i> can be used in the "Ready" state. • If there is a request, the state changes to "Service". • The state change from "Service" back to "Ready" takes place if there are no more requests.		●	●
xResetError	BOOL	TRUE	Reset axis error or software error. In the State version, the first touch probe mark subsequently has to be saved again with the teaching function.	●	●
xRegulatorOn	BOOL	TRUE	Activate controller enable of the axis (via the MC_Power function block).	●	●
xStop	BOOL	TRUE	Cancel the active movement and brake the axis to a standstill with the deceleration defined via the <i>IrStopDec</i> parameter. • The state changes to "Stop". • The STOP state is exited if (Not xStop AND Not xHalt) AND eAxisState = StandStill • The input is also active with "Internal Control".	●	●
xHalt	BOOL	TRUE	Cancel the active movement and brake the axis to a standstill with the deceleration defined via the <i>IrHaltDec</i> parameter. • The state changes to "Stop". • The technology module remains in the "Stop" state as long as <i>xStop</i> is set to TRUE (or <i>xHalt</i> = TRUE).	●	●
scPar	L_TT1P_scPar_CrossCutter[Base/State]	The parameter structure contains the parameters of the technology module. The data type depends on the version used (Base/State).		●	●
scAccessPoints	L_TT1P_scAP_CrossCutter[Base/State]	Structure of the access points The data type depends on the version used (Base/State).		●	●
xJogPos	BOOL	TRUE	Traverse axis in positive direction (manual jog). If <i>xJogNeg</i> is also TRUE, the traversing direction selected first remains set.	●	●
xJogNeg	BOOL	TRUE	Traverse axis in negative direction (manual jog). If <i>xJogPos</i> is also TRUE, the traversing direction selected first remains set.	●	●
xHomeExecute	BOOL	The input is edge-controlled and evaluates the rising edge.		●	●
		FALSE	Start homing.		
xHomeAbsSwitch	BOOL	TRUE	The function is aborted via the xStop input.	●	●
		TRUE	Connection for reference switch: For homing modes with a reference switch, connect this input to the digital signal which maps the state of the reference switch.		
xExecuteMoveToParkPos	BOOL	The input is edge-controlled and evaluates the rising edge.		●	●
		FALSE	Execute positioning of the slave axis to the parking position.		
		TRUE			

Designator	Data type	Description		Available in version	
				Base	State
xSyncCrossCutter	BOOL	FALSE	Clutching in to the cam	●	●
		TRUE	Decutching to the parking position		
xSyncOutInstant	BOOL	TRUE	Immediate declutching from the cam - The slave axis is brought to a standstill with the deceleration from the IrSyncOutInstantDec parameter. - The coupling mode in the "eSyncMode" parameter has no impact here.	●	●
xHoldInParkPos	BOOL		Rejects process (leaving out cuts)	●	●
		TRUE	After the synchronous phase, the slave axis is declutched to the parking position. - The cutting process is not continued. - The product position (position of the X axis) at the IrSetXPosOut output, however, is still calculated.		
		FALSE	After the next synchronous phase, the slave axis is clutched in to the subsequent synchronous phase. The cutting process is continued.		
IrProductLength	LREAL	The length of the product or material, depending on the operating mode in the cutting process (eCuttingMode parameter): 0: Length cutting The product length corresponds to the cutting length. 1: Flow packer The product length is only used to calculate the speed in the synchronous phase. The cut is always effected at the cycle end of the master axis. 2: Length cutting with mark correction The product length corresponds to the register length and is taken over once before the cutting process. The cutting length is set once within the product length via the array alrCutLength[1..10] in the parameter structure L_TT1P_scPar_CrossCutter[Base/State] (23). 3: Cutting on a mark (event-controlled) The product length is not evaluated in this mode. - Unit: Unit of the master axis (e.g. mm) - Initial value = 0.0		●	●
IrSetOverspeedScaled	LREAL	Factor for speeding during the synchronous phase The scaled value refers to the current material speed (master speed). - Example: IrSetOverspeedScaled = 1.0 means that the tool (e.g. cutting blade) has twice the circumferential speed than the material speed. - Initial value = 0.0 - Value range: -0.9 ... 2.0		●	●
IrSetOffsetMaster	LREAL	Position offset of the master axis The resulting X position of the cam is produced by addition of the master axis position to the "IrSetOffsetMaster" offset. - Unit: units - Initial value: 0		●	●
xTrimPos	BOOL	TRUE	Trim position profile or speed profile of the axis in positive direction. - The trimming type (position profile/speed profile) can be set via the eTrimMode parameter. - If xTrimNeg is also TRUE, the traversing direction selected first remains set.	●	●

Designator	Data type	Description		Available in version	
				Base	State
xTrimNeg	BOOL	TRUE	Trim position profile or speed profile of the axis in negative direction. - The trimming type (position profile/speed profile) can be set via the eTrimMode parameter. - If xTrimPos is also TRUE, the traversing direction selected first remains set.	●	●
xResetTrimOffset	BOOL	TRUE	The offset of the trimming function (lrOffsetTrim output) is set to zero.	●	●
xTeachMarkWindow	BOOL	TRUE	The mark position (touch probe window) is referenced. (Teaching function for setting-up operation) - Behaviour in <u>declutched</u> state: When a touch probe mark has been detected, the touch probe window is adjusted to the position of the detected mark and activated. In addition, the register position is set to the value of the internally calculated setpoint position of the sensor. - Behaviour in <u>clutched-in</u> state: When a touch probe mark has been detected, the touch probe window is adjusted to the position of the detected mark and activated. After successful execution, the xMarkWindowTeached output is set to TRUE.		●
xMarkReceive	BOOL	TRUE	A touch probe mark has been detected in the connected touch probe sensor.		●
lrActMarkPosIn	LREAL		Current position of the touch probe mark with regard to the axis reference used. Unit: units		●
lrCycPosExtEncoder	LREAL		Cyclic position of the external encoder in case the touch probe from the encoder axis is used. (eTpMode parameter = 2: External encoder) - Unit: units - Initial value = 0.0		●
xGearFactorCorr	BOOL	TRUE	Activation of the gearbox factor correction (compensation of deviating register lengths) The speed setpoint is corrected by the medium difference of the touch probe correction values.		●
xGearFactorCorrHold	BOOL	TRUE	The current gearbox factor correction value is held.		●

3.3.3 Outputs

Designator	Data type	Description		Available in version	
				Base	State
xInternalControlActive	BOOL	TRUE	The internal control of the axis is activated via the visualisation. (xEnableInternalControl input = TRUE)	●	●
eTMState	L_TT1P_States	Current state of the technology module ► State machine (□ 28)		●	●
scStatusABC	scStatus_ABC	Structure of the status data of the L_MC1P_AxisBasicControl function block		●	●
xError	BOOL	TRUE	There is an error in the technology module.	●	●
xWarning	BOOL	TRUE	There is a warning in the technology module.	●	●
eErrorID	L_IE1P_Error	ID of the error or warning message if xError = TRUE or xWarning = TRUE. L_IE1P_Error		●	●
scErrorInfo	L_IE1P_Error	Error information structure for a more detailed analysis of the error cause		●	●
scSignalFlow	L_TT1P_scSF_CrossCutter[Base/State]	Structure of the signal flow The data type depends on the version used (Base/State). ► Signal flow diagrams (□ 31)		●	●
xAxisEnabled	BOOL	TRUE	The axis is enabled.	●	●
xDone	BOOL	TRUE	The request/action has been completed successfully.	●	●
xBusy	BOOL	TRUE	The request/action is currently being executed.	●	●
xisHomed	BOOL	TRUE	The axis has been referenced (reference known).	●	●
xInParkPos	BOOL	TRUE	The axis is in the parking position. This output can be set by different functions: - Positioning into the parking position; - Holding the axis in the parking position during the cutting process; - The cutting length in the cutting process is so long that the axis comes to a stop in the parking position; - Decutching to the parking position with the input xSyncCrossCutter = FALSE.	●	●
xSynchronised	BOOL	TRUE	The axis is synchronised to the cam.	●	●
xAccDecSync	BOOL	TRUE	The synchronisation function is active. The axis is synchronised or desynchronised (clutch opens or closes).	●	●
xHoldInParkPosActive	BOOL	TRUE	During the rejects process (xHoldInParkPos input = TRUE), the slave axis is in the parking position.	●	●
xCuttingActive	BOOL	TRUE	The tool (e.g. cutting blade) at the axis is in the cutting area.	●	●
xCutDone	BOOL	TRUE	The cut has been effected. The rising edge at the xCutDone output (FALSE→TRUE), the cut length of the material is calculated and provided at the lActCutLength output.	●	●

Designator	Data type	Description	Available in version	
			Base	State
IrActCutLength	LREAL	Cut length of the material The cut length results from the difference between the current position of the master axis and the current cutting position of the sealing roller. • Unit: mm	●	●
bySetToolNoOut	BYTE	The number of the tool (e.g. cutting blade) used for cutting when the synchronisation function is active (xAccDecSync output = TRUE). The first tool is in the zero position of the axis. Further tools on the circumference of the axis are numbered consecutively in positive direction of rotation.	●	●
bySetAreaOut	BYTE	The number of the area between two tools on the circumference of the axis The first area is between the first and second tool.	●	●
IrOffset	LREAL	Position offset of the master axis from the IrSetOffsetMaster input The resulting X position of the cam is produced by addition of the master axis position to the "IrSetOffsetMaster" offset. • Unit: units	●	●
IrOffsetTrim	LREAL	Position offset from the trimming function between the master axis and the X axis from the cam • Unit: units	●	●
IrOffsetTotal	LREAL	Total position offset between the master axis and the X axis from the cam • Unit: units	●	●
IrSetXPosOut	LREAL	Position of the X axis from the cam The X axis always runs from 0 to the cutting length. • Unit: units	●	●
IrSetXVelOut	LREAL	Velocity of the X axis from the cam • Unit: units/s	●	●
IrSetXAccOut	LREAL	Acceleration of the X axis from the cam • Unit: units/s ²	●	●
IrActSlaveFollowingError	LREAL	Current following error of the slave axis • Unit: units/s ²	●	●
IrActSlavePos	LREAL	Current position of the slave axis • Unit: units	●	●
IrActSlaveVel	LREAL	Current velocity of the slave axis • Unit: units/s	●	●
IrSlaveVelMax	LREAL	Maximum velocity of the slave axis This value will be reached if the master axis is executed in the IrMasterVelMax parameter with maximum speed. The calculation is carried out when the xCamBounds input is set to TRUE. • Unit: units/s	●	●
IrSlaveAccMax	LREAL	Maximum acceleration of the slave axis This value will be reached if the master axis is executed in the IrMasterVelMax parameter with maximum speed and maximum acceleration in the IrMasterAccMax parameter. The calculation is carried out when the xCamBounds input is set to TRUE. • Unit: units/s ²	●	●
xTrqCtrlActive	BOOL	TRUE The torque feedforward control is activated.		●
IrSetMotorPreCtrlTrqOut	LREAL	Setting of the torque feedforward control value The torque refers to the output end of the gearbox (gearbox output). • Unit: Nm		●

Designator	Data type	Description		Available in version	
				Base	State
xRegisterCtrlActive	BOOL	TRUE	The register control is activated.		●
xTeachMarkWindowActive	BOOL	TRUE	The homing of the mark position (touch probe window) is activated.		●
lrSetRegPosOut	LREAL	Output setpoint position of the register control The register position is always within a rotary modulo cycle with the cycle length of the lrProductLength input.			●
lrMarkSensorPos	LREAL	The internally calculated position within the register cycle on which the touch probe mark is expected. • Unit: units			●
lrRegMarkPos	LREAL	The converted actual position of the current touch probe mark within the register cycle • Unit: units			●
lrActMarkError	LREAL	Current deviation between the real position of the touch probe mark and the expected position of the touch probe mark • Unit: units			●
lrActMarkDist	LREAL	Register length between the last two touch probe marks • Unit: units			●
xMarkReceived	BOOL	TRUE	A touch probe mark has been detected. If the mark register is used in the State version, xMarkReceived is always set when the mark register has been corrected.		●
xCorrActive	BOOL	TRUE	Activate compensating movement.		●
xCorrLimited	BOOL	TRUE	The touch probe difference is limited to the maximum value.		●
xMarkWindowOpen	BOOL	TRUE	Touch probe window open. A valid touch probe mark has been detected.		●
xMarkOutOfWindow	BOOL	TRUE	A touch probe mark has been detected outside of the touch probe window.		●
xMarkNotDetected	BOOL	TRUE	No touch probe mark has been detected within the touch probe window.		●
xMarkWindowTeached	BOOL	TRUE	Homing of the touch probe window is completed.		●
lrAverageGearFactorCorr	LREAL	Effective gearbox factor for the gearbox factor correction			●
xGearFactorCorrLimited	BOOL	TRUE	The gearbox factor correction is limited.		●

3.3.4 Parameters

L_TT1P_scPar_CrossCutter[Base/State]

The L_TT1P_scPar_CrossCutter[Base/State] structure contains the parameters of the technology module.

Designator	Data type	Description	Available in version	
			Base	State
IrStopDec	LREAL	Deceleration for the stop function and when hardware/software limit switches and the following error monitoring function are triggered • Unit: units/s ² • Initial value: 10000	●	●
IrStopJerk	LREAL	Jerk for the stop function and for the triggering of the hardware limit switches, software limit positions, and the following error monitoring function • Unit: units/s ³ • Initial value: 100000	●	●
IrHaltDec	LREAL	Deceleration for the holding function Specification of the maximum speed variation which is to be used for deceleration to standstill. • Unit: units/s ² • Initial value: 3600 • Only positive values are permissible.	●	●
IrJerk	LREAL	Jerk for compensating an offset value, trimming, clutch, or holding function • Unit: units/s ³ • Initial value: 100000	●	●
IrJogJerk	LREAL	Jerk for manual jog • Unit: units/s ³ • Initial value: 10000	●	●
IrJogVel	LREAL	Maximum speed to be used for manual jog. • Unit: units/s • Initial value: 10	●	●
IrJogAcc	LREAL	Acceleration for manual jog Specification of the maximum velocity variation which is to be used for acceleration. • Unit: units/s ² • Initial value: 100	●	●
IrJogDec	LREAL	Deceleration for manual jog Specification of the maximum speed variation which is to be used for deceleration to standstill. • Unit: units/s ² • Initial value: 100	●	●
IrHomePosition	LREAL	Home position for a reference run (homing) • Unit: units • Initial value: 0	●	●
xUseHomeExtParameter	BOOL	Selection of the homing parameters to be used • Initial value: FALSE	●	●
		FALSE The homing parameters defined in the axis data are used.		
		TRUE The scHomeExtParameter homing parameters from the application are used.		
scHomeExtParameter L_MC1P_HomeParameter		Homing parameters from the application • Only relevant if xUseHomeExtParameter = TRUE.	●	●

Designator	Data type	Description	Available in version	
			Base	State
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.	●	●
eTrimMode L_TT1P_TrimMode		Type of trimming for positive or negative direction (inputs xTrimPos or xTrimNeg) • Initial value: 0	●	●
	0	Trimming via velocity profile (with velocity in lrOffsetTrimVel parameter)		
	1	Trimming via positioning profile (with increment in lrTrimDist parameter)		
lrTrimDist LREAL		Increment for trimming via positioning profile (eTrimMode = 0) • Unit: mm • Initial value: 3	●	●
lrOffsetTrimVel LREAL		Velocity offset for compensating an offset change (lrSetOffsetMaster input) or the increment for trimming via the positioning profile (lrTrimDist parameter) • Unit: units/s • Initial value: 10	●	●
lrOffsetTrimAcc LREAL		Acceleration offset for compensating an offset change (lrSetOffsetMaster input) or the increment for trimming via the positioning profile (lrTrimDist parameter) • Unit: units/s ² • Initial value: 100	●	●
lrOffsetTrimDec LREAL		Deceleration offset for compensating an offset change (lrSetOffsetMaster input) or the increment for trimming via the positioning profile (lrTrimDist parameter) • Unit: units/s ² • Initial value: 100	●	●
ePositioningDirection MC_DIRECTION		Direction of rotation for positioning the sealing roller Use for positioning into the parking position or onto the cam. • Initial value: MC_DIRECTION.positive	●	●
lrPositioningVel LREAL		Velocity for positioning the sealing roller Use for positioning into the parking position or onto the cam. • Unit: units/s • Initial value: 10	●	●
lrPositioningAcc LREAL		Acceleration for positioning the sealing roller Use for positioning into the parking position or onto the cam. • Unit: units/s ² • Initial value: 100	●	●
lrPositioningDec LREAL		Deceleration for positioning the sealing roller Use for positioning into the parking position or onto the cam. • Unit: units/s ² • Initial value: 100	●	●
eSyncMode L_TT1P_SyncModeCrossCutter		Synchronisation of the cross cutter	●	●
	0	Cam_in: Clutch in cross cutter to moving material.		
	1	Positioning: Position the cross cutter to the stationary material.		

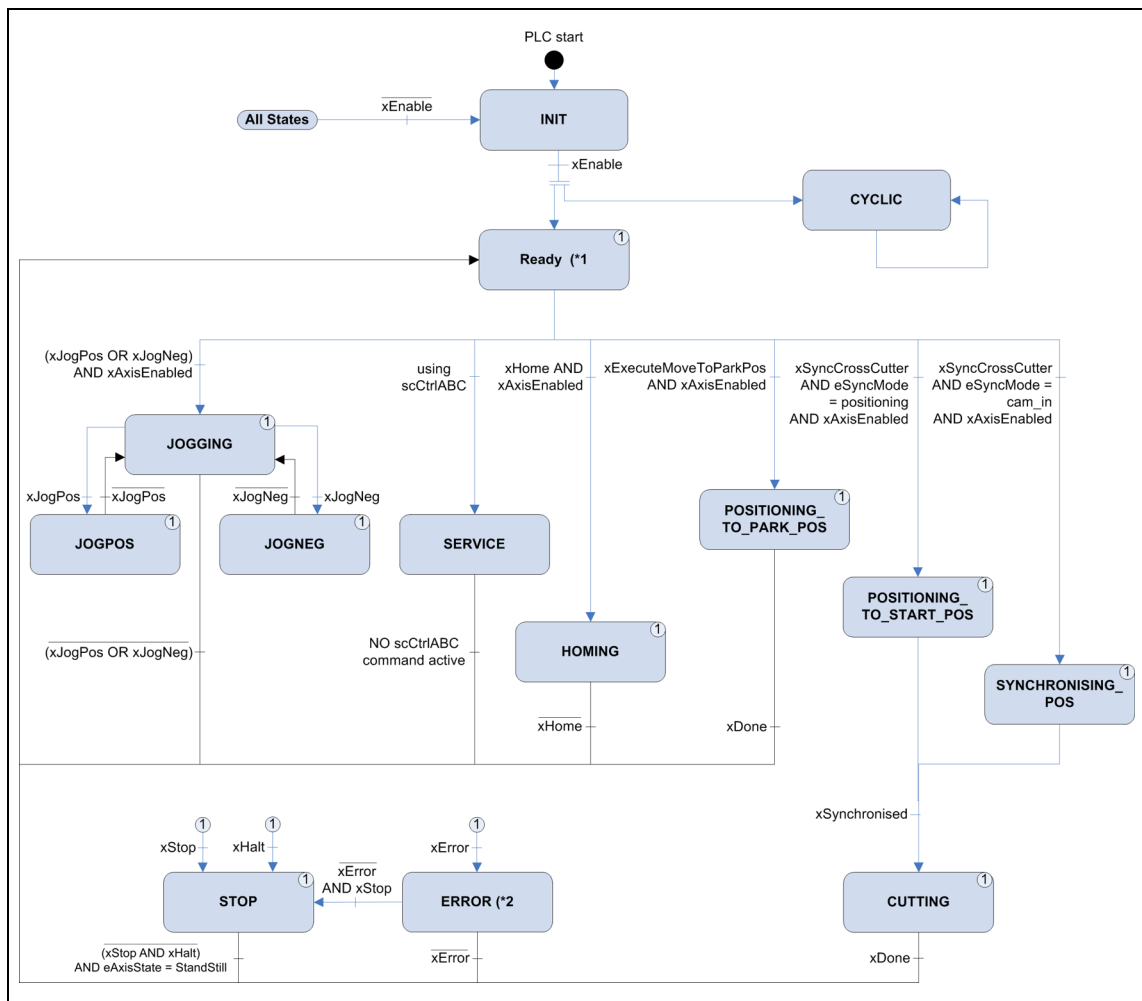
Designator	Data type	Description	Available in version	
			Base	State
IrMasterSyncInDist	LREAL	Clutch-in distance of the master axis in the "length cutting" operating mode (eCuttingMode parameter = 0) for the synchronisation mode "Cam_in" (eSyncMode parameter = 0) For the <i>IrMasterSyncInDist</i> setting = 0, an optimum clutch-in distance is calculated. • Initial value: 0	●	●
IrMasterSyncOutDist	LREAL	Declutch distance of the master axis in the "length cutting" operating mode (eCuttingMode parameter = 0) for the synchronisation mode "Cam_in" (eSyncMode parameter = 0) For the setting <i>IrMasterSyncOutDist</i> = 0, an optimum declutch distance is calculated. • Initial value: 0	●	●
IrSyncOutInstantDec	LREAL	Deceleration for declutching from the cam when the xSyncOutInstant input is set to TRUE. • Unit: units/s ² • Initial value: 1000	●	●
xMasterAbsolute	BOOL	Reference to the position of the master axis • Initial value: TRUE	●	●
		TRUE Absolute reference between the position of the master axis and the X axis of the cam		
		FALSE Relative reference: The X starting position is loaded from the IrStarttXPosRelative parameter.		
IrStartXPosRelative	REAL	Starting position for the X axis of the cam when setting the xMasterAbsolute = FALSE • Unit: units • Initial value: 0	●	●
byStartToolNo	BYTE	Number of the tool (e.g. cutting blade) on the sealing roller which executes the first cut. • Initial value: 0 (automatic tool selection)	●	●
byNumOfTools	BYTE	Number of the tools on the sealing roller The tools are spread symmetrically on the circumference of the roller. • Initial value: 1	●	●
IrCuttingAngle	LREAL	Angle of the synchronous area of the sealing roller • Unit: Degree • Value range: from '0' to 360 / number of tools (byNumOfTools) • Initial value: 10	●	●
IrCrossCutterCircumference	LREAL	Circumference of the sealing roller • Unit: Unit of the master axis (e.g. mm) • Initial value: 75.55555	●	●
IrScalingMaster	LREAL	X extension/compression factor (scaling factor) of the master axis Thus, the X position of the cam results from the multiplication of the master position by IrScalingMaster • The scaling factor is taken over in the "Ready" state. • Negative values are not permitted. • Initial value: 1	●	●
xCalcCamBounds	BOOL	TRUE Extreme values of the slave axis (IrSlaveMaxVel, IrSlaveMaxAcc) are calculated as a function of the parameters IrMasterMaxVel and IrMasterMaxAcc. • Initial value: FALSE	●	●
IrMasterMaxVel	LREAL	Maximum speed of the master axis for checking the cams • Unit: units/s • Initial value: 100	●	●

Designator	Data type	Description	Available in version	
			Base	State
lrMasterMaxAcc	LREAL	Maximum acceleration of the master axis for checking the cams • Unit: units/s ² • Initial value: 1000	●	●
eCuttingMode	L_TT1P_CuttingMode [Base/State]	Selection of the operating mode in the cutting process • Initial value: 0 (length cutting)	●	●
	0	Length cutting (initial value)		
	1	Flow packer		
	2	Length cutting with mark correction		
	3	Cutting on a mark (event-controlled)		
alrCutLengthInProduct (ARRAY[1..10] OF LREAL)		Cutting lengths within the product length (lrProductLength input) Use only in the "length cutting operating mode with mark correction": eCuttingMode parameter = 2 • Initial value: 0 (cutting lengths deactivated)		●
lrMinProductLength	LREAL	Minimum cutting length and therefore the minimum distance between two (touch probe) marks Use only in the "cutting on a mark (event-controlled)": eCuttingMode parameter = 3 • Unit: units • Initial value: 50		●
lrTrqCtrlGain	LREAL	Gain factor for the torque feedforward control • Initial value: 0 (torque feedforward control deactivated)		●
lrTrqCtrlRateTime	LREAL	Rate time for the torque feedforward control • Unit: s • Initial value: 0		●
lrMIInertia	LREAL	Moment of inertia for the torque feedforward control with regard to the sealing rollers at the gearbox output of the motor. • Unit: kg/cm ² • Initial value: 0 (torque feedforward control deactivated)		●
lrSensorToolDist	LREAL	Distance of the touch probe sensor to the attack position of the tool (e.g. cutting blade) on the material This parameter is required to automatically calculate the mark register. If touch probe sensor and axis are within one register cycle, this value can be set to "0". • Unit: units • Initial value: 0		●
xMarkCorrection	BOOL	TRUE Activate correction of the touch probe deviation. • Initial value: TRUE		●
lrMarkWindowSize	LREAL	Size of the touch probe window with regard to the X axis of the cam. The touch probe window is put symmetrically around the expected touch probe position. • Unit: units • Initial value: 20		●
eTPMode	L_TT1P_TpMode	Touch probe source • Initial value: 0 (master axis)		●
	0	Master axis		
	1	Slave axis		
	2	External encoder		
lrCycleLengthExtEncoder	LREAL	Cycle time for the external encoder (Only relevant if eTPMode parameter is set = "2: External encoder") • Unit: s • Initial value: 360		●

Designator	Data type	Description	Available in version	
			Base	State
IrMaxCorrPos	LREAL	Maximum positive correction distance per register cycle (X units) • Unit: units • Initial value: 30.0		●
IrMaxCorrNeg	LREAL	Maximum negative correction distance per register cycle (X units) • Unit: units • Initial value: -30.0		●
IrGearFactorCorrGain	LREAL	Gain factor of the gearbox factor correction • Initial value: 0.1		●
IrMaxGearFactorCorr	LREAL	Maximum deviation of the gearbox factor correction • Unit: units • Initial value: 10		●
dwMaxMissedMarks	DWORD	Maximum number of permitted touch probe failures If no touch probe is detected within the touch probe window, an artificial mark is generated. This happens as long as the number of marks set here is not exceeded. If the number of marks was exceeded, the xError output is set to TRUE. • Initial value: 5		●
IrSetOffsetMarkWindow	LREAL	Offset for moving the touch probe window relative to the register measurement system. • Unit: mm • Initial value: 0		●

3.4

State machine

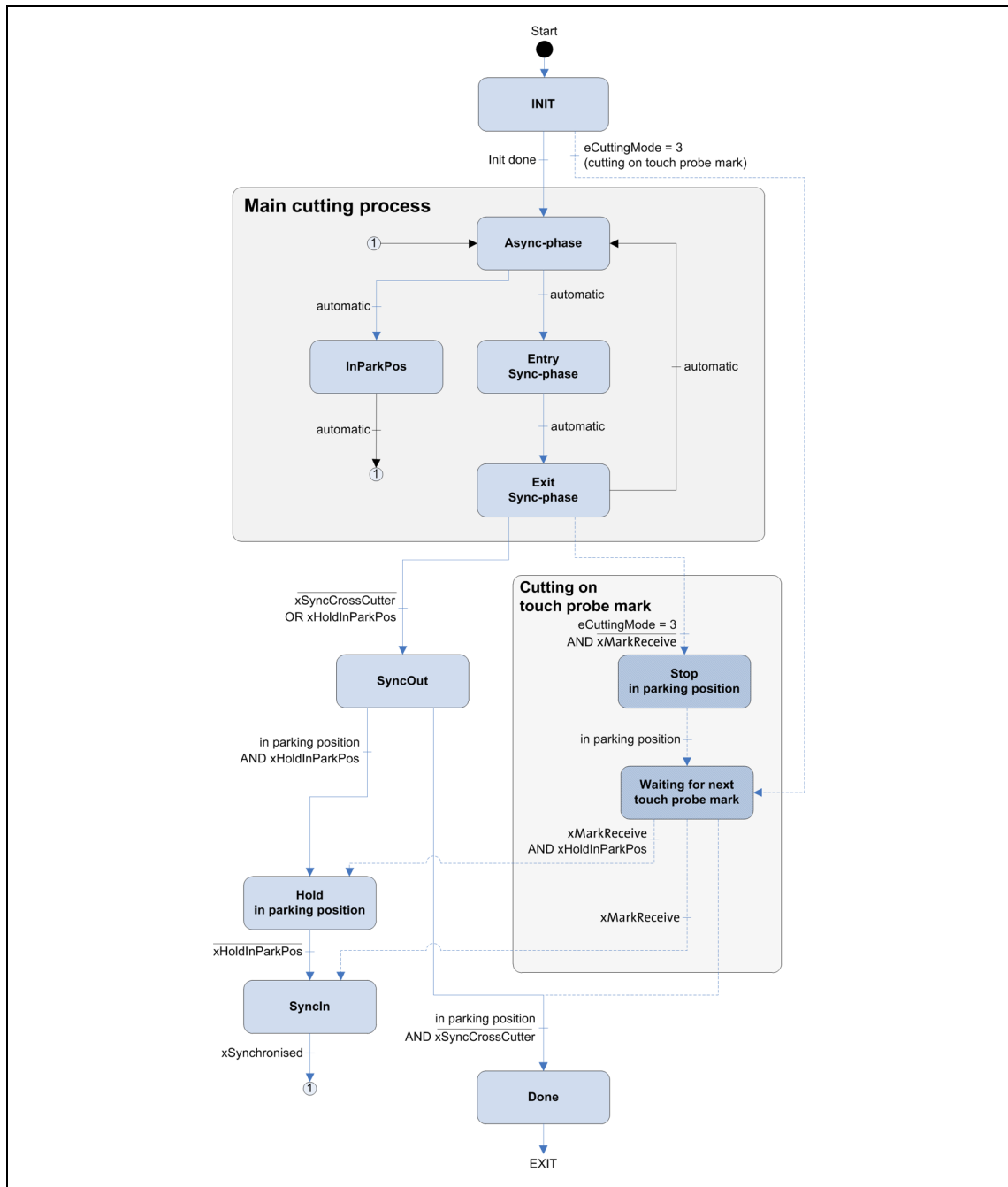


[3-3] State machine of the technology module

① A transition to the ERROR or STOP state is possible.

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

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



[3-4] State machine for the cutting process

States of the eTMState output (L_TT1P_States)

No.	L_TT1P_States	Description
1	INIT	Initialisation of the technology module active.
2	READY	Technology module ready for operation.
3	HOMING	Referencing active.
10	JOGGING	Manual jog active.
11	JOGPOS	Manual jog in positive direction active.
12	JOGNEG	Manual jog in negative direction active.
32	POSITIONING_TO_START_POS	Movement to start position active.
34	POSITIONING_TO_TARGET_POS	Movement to target position active.
35	POSITIONING_TO_PARK_POS	Movement to park position active.
170	CUTTING	Cutting process active.
171	IN_PARK_POS	Tool located in park position.
179	WAIT_OF_NEXT_CUT	Waiting for the next cut in park position.
996	STOP	Stop/Halt active.
998	SERVICE	The technology module is in service mode. The internal L_MC1P_AxisBasicControl function block is controlled via the <i>scCtrlABC</i> input structure. The status of the function block can be viewed via the <i>scStatusABC</i> output structure.
999	ERROR	Error state
1000	SYSTEMFAULT	System fault

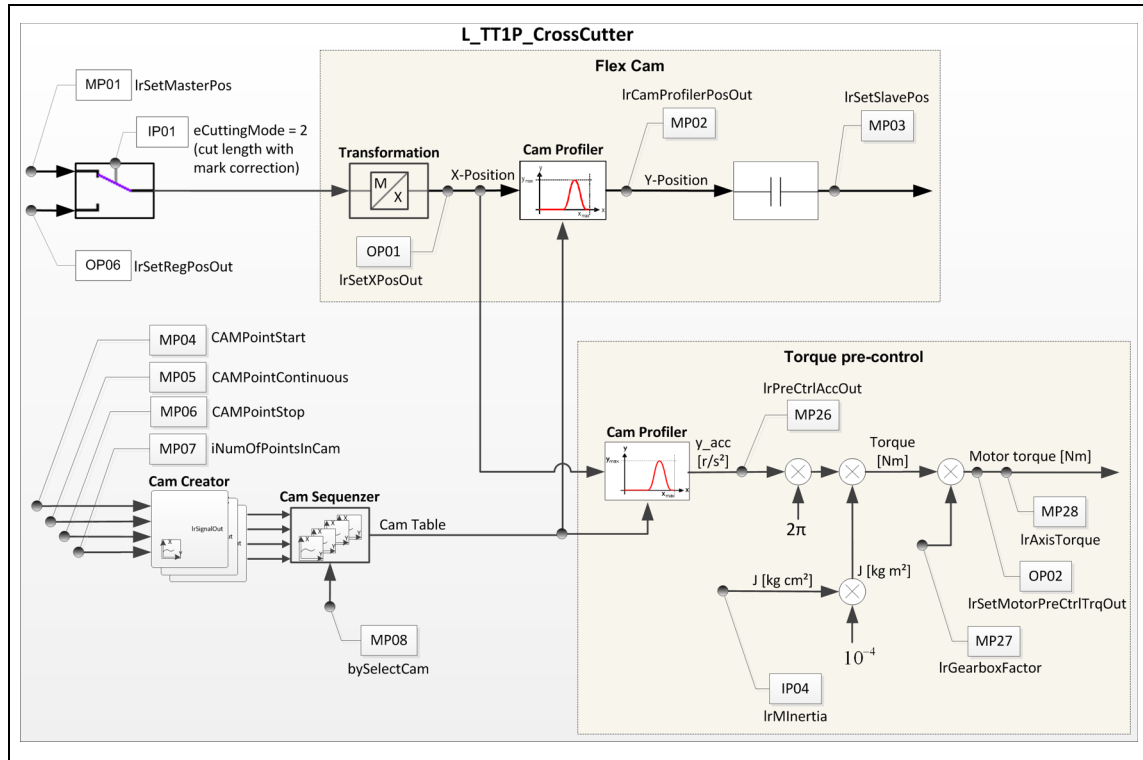
3 Functional description of "Cross Cutter"

3.5 Signal flow diagrams

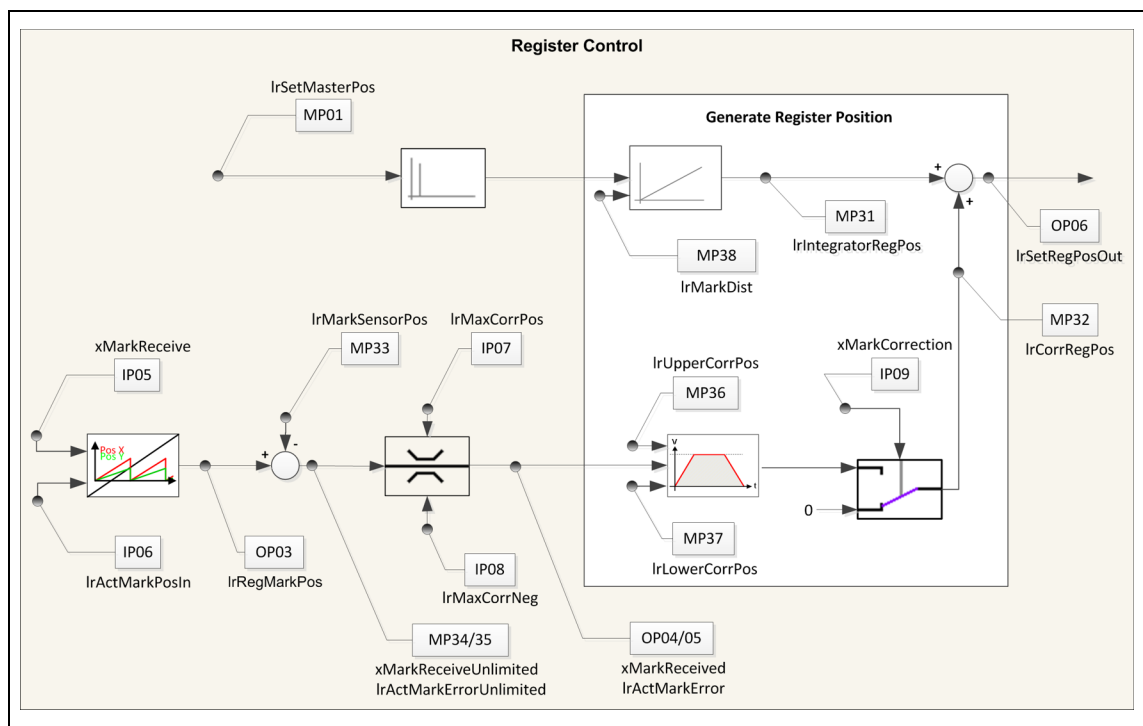
3.5 Signal flow diagrams

The illustrations show the main signal flow of the functions implemented.

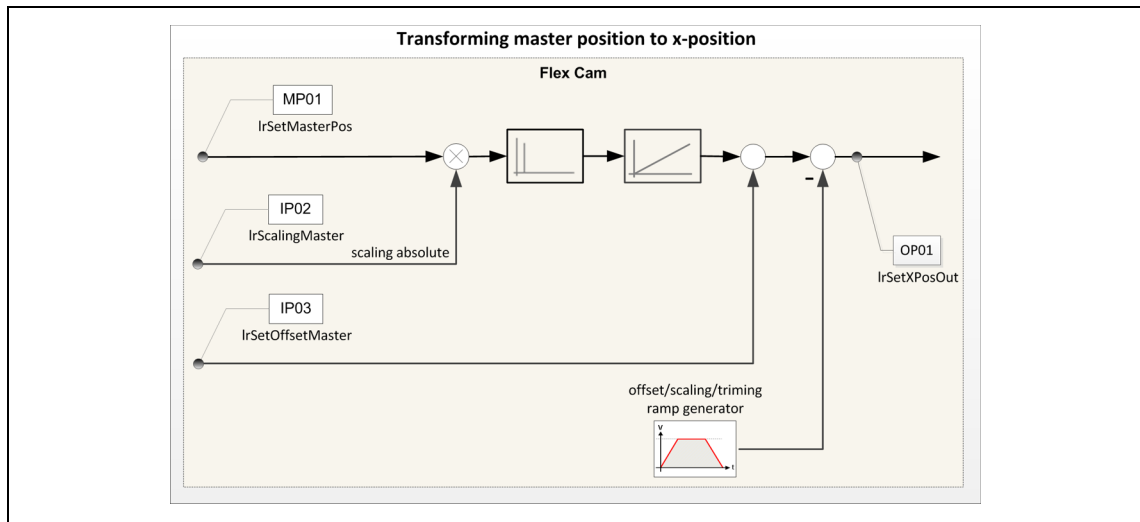
The signal flow of the additional functions such as "manual jog" is not displayed here.



[3-5] Signal flow: Cross Cutter - cam and torque feedforward control



[3-6] Partial signal flow: Register Control



[3-7] Partial signal flow: Transform position of the master axis to X position

3.5.1 Structure of the signal flow

L_TT1P_scSF_CrossCutter[Base/State]

The contents of the **L_TT1P_scSF_CrossCutter[Base/State]** structure is read-only and offer a practical diagnostics option within the signal flow ([Signal flow diagrams \(□ 31\)](#)).

Designator	Data type	Description		Available in version	
				Base	State
IP01_eCuttingMode L_TT1P_CuttingMode [...]		Selection of the operating mode in the cutting process		●	●
		• Initial value: 0 (length cutting)			
		0	Length cutting		
		1	Flow packer		
		2	Length cutting with mark correction		
		3	Cutting on a mark (event-controlled)		
IP02_lrScalingMaster LREAL		X extension/compression factor (scaling factor) of the master axis Thus, the X position of the cam results from the multiplication of the master position by lrScalingMaster		●	●
		• The scaling factor is taken over in the "Ready" state.			
		• Negative values are not permitted.			
		• Initial value: 1			
IP03_lrSetOffsetMaster LREAL		Position offset of the master axis The resulting X position of the cam is produced by addition of the master axis position to the "lrSetOffsetMaster" offset.		●	●
		• Unit: units			
		• Initial value: 0			
IP04_lrMIInertia LREAL		Moment of inertia for the torque feedforward control with regard to the sealing rollers at the gearbox output of the motor.			●
		• Unit: kg/cm ²			
		• Initial value: 0 (torque feedforward control deactivated)			
IP05_xMarkReceived BOOL		TRUE	A touch probe mark has been detected in the connected touch probe sensor.		●
IP06_lrActMarkPosIn LREAL		Current touch probe position with regard to the axis reference used.			●
		• Unit: units			
IP07_lrMaxCorrPos LREAL		Maximum positive correction distance per register cycle (X units)			●
		• Unit: units			
		• Initial value: 30.0			
IP08_lrMaxCorrNeg LREAL		Maximum negative correction distance per register cycle (X units)			●
		• Unit: units			
		• Initial value: -30.0			
IP09_xMarkCorrection BOOL		TRUE	Activate correction of the touch probe deviation.		●
		• Initial value: TRUE			
MP01_lrSetMasterPos LREAL		Set position of the master axis		●	●
		• Unit: units			
MP02_lrCamProfilerPosOut LREAL		The Y position directly from the cam generator		●	●
		• Unit: units			
MP03_lrSetSlavePosOut LREAL		Set position of the slave axis		●	●
		• Unit: units			
MP04_CAMPointStart ARRAY[0...3] OF SMC_CAMXYVA		The interpolation points of the start cam		●	●
MP05_CAMPointContinuous ARRAY[0...5] OF SMC_CAMXYVA		The interpolation points of the cutting cam		●	●

Designator	Data type	Description		Available in version	
				Base	State
MP06_CAMPointStop ARRAY[0...3] OF SMC_CAMXYVA		The interpolation points of the stop cam		●	●
MP07_iNumOfPointsInCam INT		Number of points in the current cam		●	●
MP08_bySelectCam BYTE		Cam used		●	●
		1	Cutting curve		
		2	Cutting curve		
		3	Starting curve		
		4	Stop curve		
MP09_eCuttingProcessState L_TT1P_States		State of the cutting process in the technology module (output eTMState = CUTTING) ► State machine (28)		●	●
MP10_lrYStart LREAL		Internally calculated starting position of the Y axis for the cam		●	●
MP11_lrYPark LREAL		Internally calculated parking position of the Y axis for the cam		●	●
MP12_lrYEnd LREAL		Internally calculated target position of the Y axis for the cam		●	●
MP13_lrXStart LREAL		Internally calculated starting position of the X axis for the cam		●	●
MP14_lrXEndSyncPos LREAL		Internally calculated position of the X axis for the cam in the transition to the asynchronous area		●	●
MP15_lrXStartParkPos LREAL		Internally calculated starting position of the X axis for the parking position of the Y axis for the cam		●	●
MP16_lrXEndParkPos LREAL		Internally calculated target position of the X axis for the parking position of the Y axis for the cam		●	●
MP17_lrXStartSyncPos LREAL		Internally calculated position of the X axis for the cam in the transition to the synchronous area		●	●
MP18_lrXEnd LREAL		Internally calculated target position of the X axis for the cam		●	●
MP19_xXAbsolute BOOL		TRUE	The absolute reference between X axis and master axis is activated or already active.	●	●
MP20_xSlaveAbsolute BOOL		TRUE	The absolute reference between X axis and slave axis is activated or already active.	●	●
MP21_xEndOfProfile BOOL		TRUE	The last cycle in the current profile of the cam is reached/active. • The current values are extrapolated for detection. • The signal is applied for one clock cycle.	●	●
MP22_xEndOfXCycle BOOL		TRUE	The last position on the X axis is reached. • For the identification, the current values are extrapolated. The signal is applied for one clock cycle.	●	●
MP23_xEnablePosition Follower BOOL		TRUE	Enable position follower for the setpoint positions of the slave axis from "Flex Cam".	●	●
MP25_xEnableTorque Follower BOOL		TRUE	Enable torque follower for the torque feedforward control of the slave axis.		●
MP26_lrPreCtrlAccOut LREAL		Internally calculated acceleration from the cam for the next setpoint			●

Designator	Data type	Description		Available in version	
				Base	State
MP27_IrGearboxFactor	LREAL	Transmission factor of torque from cross-cutter axis to motor axis			●
MP28_IrAxisTorque	LREAL	Setpoint torque of the slave axis • Unit: Nm			●
MP30_IrSetVelocityRegister	LREAL	Input velocity of the integrator for creating the register position			●
MP31_IrIntegratorRegPos	LREAL	Position value of the integrator for generating the register position without correction • Unit: units			●
MP32_IrCorrRegPos	LREAL	Position profile of the correction motion (is added to the integrator position of the register) • Unit: units			●
MP33_IrMarkSensorPos	LREAL	Internally calculated position of the touch probe sensor within the register cycle • Unit: units			●
MP34_xMarkReceive Unlimited	BOOL	TRUE	A touch probe signal has been detected <u>before</u> the mark error limitation.		●
MP35_IrActMarkError Unlimited	LREAL	Deviation (touch probe error) <u>before</u> the mark error limitation			●
MP36_IrUpperCorrPos	LREAL	Upper limit value of the correction window for the compensating movement of the touch probe correction The window must not amount to the entire cycle length. • Unit for axis: units • Unit for register: mm • Initial value: 180			●
MP37_IrLowerCorrPos	LREAL	Lower limit value of the correction window for the compensating movement of the touch probe correction • Unit for axis: units • Unit for register: mm • Initial value: 90			●
MP38_IrMarkDist	LREAL	Register length in units of the measuring system of the master axis • Initial value: 360.0			●
OP01_IrSetXPosOut	LREAL	Position of the X axis from the cam • Unit: units			●
OP02_IrSetMotorPreCtrlTrq Out	LREAL	Torque feedforward control value • Unit: Nm			●
OP03_IrRegMarkPos	LREAL	The converted actual position of the current touch probe mark within the register cycle • Unit: units			●
OP04_xMarkReceived	BOOL	TRUE	A touch probe mark has been detected. Base version: • xMarkReceived corresponds to the output at the limitation module. State version: • xMarkReceived corresponds to the output at the touch probe module.		●

Designator	Data type	Description	Available in version	
			Base	State
OP05_IrActMarkError	LREAL	Current deviation between detected touch probe mark and expected touch probe position - Unit: mm Base version: - IrActMarkError corresponds to the touch probe error at the output of the limitation module. State version: - IrActMarkError corresponds to the touch probe error at the output of the touch probe module.		●
OP06_IrSetRegPosOut	LREAL	Output setpoint position of the register control The register position is always within a rotary modulo cycle with the cycle length of the IrProductLength input.		●

3.5.2 Structure of the access points

L_TT1P_scAP_CrossCutter[Base/State]

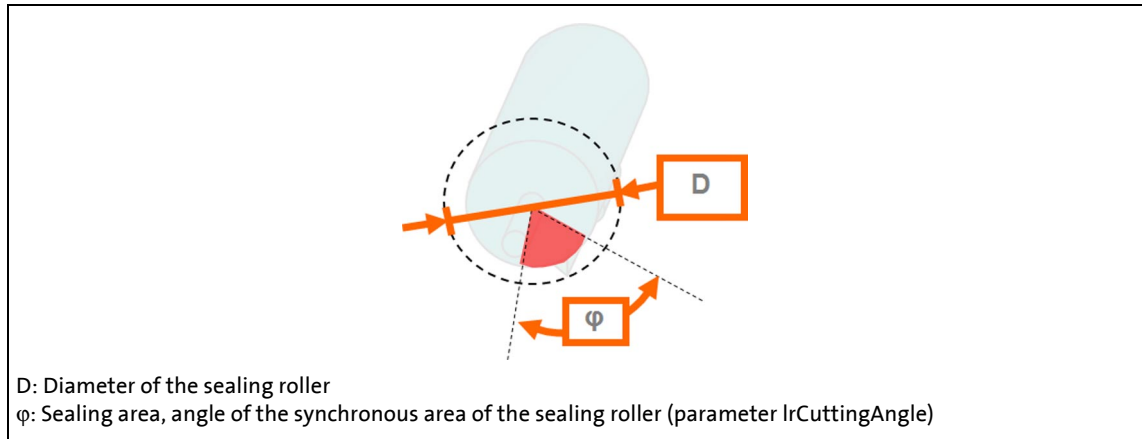
The access points (AP) can be used to influence signals. In the initial state, the access points do not have any effect.

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

Designator	Data type	Description		Available in version	
				Base	State
AP01_xEndOfProfileWindow Size	BOOL	Enable of the AP01_lrEndOfProfileWindowSize access point		●	●
		TRUE	The access point overwrites the values at the access point in the signal flow.		
AP01_lrEndOfProfileWindow Size	LREAL	Tolerance window for a stable/safe detection of the last cycle in the current profile of the cam (MP21: xEndOfProfile) - Within the window, xEndOfProfile = TRUE can only occur once. - Only after the window has been left (after xEndOfProfile = TRUE), the technology module is able to detect the last cycle in the sequence profile. - Unit: units (with reference to the master axis)			
AP02_xOverspeedUpper Limit	BOOL	Enable of the AP02_lrOverspeedUpperLimit access point		●	●
		TRUE	The access point overwrites the values at the access point in the signal flow.		
AP02_lrOverspeedUpper Limit	LREAL	The lrSetOverspeedScaled input serves to define the factor for speeding during the synchronous phase. The factor has the value range -0.9 ... 2.0. With the AP02_lrOverspeedUpperLimit access point, the upper limit can be defined to a value greater than '2.0'.			
AP05_xUpperCorrPos	BOOL	Enable of the AP05_lrUpperCorrPos access point			●
		TRUE	The access point overwrites the values at the access point in the signal flow.		
AP05_lrUpperCorrPos	LREAL	Upper limit value of the correction window with regard to the register position for the compensating movement of the touch probe correction - The window must not amount to the entire cycle length. - Unit for register position: mm			
AP06_xLowerCorrPos	BOOL	Enable of the AP06_lrLowerCorrPos access point			●
		TRUE	The access point overwrites the values at the access point in the signal flow.		
AP06_lrLowerCorrPos	LREAL	Lower limit value of the correction window with regard to the register position for the compensating movement of the touch probe correction - The window must not amount to the entire cycle length. - Unit for register position: mm			

3.6 Parameterisation of the sealing roller

The technology module calculates the cam for synchronous motion of a cross sealing axis and synchronises it with a master axis.



[3-8] Definition of the sealing roller diameter and the sealing area

In order to calculate the seal motion, the circumference (= diameter $\times \pi$) of the sealing roller is specified using the *lrCrossCutterCircumference* parameter, in the master axis unit (e.g. mm). The value cannot be changed during the cutting operation.

The sealing area is parameterised via the *lrCuttingAngle* parameter in the unit (angular) degrees. The sealing area can be changed at any time during operation and thus adapted to the cutting process. New values are always accepted after the synchronous phase.

The permissible value range for the *lrCuttingAngle* parameter goes from a minimum of '0' to a maximum of $360 / \text{number of tools}$ (*byNumOfTools* parameter).

Parameters to be set

The parameters for the sealing roller are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) ([23](#)) parameter structure.

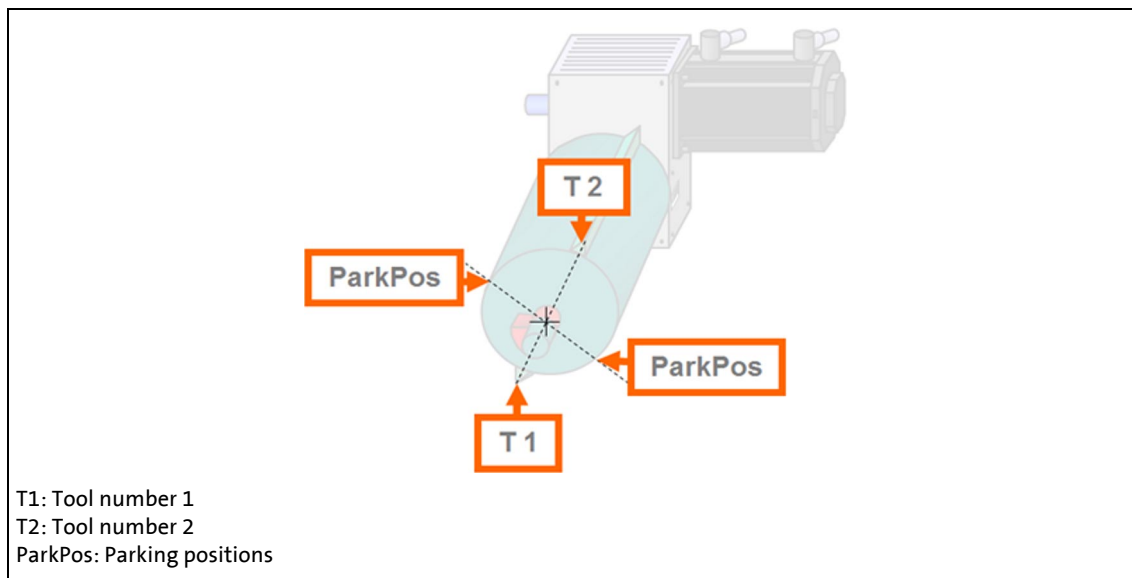
```
lrCrossCutterCircumference : LREAL := 360;
lrCuttingAngle : LREAL := 10;
```

3 Functional description of "Cross Cutter"

3.7 Positioning to the parking position

3.7 Positioning to the parking position

The technology module supports the calculation of the profiles of several tools (e.g. cutting blades) on the sealing roller. The number of tools per sealing roller is specified by the *byNumOfTools* parameter. The first tool is always on the zero coordinate of the sealing roller and is defined as tool number 1. Any further tools are distributed symmetrically on the roller circumference in positive cutting direction. The parking positions are exactly in the middle between two neighbouring tools.



[3-9] Distribution of two tools and parking positions on the sealing roller

The *byStartToolNo* parameter serves to select the tool with the corresponding number which will make the first cut in the material. It also specifies the associated parking position before the selected tool.

If the tools on the sealing roller are not different, set the automatic tool selection using parameter *byStartToolNo* = 0. Then the next possible parking position will be used as target parking position.

With a rising edge (FALSE→TRUE) at the *xExecuteMoveToParkPos* input, the slave axis is positioned to the parking position.

The slave axis is then moved to the parking position using the motion parameters *ePositioningDirection*, *lrPositioningVel*, *lrPositioningAcc* and *lrPositioningDec*.

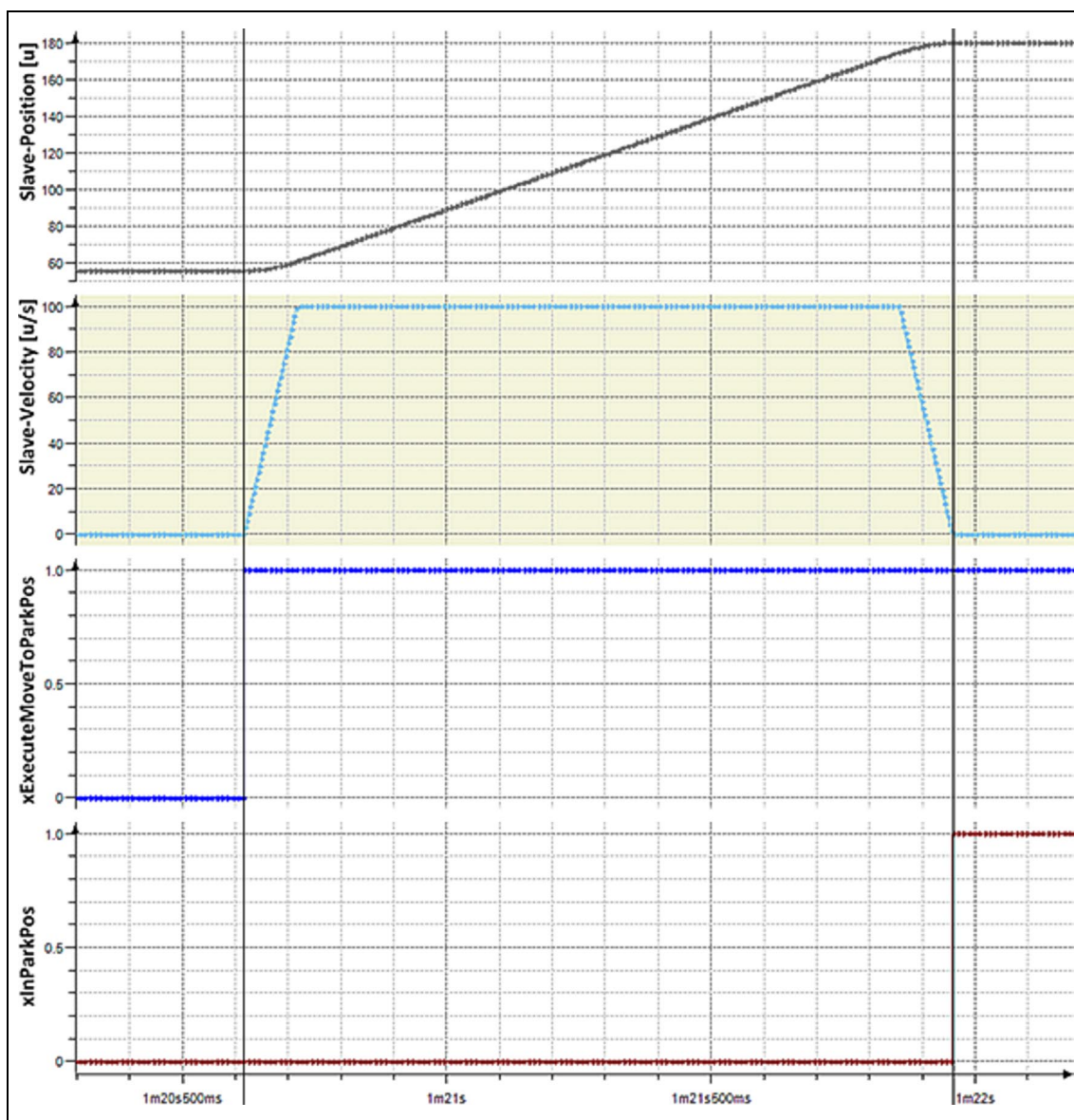
Parameters to be set

The parameters for positioning are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

```
byNumOfTools : BYTE := 1;  
byStartToolNo : BYTE := 0;  
ePositioningDirection : MC_DIRECTION := MC_DIRECTION.positive;  
lrPositioningVel : LREAL := 10;  
lrPositioningAcc : LREAL := 100;  
lrPositioningDec : LREAL := 100;
```

3 Functional description of "Cross Cutter"

3.7 Positioning to the parking position



[3-10] History: Positioning to the parking position for a tool

3.8 Manual jog (jogging)

Precondition

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

Execution

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

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

The current travel command cannot be replaced by another jog command. Only if both inputs have been reset, the [State machine](#) (28) changes to the "Ready" state again.

Parameters to be set

The parameters for the manual jog are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

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

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

3.9 Homing

Precondition

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

Execution

Homing is started with a rising edge ($FALSE \rightarrow TRUE$) at the $xHomeExecute$ input. The axis will be travelling until the home position is reached. After successful homing, the [State machine](#) (28) changes back again to the "Ready" state.

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

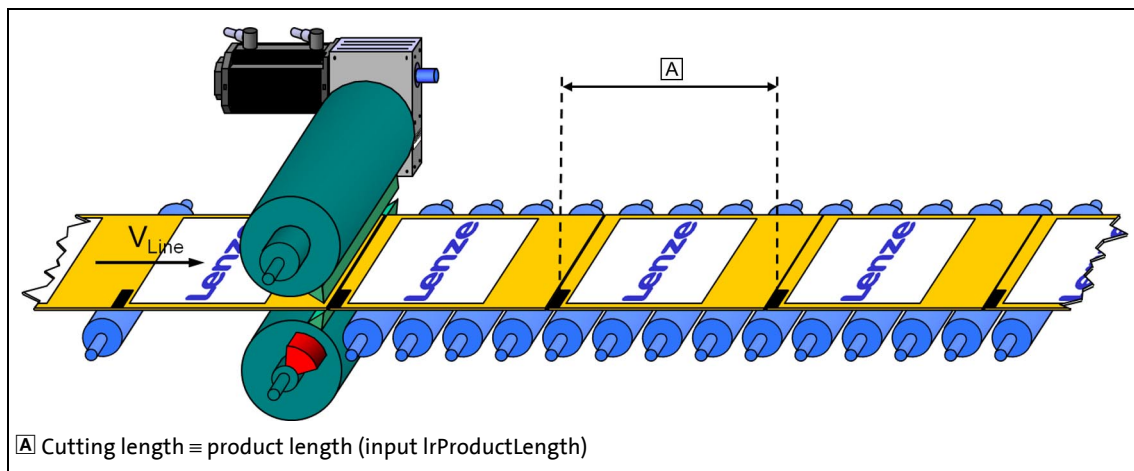
Parameters to be set

The parameters for homing are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

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

3.10

"Length cutting" operating mode



[3-11] Cutting length in the "Length cutting" operating mode

In the "Length cutting" operating mode, the profiles are calculated in real time which provides the possibility to carry out flexible changes to the cutting lengths and the synchronous phase during the cutting process. Any changes are accepted after the synchronous phase and effective for the next cut. The synchronous speed of the sealing roller is calculated as a function of the roller circumference in the `lrCrossCutterCircumference` parameter, the speed of the master axis and the cycle length of the sealing roller.

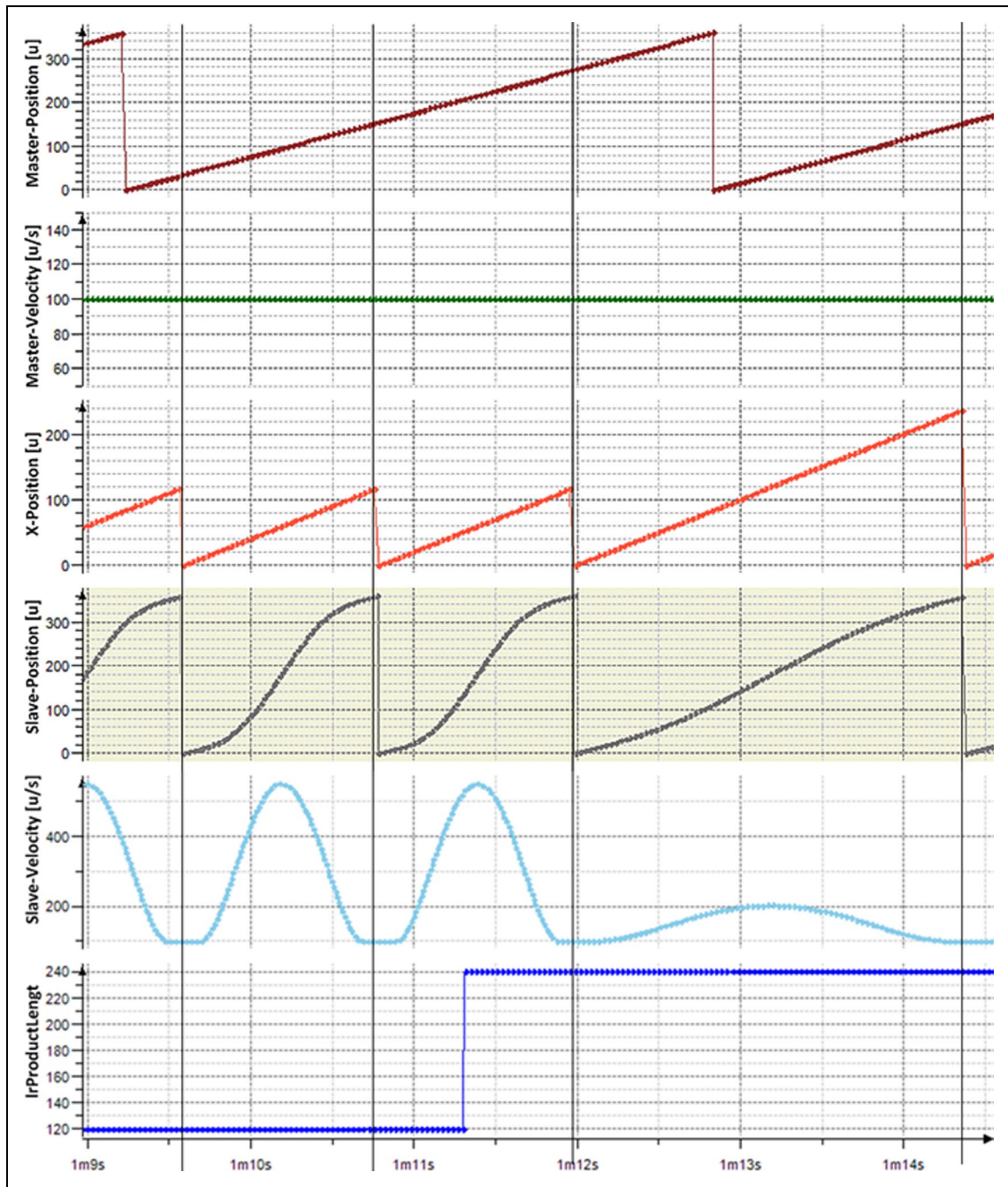
The "Length cutting" operating mode is selected via parameter `eCuttingMode = 0`.

The product/material is cut to the length defined in the `lrProductLength` input. The length is defined in the unit of the master axis (e.g. mm).

Parameters to be set

The parameters for the "length cutting" operating mode are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

```
xMasterAbsolute : BOOL := FALSE;
lrStartXPosRelative : LREAL := 0;
eCuttingMode : L_TT1P_CuttingMode[Base/State] := 0;
lrCrossCutterCircumference : LREAL := 360;
lrCuttingAngle : LREAL := 10;
byNumOfTools : BYTE := 1;
byStartToolNo : BYTE := 0;
```



[3-12] History: Length-controlled cutting process

3.11

Top cut

**Note!**

This function is only supported in the ["Length cutting" operating mode](#) (42) (parameter *eCuttingMode* = 0).

The first cutting length of the material is called top cut. The top cutting length can be defined in the technology module via the parameterisation.

Input *xSyncCrossCutter* = TRUE serves to integrate the material speed and thus to calculate the resulting position of the material which is depicted as position on the X axis (output *lrSetXPosOut*).

The end position on the X axis and thus the cutting position of the material results from the sum of the starting position for the X axis (parameter *lrStartXPosRelative*) and half of the path of the synchronous phase. Therefore the X axis will cover the following distance after the FALSE→TRUE edge at the *xSyncCrossCutter* input until the cut:

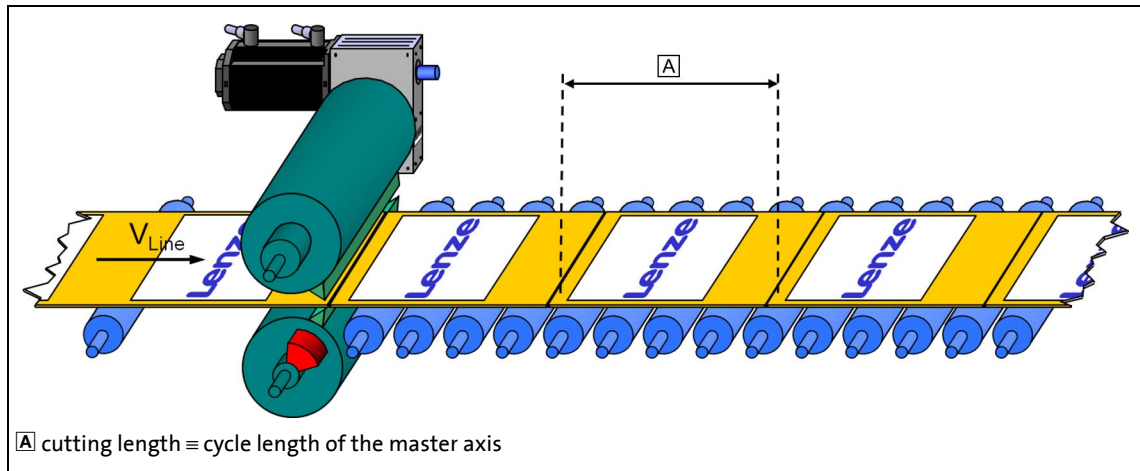
$$lrXDist = lrMasterSyncInDist + \frac{lrCuttingAngle}{2} \cdot \frac{lrCrossCutterCircumference}{360} + lrSetOffsettMaster$$

Parameters to be set

The parameters to be set are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

```
lrStartXPosRelative : LREAL := 0;
lrMasterSyncInDist : LREAL := 0;
lrCrossCutterCircumference : LREAL := 360;
lrCuttingAngle : LREAL := 10;
```

3.12 "Flow packer" operating mode



[3-13] Cutting length in the "Flow packer" operating mode

In the "Flow packer" operating mode, the cycle length of the master axis is decisive for the cut. In this case, the cross cutter is coupled to the master axis via the cycle length. The technology module always effects a cut at the end of the cycle of the master axis, independent of the product length set in the *lrProductLength* input.

The product length is required to calculate the synchronous speed. The synchronous speed of the sealing roller is calculated as a function of the roller circumference in the *lrCrossCutterCircumference* parameter (in units), the speed of the master axis, the cycle length of the master axis and the cycle length of the sealing roller.

The profiles are calculated in real time which provides the possibility to carry out flexible changes to the cutting lengths and the synchronous phase during the cutting process. Any changes are accepted after the synchronous phase and effective for the next cut. The synchronous speed of the sealing roller is calculated as a function of the roller circumference in the *lrCrossCutterCircumference* parameter, the speed of the master axis and the cycle length of the sealing roller.

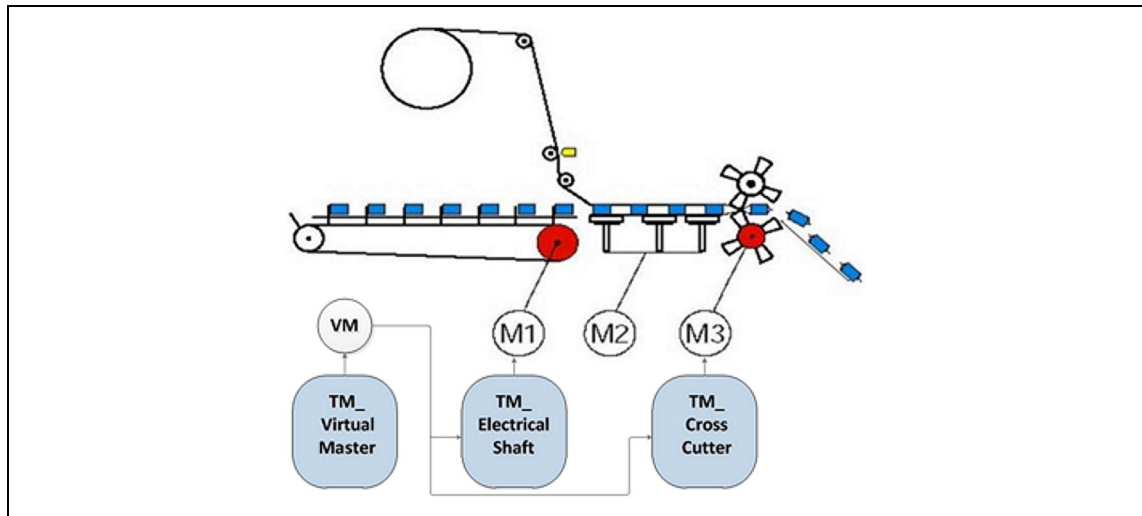
The product/material is cut to the length defined in the *lrProductLength* input. The length is defined in the unit of the master axis (e.g. mm).

The "Flow packer" operating mode is selected via parameter *eCuttingMode* = 1.

Parameters to be set

The parameters for the "flow packer" operating mode are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

```
xMasterAbsolute : BOOL := FALSE;
lrStartXPosRelative : LREAL := 0;
eCuttingMode : L_TT1P_CuttingMode[Base/State] := 1;
lrCrossCutterCircumference : LREAL := 360;
lrCuttingAngle : LREAL := 10;
byNumOfTools : BYTE := 1;
byStartToolNo : BYTE := 0;
```

Example: "Flow packer"

[3-14] Sample structure: "Flow packer"

The "Virtual Master" technology module controls a virtual axis with the modulo cycle 0 ... 360°. The virtual axis is the virtual master for all processing stations in the "Flow packer".

The M1 conveying belt is controlled by the "Electrical Shaft Position" technology module (Base variant). The cycle length of M1 is specified equal to the compartment length of 100 mm in which the product is transported. While the master effects a cycle rotation of 360°, the conveying belt covers a distance of 100 mm, i.e. exactly one compartment length.

The M3 cross cutter is operated by the "Cross Cutter" technology module in the "Flow packer" operating mode. The product length of 100 mm is transferred to the "Cross Cutter" technology module. Thus the technology module also specifies the correct synchronous speed for the cut. The cross cutter makes a cut at the end of the cycle of the virtual master, i.e. at position 360.

The virtual master controls the entire cutting process in this way. With each cycle of the virtual master, the conveying belt covers another 100 mm and the cross cutter makes a cut.

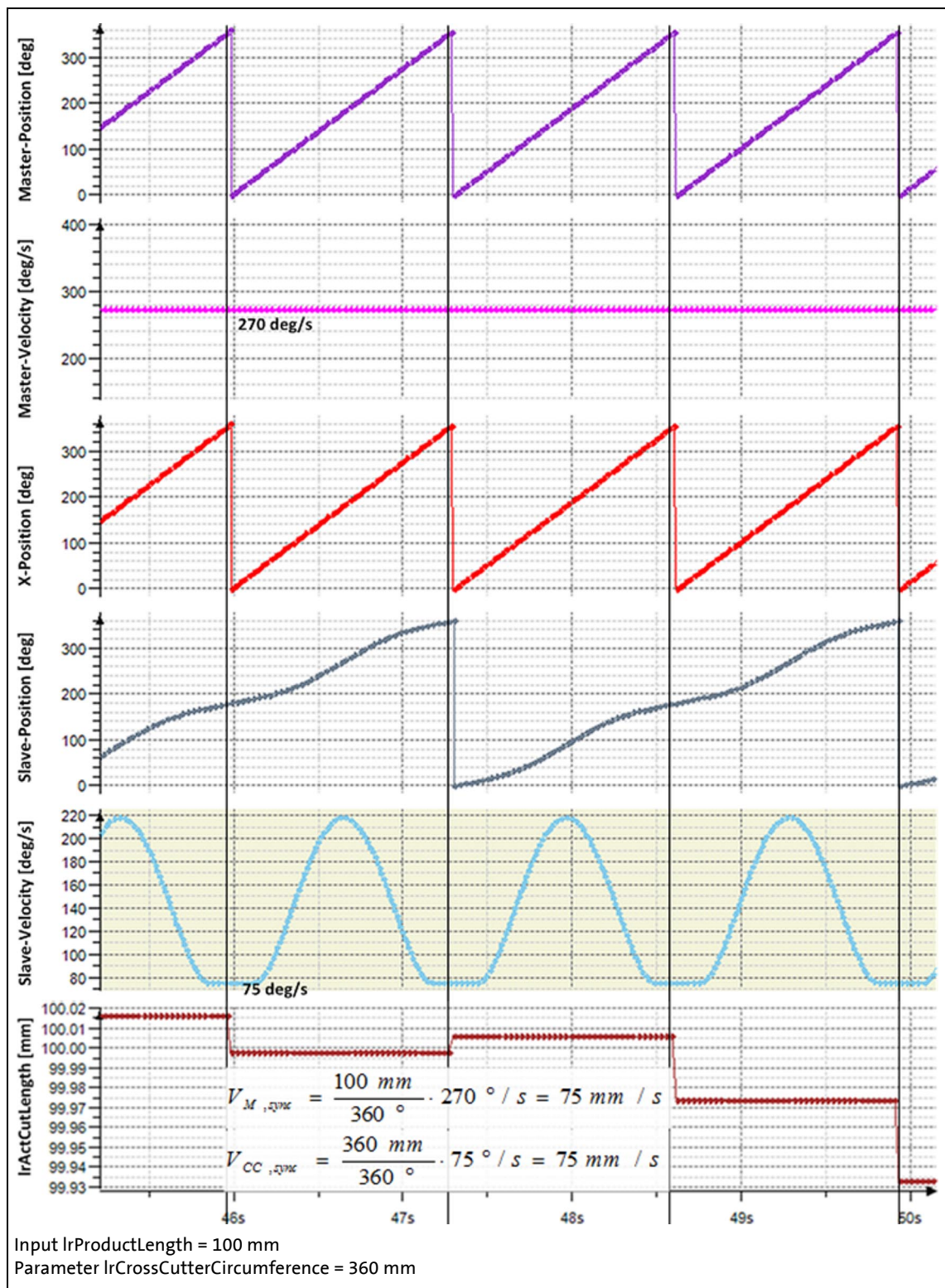
Calculation of the synchronous speed

The synchronous speed (in mm/s) can be derived from the cycle speed of the master axis in the unit [deg/s]:

Calculation of the synchronous speed from the cycle speed of the master axis		
$V_{M, sync} = \frac{L_{Prod}}{L_{MCycle}} \cdot V_{\delta M, sync}$		
Symbol	Description	Dimension unit
$V_{M, sync}$	Cycle speed of the master axis	mm/s
L_{Prod}	Product length	mm
L_{MCycle}	Length of the master cycle	deg
$V_{\delta M, sync}$	Cycle speed of the master axis	deg/s

The synchronous speed can equally be derived from the angular velocity of the cross cutter axis (in deg/s):

Calculation of the synchronous speed from the angular velocity of the cross cutter axis		
$V_{CC, sync} = \frac{Circ_{CC}}{L_{SCycle}} \cdot V_{\delta CC, sync}$		
Symbol	Description	Dimension unit
$V_{CC, sync}$	Angular velocity of the cross cutter axis	mm/s
$Circ_{CC}$	Circumference of the cross cutter axis	mm
L_{SCycle}	Length of the slave cycle	deg
$V_{\delta CC, sync}$	Angular velocity of the cross cutter axis	deg/s



[3-15] History: "Flow packer" cutting process

3.13 Positioning the cross cutter to static material



Note!

During the positioning process the master axis and thus the material must not move.

The "Positioning to static material" clutching type is selected via parameter *eSyncMode* = 1 (positioning).

The cross cutter is synchronised with the material position (X position and resulting Y position from the cam) using the input *xSyncCrossCutter* = TRUE.

There are two possibilities to specify the current material position:

- Specify the material position via the *lrStartXPosRelative* parameter.
For this purpose, the parameter *xMasterAbsolute* must be set to FALSE.
- Set the parameter *xMasterAbsolute* = TRUE.
This serves to set the current material position to the master axis position and thus to define an absolute reference between these two positions to position the slave axis.

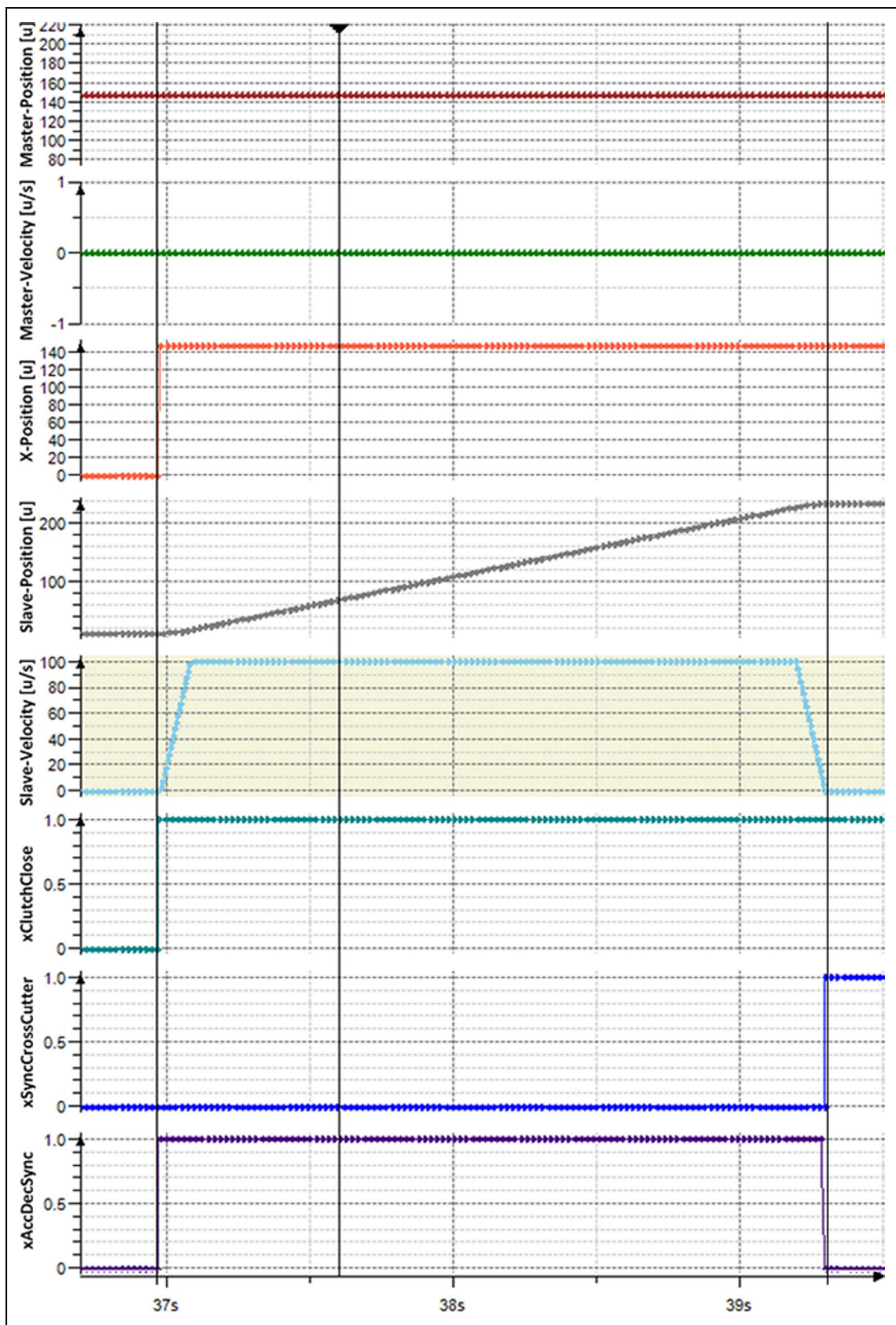
The slave axis is positioned from its current position via the parameters *ePositioningDirection*, *lrPositioningVel*, *lrPositioningAcc* and *lrPositioningDec* to the resulting Y position from the cam.

The positioning process is carried out in the "POSITIONING_TO_START_POS" status. After successful clutch-in, the output *xSynchronised* = TRUE and the "CUTTING" status is set.

Parameters to be set

The parameters for positioning the cross cutter to the static material are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

```
xMasterAbsolute : BOOL := FALSE;
lrStartXPosRelative : LREAL := 0;
eSyncMode : L_TT1P_SyncModeCrossCutter := 0;
ePositioningDirection : MC_DIRECTION := MC_DIRECTION.positive;
lrPositioningVel : LREAL := 10;
lrPositioningAcc : LREAL := 100;
lrPositioningDec : LREAL := 100;
```



[3-16] History: Positioning to the cam

3.14 Clutching the cross cutter to moving material



Note!

For the clutch-in process to moving material, the cross cutter must be in the parking position (set input `xExecuteMoveToParkPos` = TRUE).

► [Positioning to the parking position](#) (□ 39)

The clutching type "Clutching to moving material" is selected via parameter `eSyncMode` = 1 (Cam_in).

The cross cutter is synchronised with the material position (X position and resulting Y position from the cam) using the input `xSyncCrossCutter` = TRUE.

There are two possibilities to specify the current material position:

- Specify the material position via the `lrStartXPosRelative` parameter.
For this purpose, the parameter `xMasterAbsolute` must be set to FALSE.
- Set the parameter `xMasterAbsolute` = TRUE.
This serves to set the current material position to the master axis position and thus to define an absolute reference between these two positions to position the slave axis.

Starting from the parking position, the slave axis is clutched into the synchronous phase using the calculated cam.

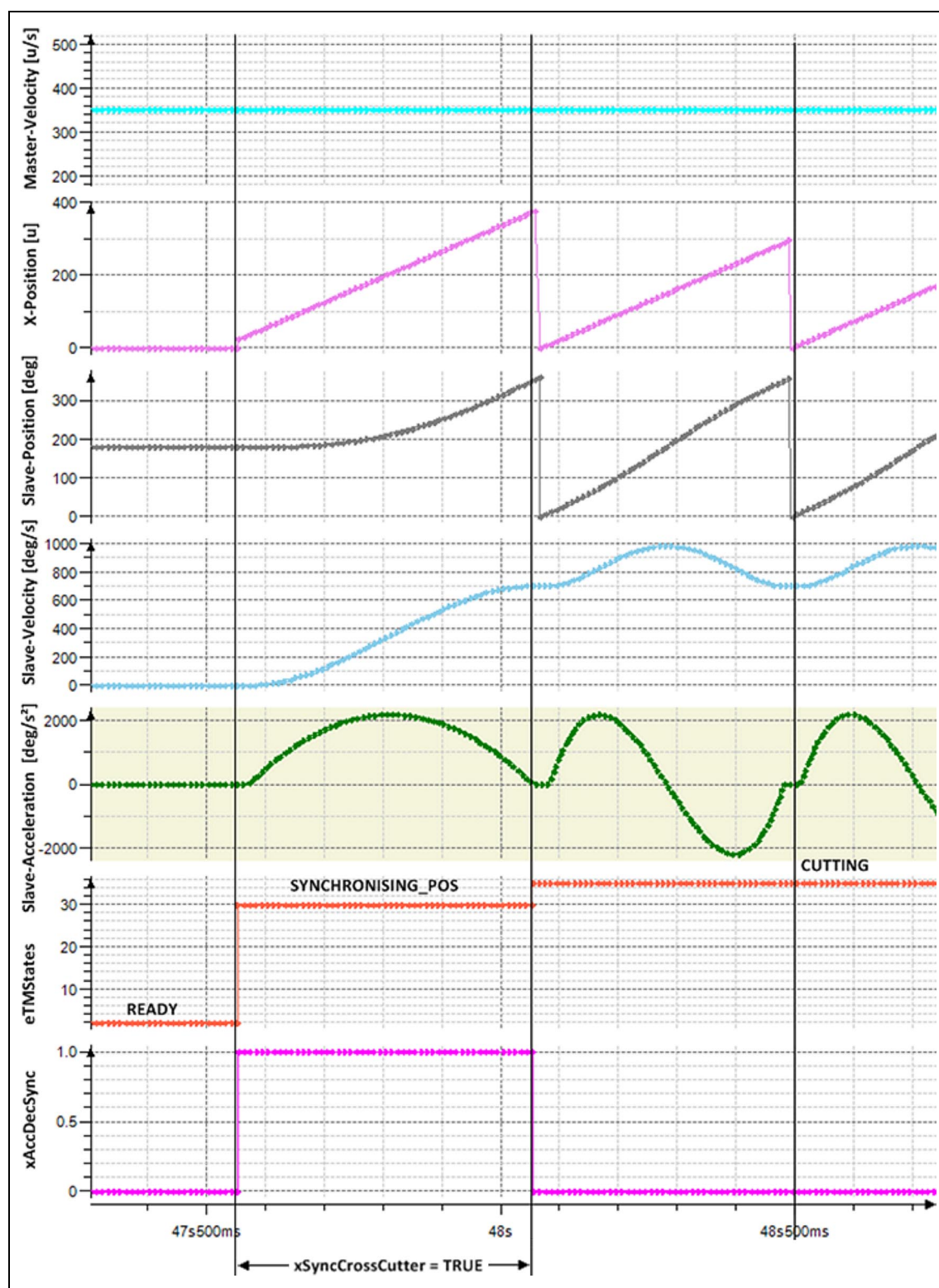
In the "Length cutting" operating mode (parameter `eCuttingMode` = 0) you can set the clutch-in distance of the master axis using the `lrMasterSyncInDist` parameter. Then the first cutting length is defined. Setting `lrMasterSyncInDist` = 0 serves to calculate the optimal clutch-in distance.

The clutch-in process is carried out in the "SYNCHRONISING_POS" status and the `xAccDecSync` output is set to TRUE. After successful clutch-in, the output `xSynchronised` = TRUE and the "CUTTING" status is set.

Parameters to be set

The parameters for clutching the cross cutter to moving material can be found in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

```
xMasterAbsolute : BOOL := FALSE;
lrStartXPosRelative : LREAL := 0;
eSyncMode : L_TT1P_SyncModeCrossCutter := 1;
lrMasterSyncInDist : LREAL := 0;
```



[3-17] History: Clutching to moving material.

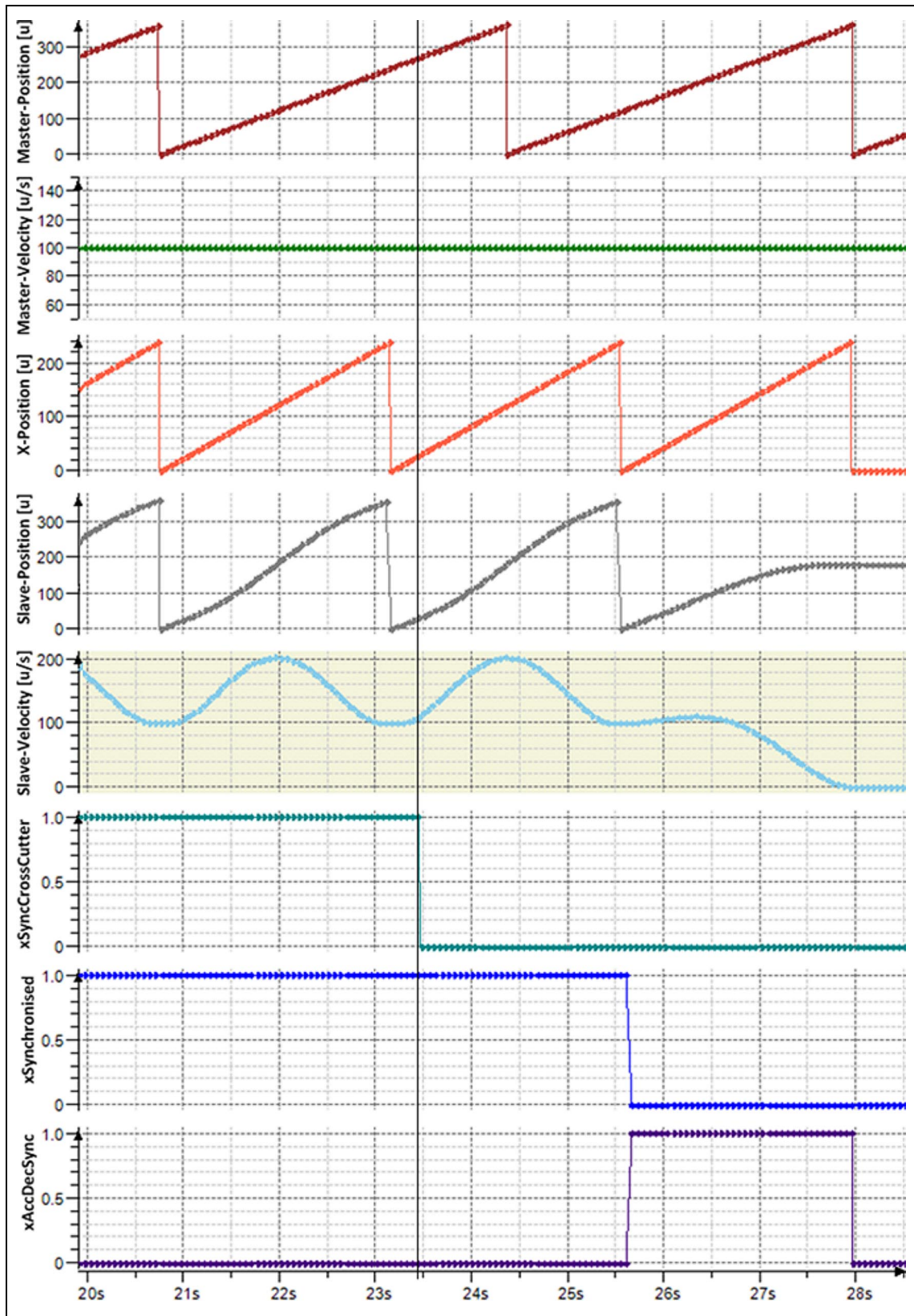
3.15 Declutching the cross cutter

The declutch process can only be carried out if the master axis is moving. It is effected via the input *xSyncCrossCutter* = FALSE.

The declutch process starts after the synchronous phase and can take several cutting cycles. The cross cutter is stopped in the parking position.

Position-controlled declutching from the current cam is carried out by a 5th degree polynomial.

The declutch process is indicated by the output *xAccDecSync* = TRUE. After successfull declutching, the outputs *xDone* = TRUE and *xAccDecSync* = FALSE are set.



[3-18] History: Declutching from the cutting process

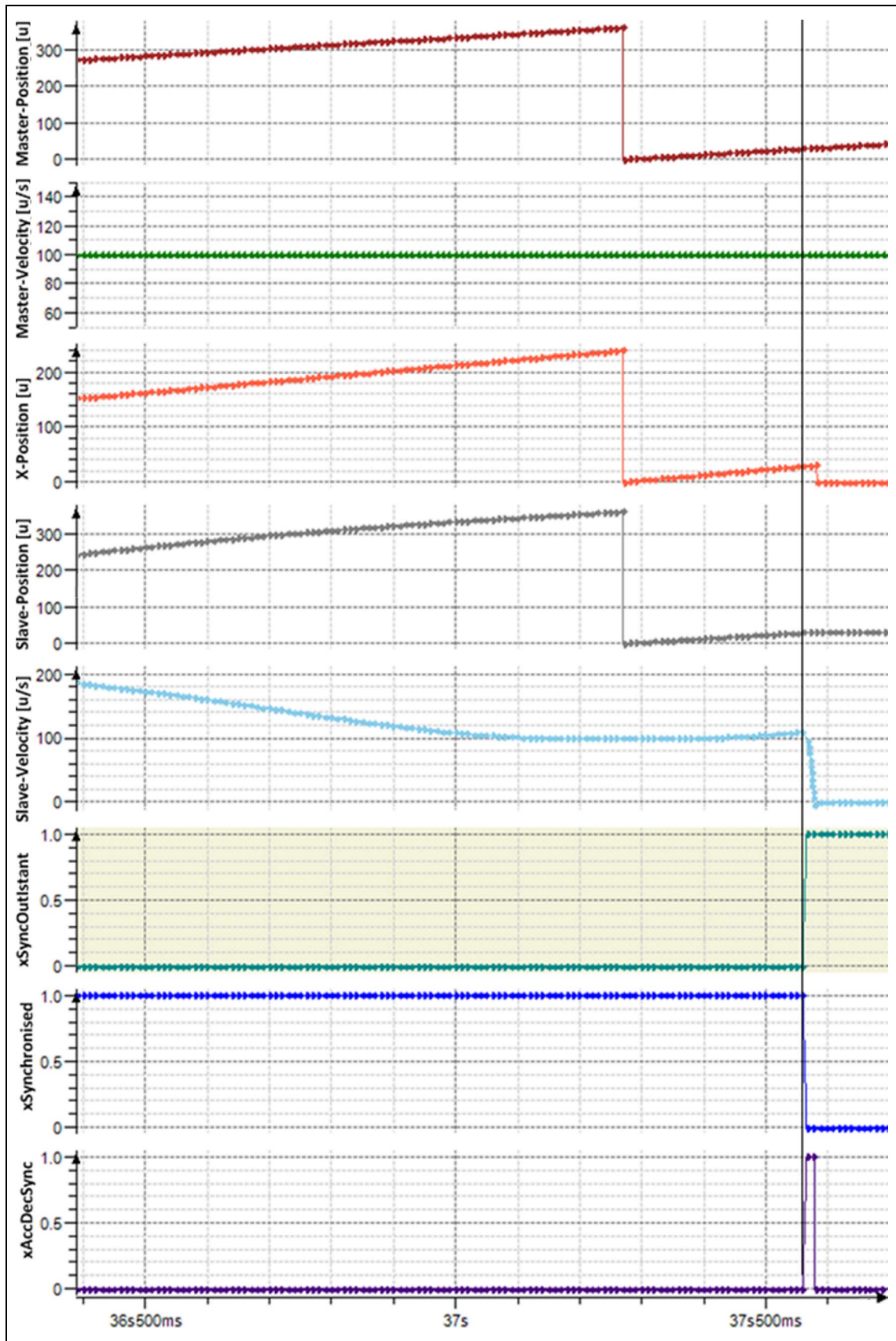
3.16 Positive opening operation / Emergency opening operation

The input *xSyncOutInstant* = TRUE serves to immediately declutch and stop the slave axis via the deceleration specified in the *lrSyncOutInstantDec* parameter at the current curve position.

The clutch remains opened as long as the input *xSyncOutInstant* is set to TRUE. The *xSyncOutInstant* input has higher priority than the *xSyncCrossCutter* input.

The declutch process is indicated by the output *xCuttingActive* = TRUE. After successful declutching, the outputs *xDone* = TRUE and *xCuttingActive* = FALSE are set.

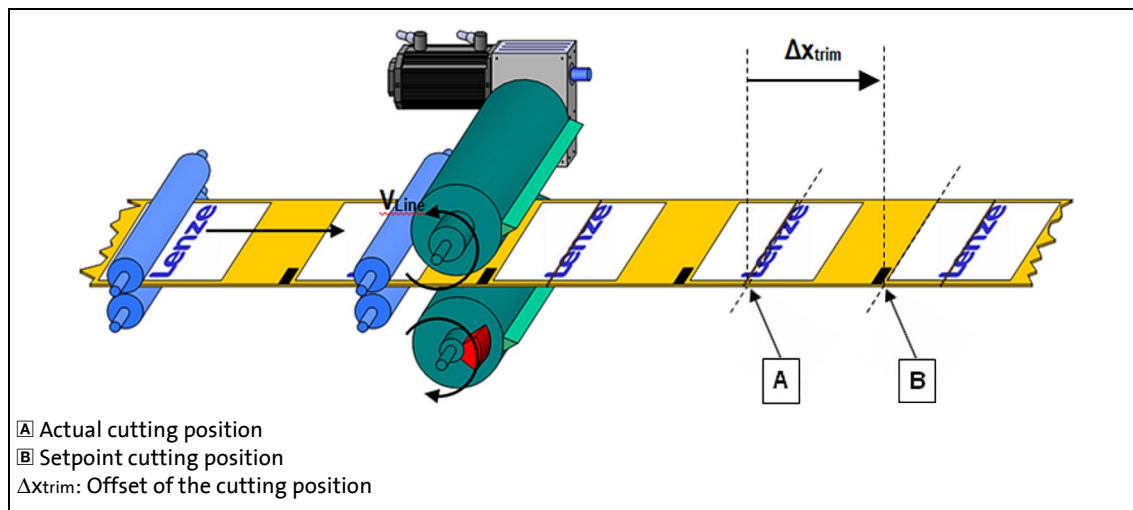
The *lrSyncOutInstantDec* parameter can be found in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.



[3-19] History: Positive opening operation / Emergency opening operation

3.17 Trimming and offset of the cutting position

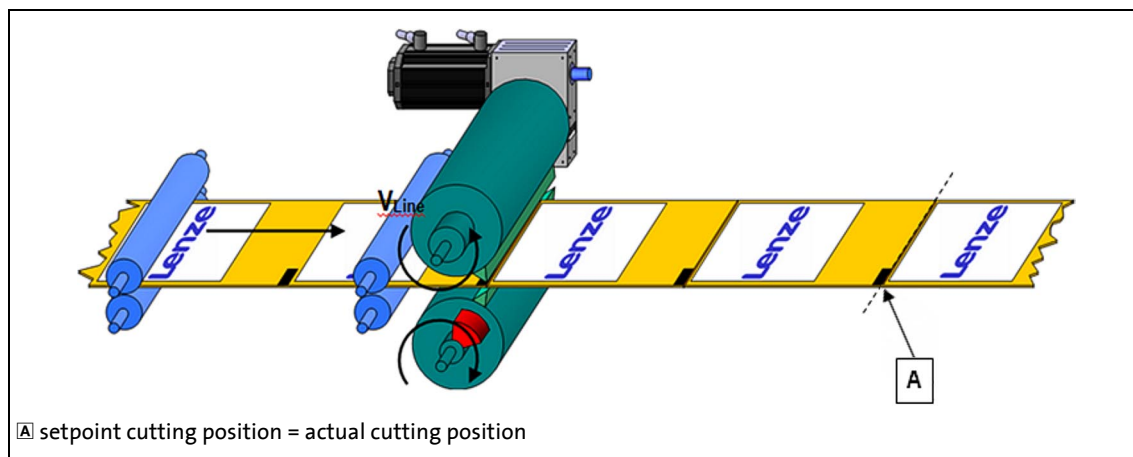
The trimming and offset functions provide a possibility to shift the cut in relation to the product cycle.



[3-20] Cutting position: the cut is in the middle of the print image

In fig. [3-20] the cut is in the middle of the print image. In order to get the cut into the required position, the cross cutter drive must be trimmed by the corresponding differential travel Δx_{trim} with respect to the current cutting position.

After completing the trimming, the cut - with reference to the product cycle - is shifted and in the ideal case it is placed correctly (fig. [3-21]: setpoint cutting position = actual cutting position).



[3-21] Optimal cutting position: the cut is exactly at the cycle transition

The technology module provides the following functions for cut trimming and cut offset:

- ▶ [Position cut trimming](#) (□ 58)
- ▶ [Speed cut trimming \(Tip trimming\)](#) (□ 60)
- ▶ [Absolute positioning of the cutting position via master offset](#) (□ 62)

3.17.1 Position cut trimming

The position cut trimming is enabled when *eTrimMode* parameter = 0.

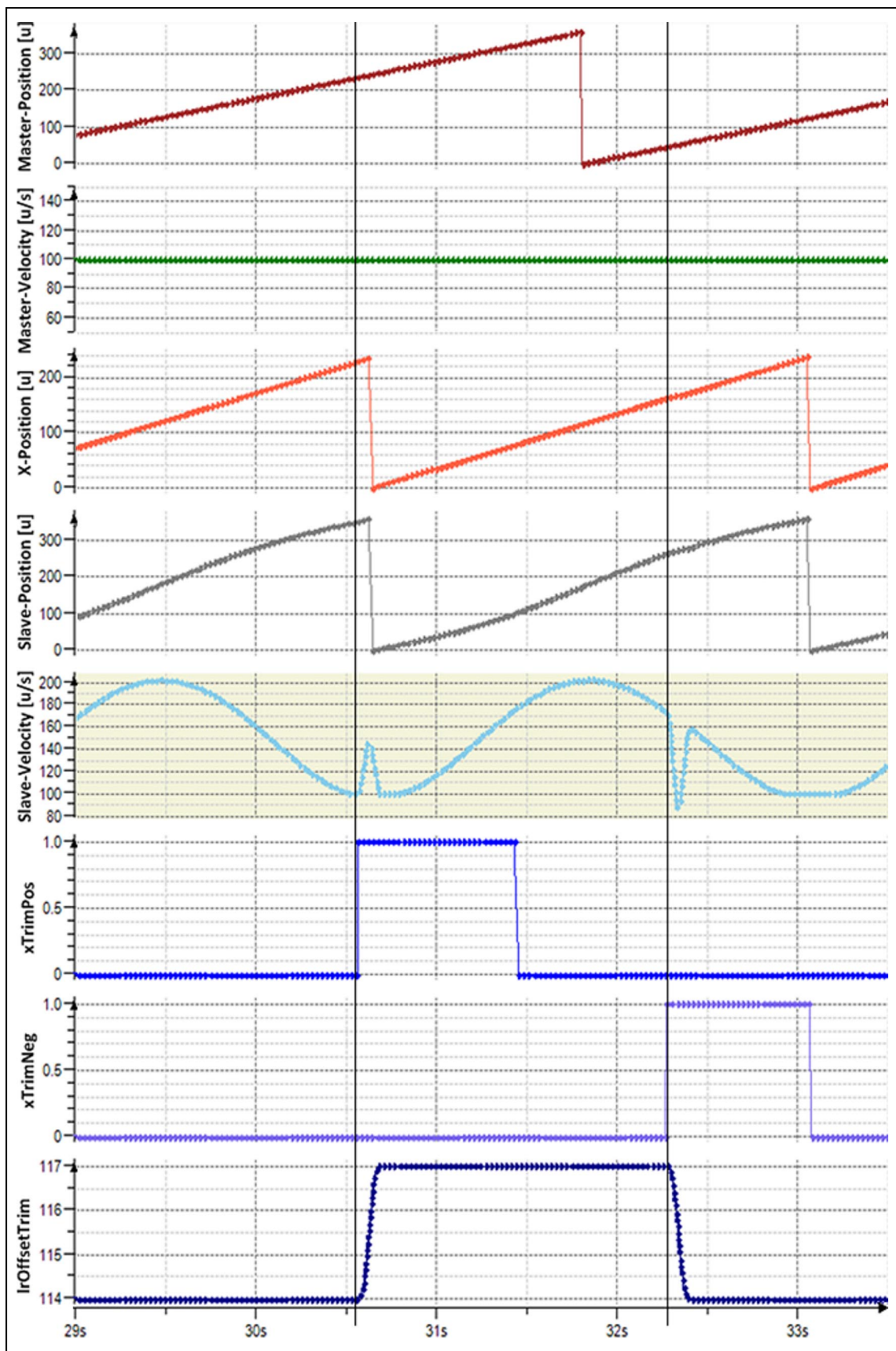
In the position cut trimming, the cut is trimmed via a defined increment.

The cutting position is shifted using the inputs *xTrimPos* = TRUE (in positive direction) or *xTrimNeg* = TRUE (in negative direction) by the increment specified in the *lrTrimDist* parameter.

Parameters to be set

The parameters for position cut trimming are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) ([23](#)) parameter structure.

```
eTrimMode : L_TT1A_TrimMode := 0;  
lrTrimDist : LREAL := 3;  
lrOffsetTrimVel : LREAL := 10;  
lrOffsetTrimAcc : LREAL := 100;  
lrOffsetTrimDec : LREAL := 100;
```



[3-22] Progression: Position cut trimming

3.17.2 Speed cut trimming (Tip trimming)

The speed cut trimming is enabled when *eTrimMode* parameter = 1.

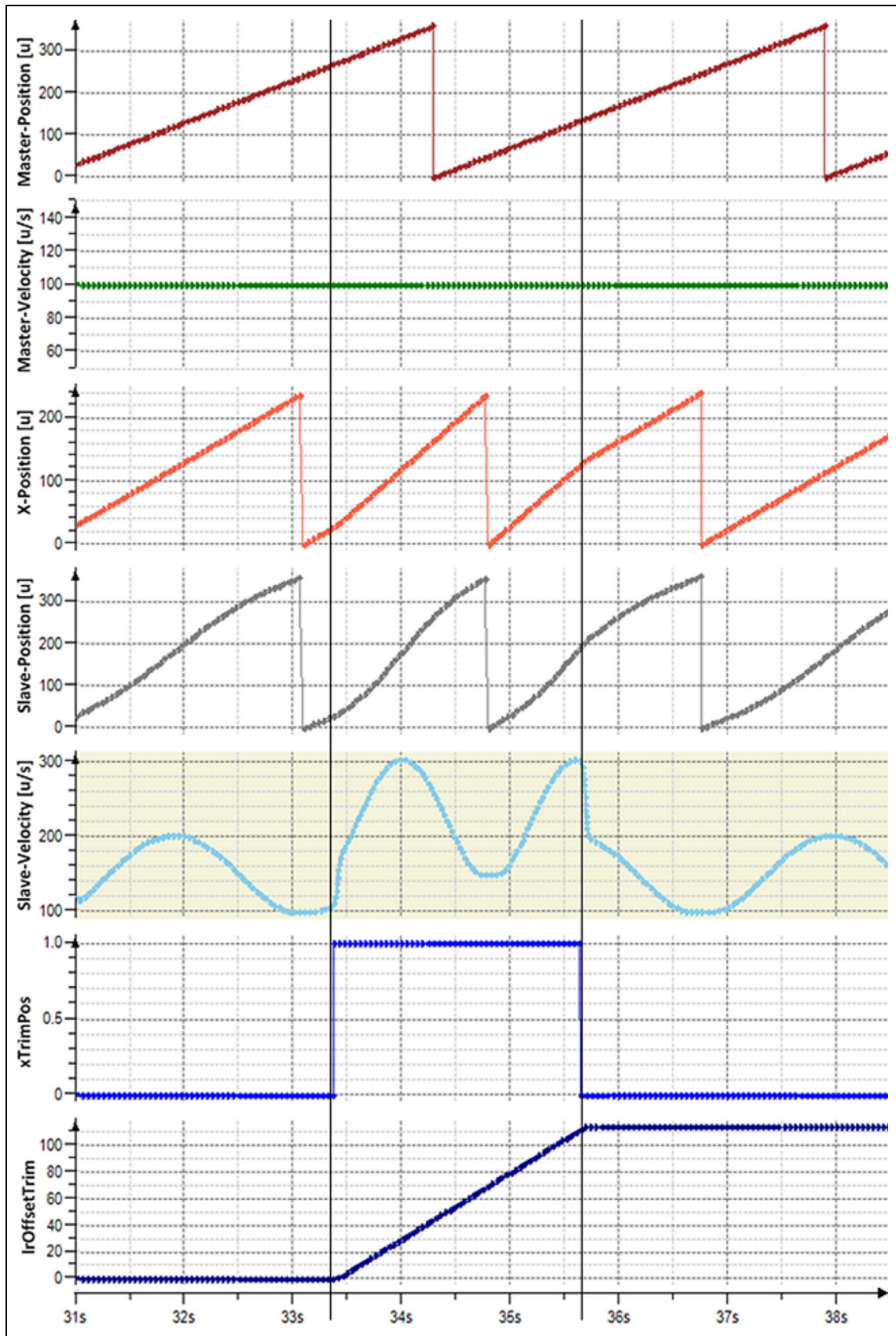
With speed cut trimming, the cutting position is shifted using the *xTrimPos* (in positive direction) or *xTrimNeg* (in negative direction) inputs. The offset per time unit is specified by the trimming velocity (*lrOffsetTrimVel* parameter).

Similar to [Manual jog \(jogging\)](#) (41) the setpoint cutting position is shifted back or forth at constant speed until one of the inputs *xTrimPos* or *xTrimNeg* is TRUE.

Parameters to be set

The parameters for speed cut trimming are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

```
eTrimMode : L_TT1A_TrimMode := 1;  
lrOffsetTrimVel : LREAL := 10;  
lrOffsetTrimAcc : LREAL := 100;  
lrOffsetTrimDec : LREAL := 100;
```



[3-23] Progression: Speed cut trimming

3.17.3 Absolute positioning of the cutting position via master offset

In this mode, the cutting position is shifted by an offset already known before the start of the cut trimming. The offset must be specified as master offset via the *lrSetOffsetMaster* input. The offset is applied by a profile generator and is independent of the phase (synchronous or asynchronous).

The absolute offset from the sum of the trimming and the master offset are displayed via the *lrOffset* output.

In order to restore the initial reference, the offset must be set to '0'. The application of the offset is started automatically as soon as a new value different from the previous setpoint is set.

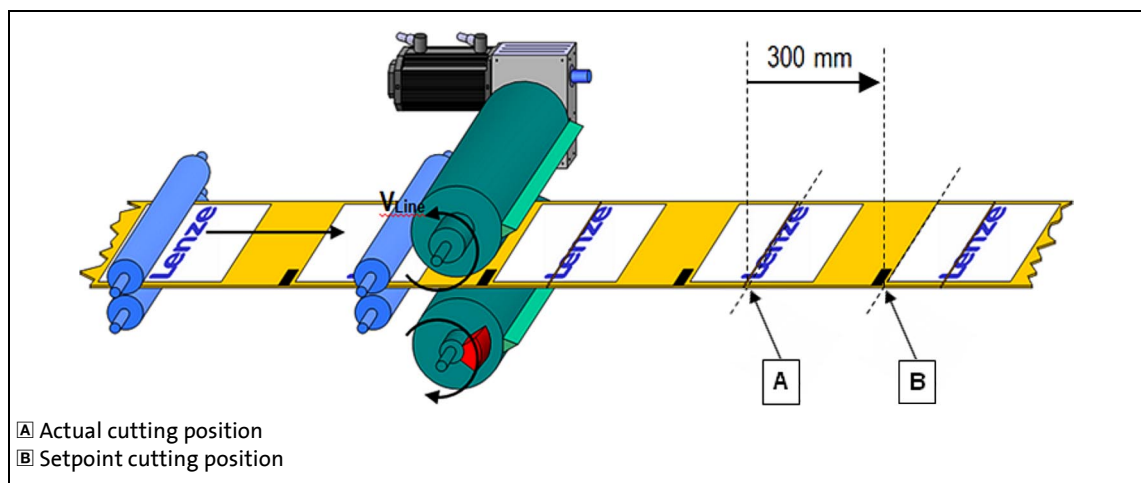
Parameters to be set

The parameters for the absolute positioning of the cutting position are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

```
eTrimMode : L_TT1A_TrimMode := 0;
lrOffsetTrimVel : LREAL := 10;
lrOffsetTrimAcc : LREAL := 100;
lrOffsetTrimDec : LREAL := 100;
```

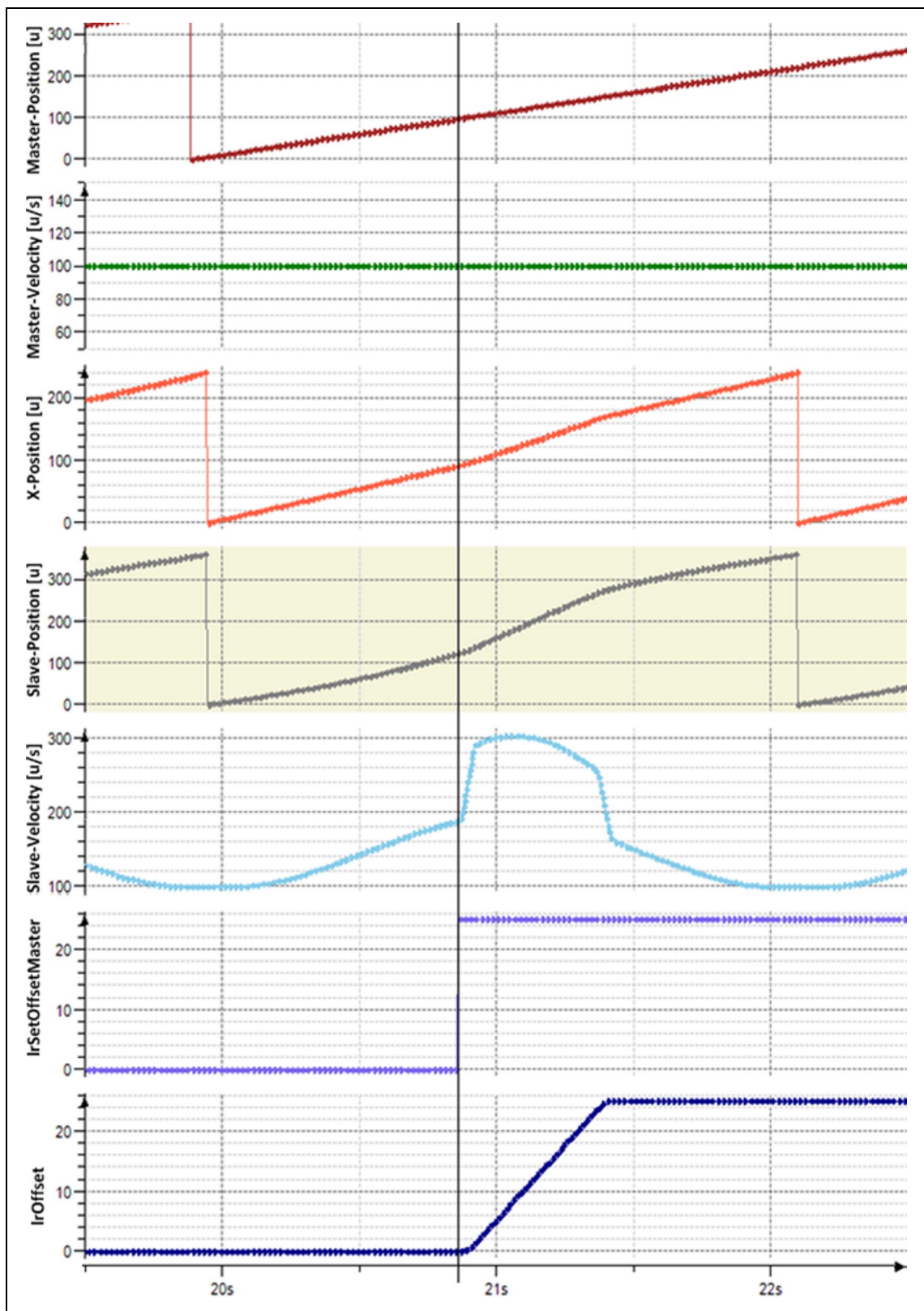
Example

In a current actual cutting position (X position), the cut is 300 mm before the setpoint cutting position.



[3-24] Compensation of a known cutting offset

By means of the cut trimming, the cut is shifted by 300 mm in material flow direction to the required position (offset in positive direction). For this purpose, the required offset of +300 mm is specified for the current actual cutting position. The offset is automatically applied immediately.



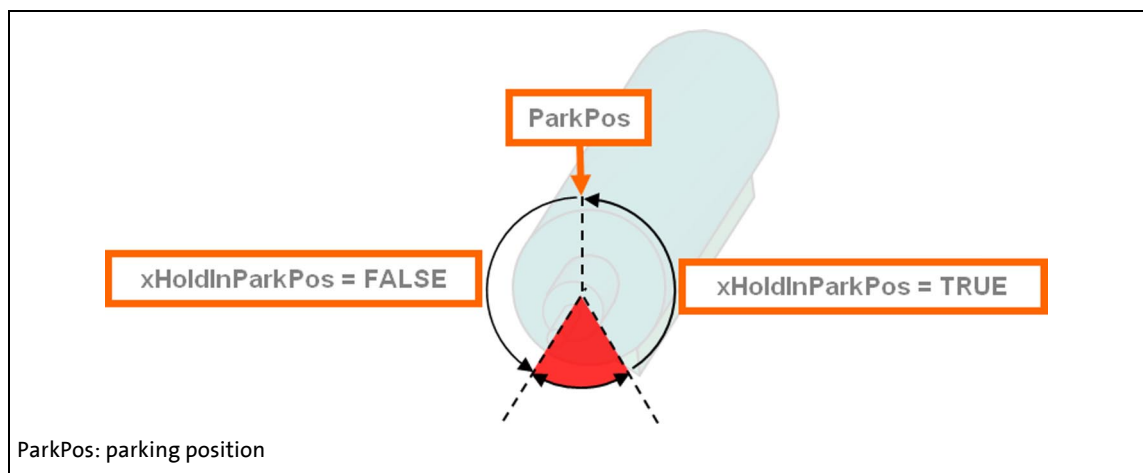
[3-25] History: Acceptance of the offset by the profile generator

3.18 Method of elimination

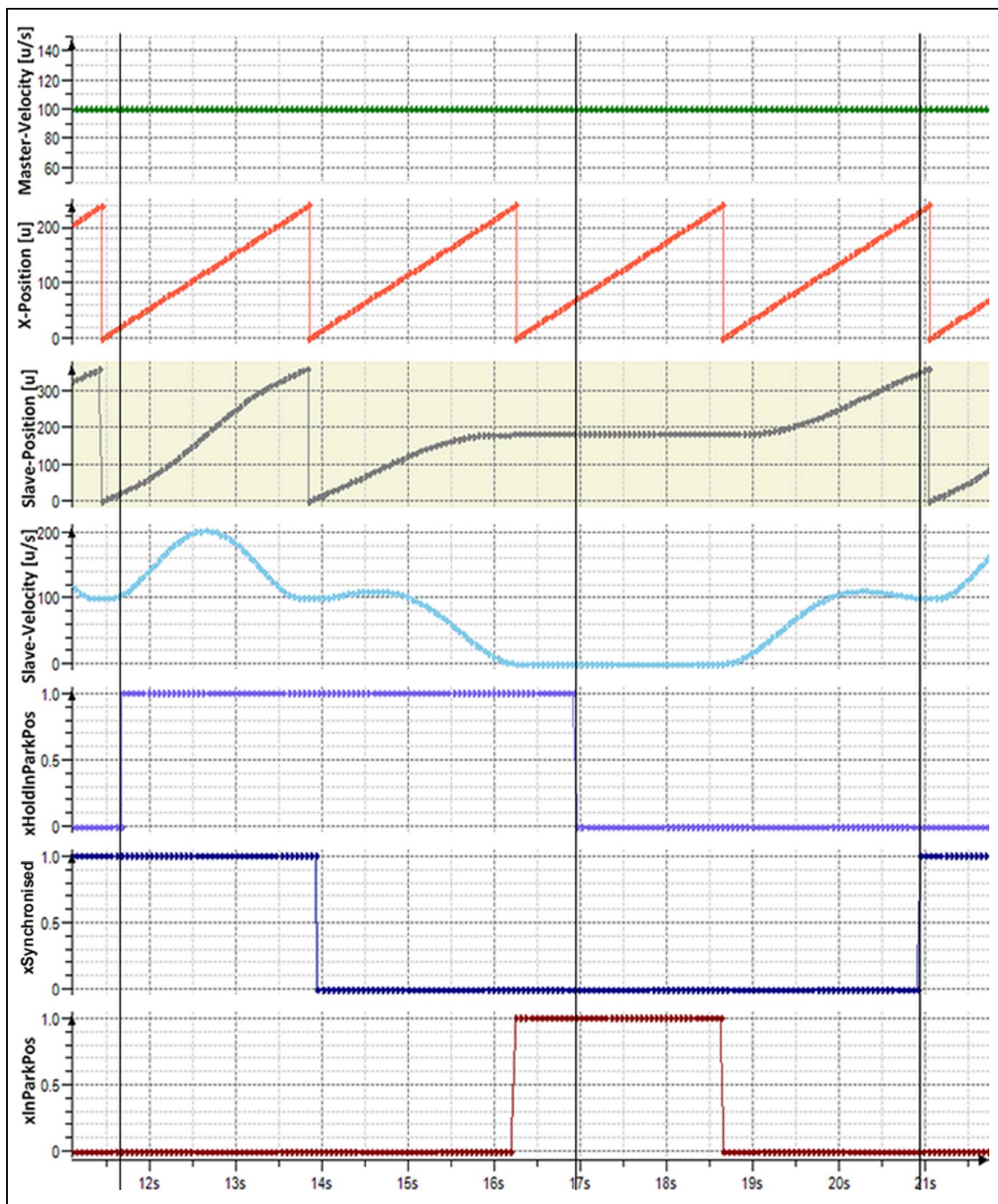
The elimination function serves to leave out cuts in the material. For this purpose the cross cutter is brought to the parking position in order not to carry out the cutting process.

The input *xHoldInParkPos* = TRUE serves to declutch the cross cutter after the synchronous phase into the parking position. As soon as the parking position is reached, the output *xInParkPos* is set to TRUE. The position of the material is still determined. The cross cutter remains in the parking position until the input *xHoldInParkPos* is set to FALSE.

The cutting process is resumed after the synchronous phase, i.e. the cross cutter is clutched into the next synchronous phase.



[3-26] Parking position in the elimination function

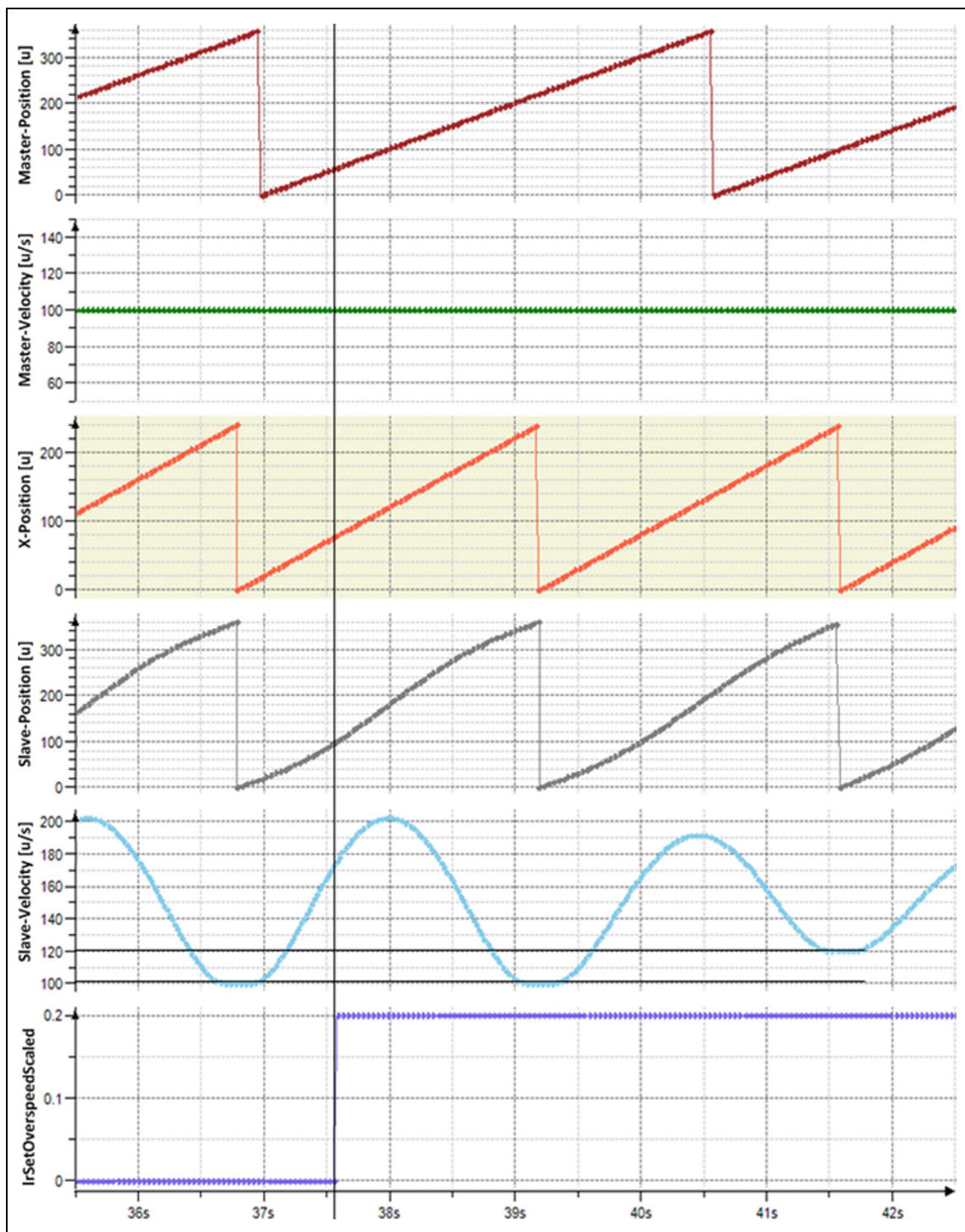


[3-27] History: Stopping the cross cutter in the cutting process

3.19 Increasing/decreasing the speed (overspeed)

In certain applications, the motion in the cutting phase must be undersynchronous or oversynchronous with the material speed by an adjustable factor. Increasing or decreasing the speed is made possible by the *IrSetOverspeedScaled* input. A scaled value with reference to the current material speed is expected here.

If, for example, the synchronous speed of the sealing roller is to be decreased by 20%, the input *IrSetOverspeedScaled* must be set to -0.2. The setting '0.5' results in an increase in the synchronous speed by 50%. The valid value range of the *IrSetOverspeedScaled* input is from -0.9 to 2.0. Any change to the value is accepted after the synchronous phase and is effective for the next step.



[3-28] History: Oversynchronous cutting process

3.20 Calculating extreme values of a cam

The technology module provides the possibility to examine the profiles for the slave axis for maxima of speed and acceleration. The maxima are calculated using the parameter *xCalcCamBounds* = TRUE.

The calculation requires the maximum speed (*lrMasterMaxVel* parameter) and the maximum acceleration (*lrMasterMaxAcc* parameter) of the master axis.

The outputs *lrSlaveMaxVel* and *lrSlaveMaxAcc* serve to put out the maximum velocity and the maximum acceleration of the slave axis. These values are updated automatically.

Parameters to be set

The parameters to be set are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

```
xCalcCamBounds : BOOL := TRUE;  
lrMasterMaxVel  : LREAL := 100;  
lrMasterMaxAcc  : LREAL := 1000;
```

3.21 Calculating the cutting length

After each cut, the technology module calculates the material length cut (output *xCutDone* = TRUE).

The cut length results from the difference between the current position of the master axis and the current cutting position of the sealing roller.

The current cutting length is put out at the *lrActCutLength* output.

3.22 "Length cutting with mark correction" operating mode

In the "Length cutting with mark correction" operating mode, the integrated register controller as higher-level control loop controls the position relative to a mark detected on the material (touch probe). For this purpose the technology module creates a register axis which contains the corrected position. This register position is used as master value for the cutting formats.

The "Length cutting with mark correction" operating mode is selected using parameter *eCuttingMode* = 2.



Note!

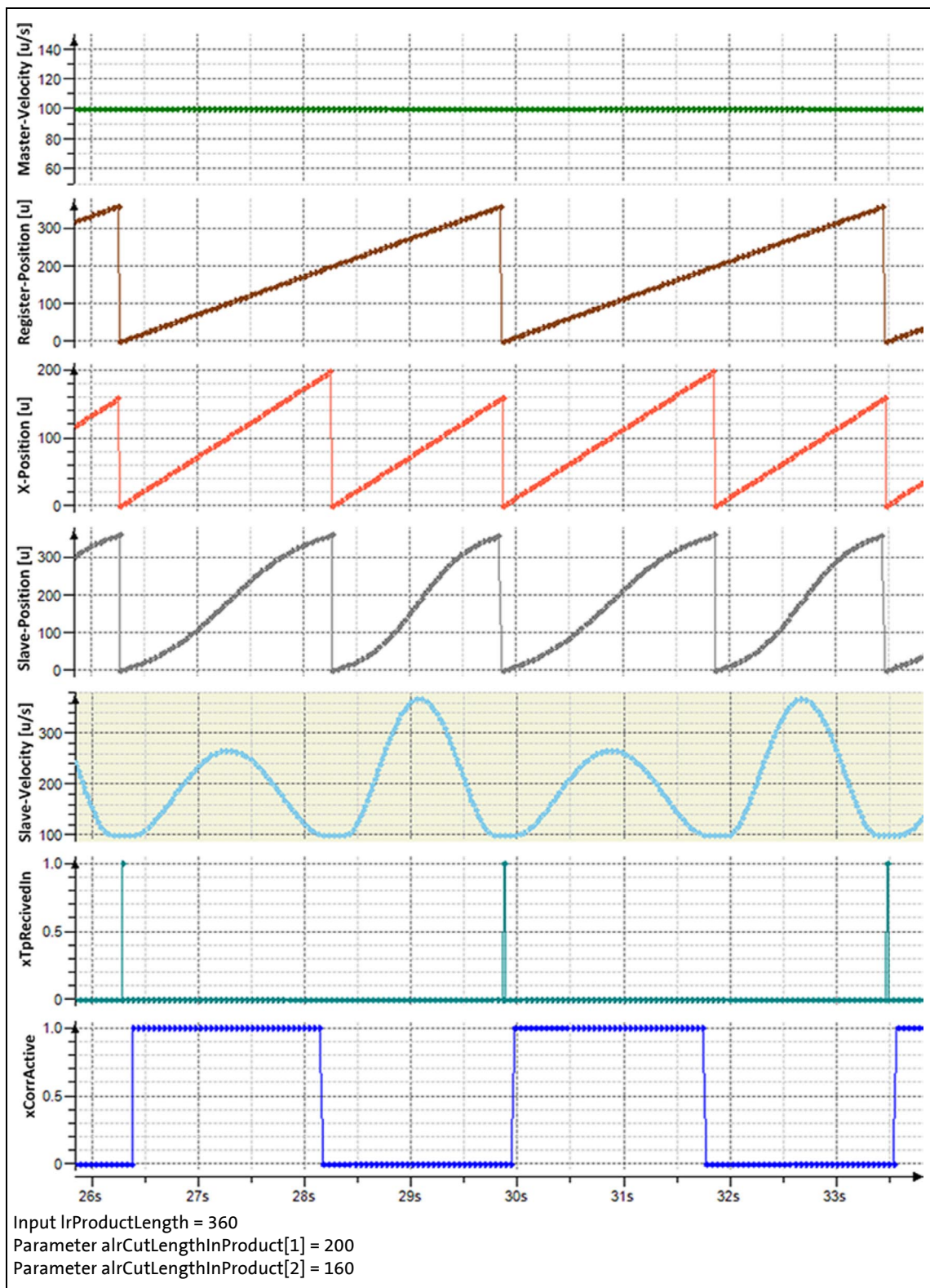
Before the cutting process ...

- you must select a touch probe source.
‣ [Selecting the touch probe source](#) (□ 74)
- you must reference the mark position (touch probe) for register control;
‣ [Referencing the mark position](#) (□ 75)

Parameters to be set

The parameters for the "length cutting with mark correction" operating mode are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

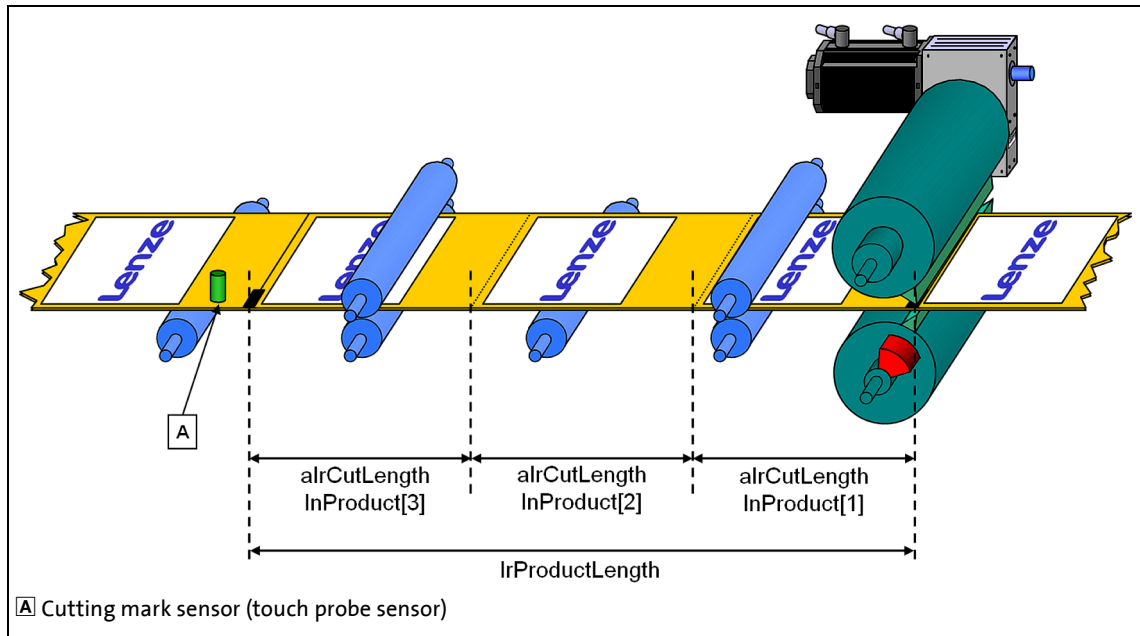
```
xMasterAbsolute : BOOL := FALSE;
lrStartXPosRelative : LREAL := 0;
eCuttingMode : L_TT1P_CuttingModeState := 2;
eTpMode : L_TT1P_TpMode := 0;
lrSensorToolDist : LREAL := 0;
lrCrossCutterCircumference : LREAL := 360;
lrCuttingAngle : LREAL := 10;
byNumOfTools : BYTE := 1;
byStartToolNo : BYTE := 0;
xMarkCorrection : BOOL := TRUE;
lrMarkWindowSize : LREAL := 90;
lrMaxCorrPos : LREAL := 30.0;
lrMaxCorrNeg : LREAL := -30.0;
```



[3-29] History: Cutting process with mark correction via register control

3.23

Partial cutting lengths in product length (print/cut format)



[3-30] Cutting lengths in product length (print/cut format)

The product length is defined by the *lrProductLength* input. In this way you also define the cycle length of the register. This value is always constant for the cutting process and cannot be changed. Cuts are always expected at the interval of the product length.

The parameter (array) *alrCutLengthInProduct[1..10]* serves to define up to 10 partial cuts with different cutting lengths within the product length (or the register cycle). The first cut *alrCutLengthInProduct[1]* is made at the beginning of the product, the second cut *alrCutLengthInProduct[2]* subsequently, etc. The sum of all cutting lengths in the array must correspond to the product length (the register cycle).

All partial cutting lengths defined with the value '0' are deactivated. In the event that all partial cutting lengths are defined with the value '0', the cutting length is equal to the product length (input *lrProductLength*).

Parameters to be set

The parameters to be set are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

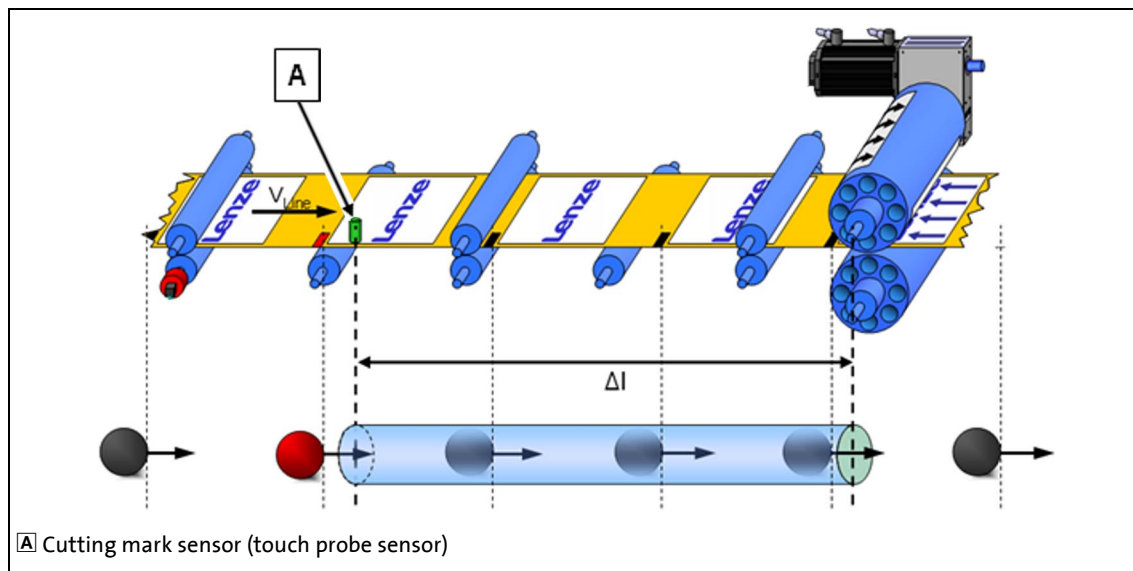
```
eCuttingMode : L_TT1P_CuttingMode[Base/State] := 2;
eTpMode : L_TT1P_TpMode := 0;
lrMarkWindowSize : LREAL := 90;
```

3.24

Cutting mark register

The cutting mark register allows for the installation of the cutting mark sensor further away from the axis with the tool (e.g. cutting blade) than one register cycle.

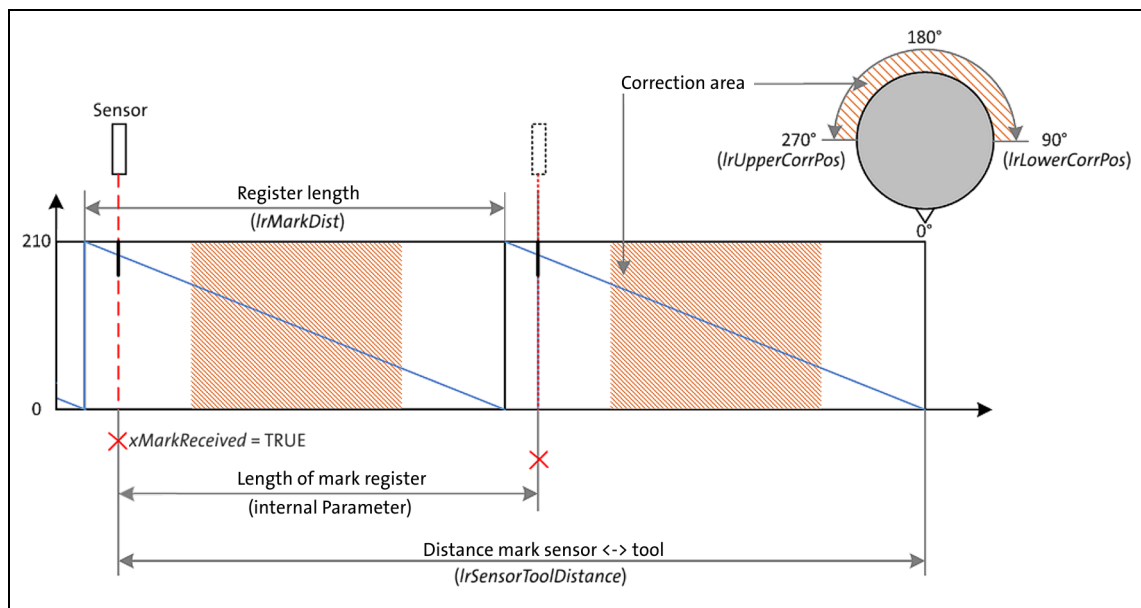
Fig. [3-31] shows the use of a cutting mark register. Here the distance (Δl) from the cutting mark sensor to the tool is larger than the specified register cycle.



[3-31] Systematic representation of the cutting mark register

The aim should always be to mount the cutting mark sensor as close to the axis as possible. The further away from the axis the sensor is mounted, the more changes to the material flow will remain undetected and result in cutting inaccuracies.

Up to 100 mark signals can be managed in the mark register. They are available for the system at the right time – delayed by the mark register length. In this way, a cut can always be made on the right cutting mark signal, for instance.



[3-32] Cutting mark register with correction areas

The distance of the touch probe sensor to the attack position of the tool on the material is defined via the $lrSensorToolDist$ parameter.

After a mark has been detected, the value of the mark deviation will only be enabled after the position in the register cycle has covered the distance of the mark register.

3.25 Selecting the touch probe source

The touch probe source for detecting the mark on the material is selected via the *eTPMode* parameter.

If an external encoder is used (*eTPMode* = 2), the *lrCycPosExtEncoder* input (cyclic position of the encoder) has to be interconnected and the *lrCycleLengthExtEncoder* parameter (cycle length of the encoder) has to be set. This information is required to convert the axis-side touch probe event to the internal register format.

Parameters to be set

The parameters to be set are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

```
eTpMode : L_TT1P_TpMode := 0;  
lrCycleLengthExtEncoder : LREAL := 0;
```

3.26 Referencing the mark position

The mark position must be referenced before the cutting process can be started in the "Length cutting with mark correction" operating mode (parameter *eCuttingMode* = 2).

Preconditions for referencing the mark position (touch probe) are the selection of the operating mode (parameter *eCuttingMode*), the determination of the product length (input *lrProductLength*) and no active cutting process. The product length corresponds to the length of one register cycle of the register axis.

During homing, the axis is moved from the touch probe source (parameter *eTpMode*) and a valid touch probe mark is transferred via inputs *xMarkReceive* and *lrActMarkPosIn*. The next valid touch probe mark serves to set the register position. The position where the mark is expected is determined internally and is indicated via output *lrMarkSensorPos*. A touch probe window with the width specified in parameter *lrMarkWindowSize* is symmetrically ($\pm lrMarkWindowSize / 2$) put around the current touch probe mark.

The register axis is coupled with the master axis using a synchronism. The output *xMarkWindowTeached* is set to TRUE if the home position of the mark is known. The product length must then not be changed any more. If you want to change the product length and thus the register register cycle, you must re-reference the mark.

Referencing the mark position is started using input *xTeachMarkWindow* = TRUE. During the referencing, the output *xTeachMarkWindowActive* is set to TRUE.

The *lrActMarkDist* output contains the register length between the last two touch probe marks.

Parameters to be set

The parameters to be set are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

```
eCuttingMode : L_TT1P_CuttingModeState := 2;
eTpMode      : L_TT1P_TpMode          := 0;
lrMarkWindowSize : LREAL              := 90;
```

3.27 Speed synchronism of master and register axis

For the synchronism of the register axis with the master axis, a register position is created within the technology module which is used by the cross cutter (slave axis) as master position.

The register position is created by integrating the setpoint speed of the master axis into the register cycle (input *IrProductLength*). The calculated register position is put out at the *IrSetRegPosOut* output.

By coupling the input *xSyncCrossCutter* = TRUE, the slave axis is connected with the register axis without jerking.

3.28 Gearbox factor correction

Changing register features (e.g. within a paper reel) will result in a changed real register length. The difference to the parameterised register length results in turn in corrections in always the same direction (positive/negative). This is inefficient and contributes to higher energy consumption as well as increased mechanic load.

For this case, the gearbox factor correction calculates the optimal speed setpoint of the master axis. This optimisation causes the corrections to be carried out equally in the positive and negative direction.

The correction value is calculated from the mean value of the touch probe deviations. It multiplicatively affects the setpoint speed of the master axis via an additional correction gearbox factor.

The gearbox factor correction is activated by input *xGearFactorCorr* = TRUE and is active until the input *xSyncCrossCutter* is set to TRUE.

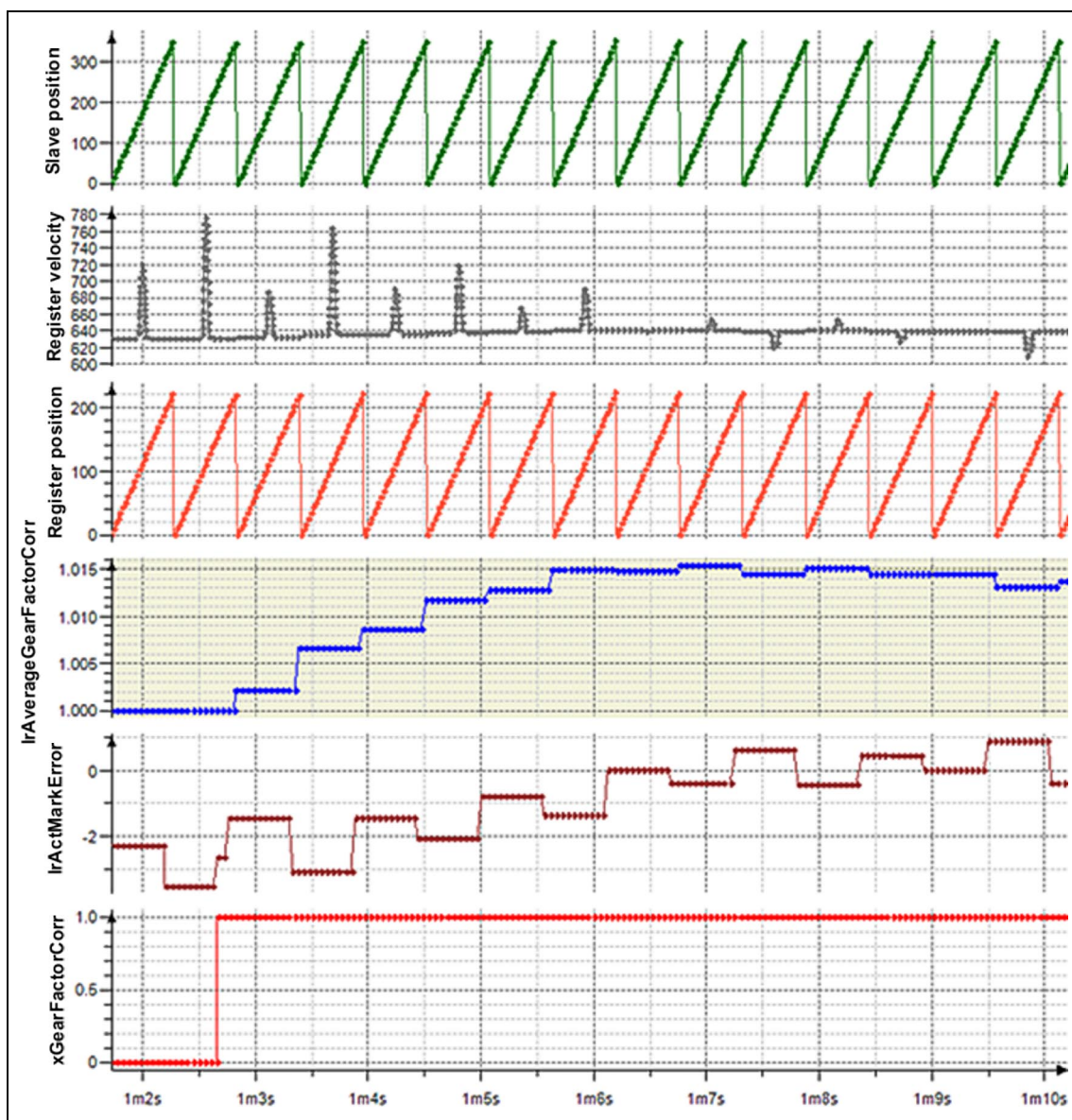
The gain of the gearbox factor correction is set via the gain factor in the *lrGearFactorCorrGain* parameter. When the gearbox factor correction is activated, the current value of the corrected gearbox factor is provided via the *lrAverageGearFactorCorr* output.

The maximum value of the gearbox factor correction is defined using the *lrMaxGearFactorCorr* parameter. If the gearbox factor correction operates at the positive or negative limit, this will be displayed via the *xGearFactorCorrLimited* output.

Parameters to be set

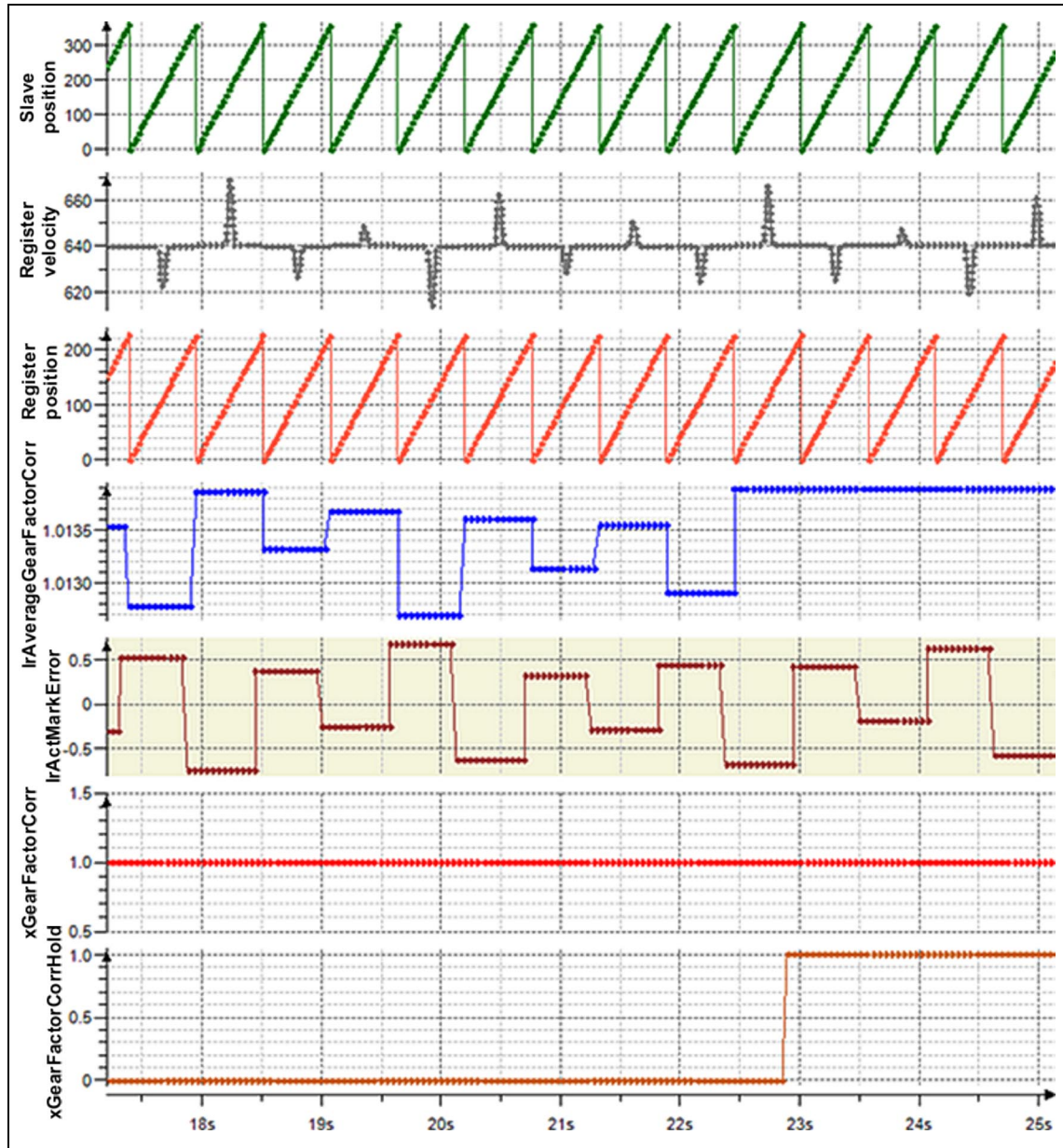
The parameters to be set for the gearbox factor correction are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

```
lrGearFactorCorrGain : LREAL := 0.1;
lrMaxGearFactorCorr  : LREAL := 10;    // [units]
```



[3-33] Signal characteristic with active gearbox factor correction

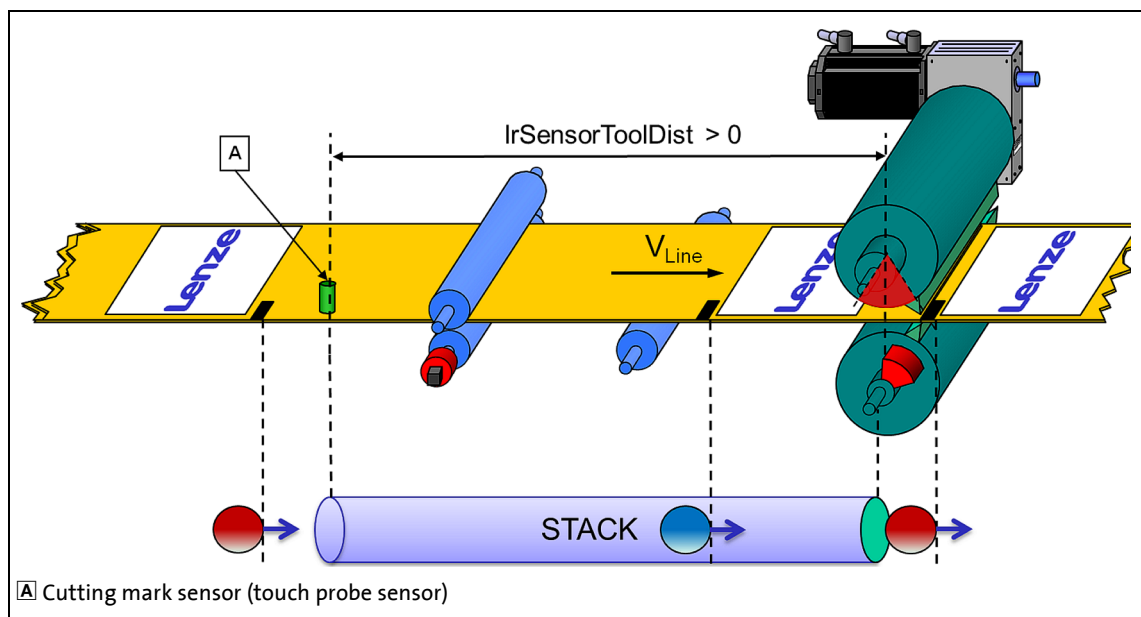
As soon as the optimum corrected gearbox factor has been detected, correction movements occur both in positive and negative direction. The detected gearbox factor value can be permanently accepted when the `xGearFactorCorrHold` input = TRUE. The value will only be changed when `xGearFactorCorrHold` is set to FALSE.



[3-34] Hold signal characteristic with optimised gearbox factor

3.29

"Cutting on a mark (event-controlled)" operating mode



[3-35] "Cutting on a mark (event-controlled)" operating mode

The "Cutting on a mark (event-controlled)" operating mode is selected using parameter *eCuttingMode* = 3. Using this operating mode is only possible, if the cutting mark sensor is installed in front of the cross cutter. The distance between cross cutter and cutting mark sensor is set using parameter *IrSensorToolDist* > 0.

The profiles are calculated in real time in this operating mode which provides the possibility to make flexible changes to the cutting lengths and the synchronous phase during the cutting process. Any changes are accepted after the synchronous phase and effective for the next step. The synchronous speed of the sealing roller is calculated as a function of the roller circumference in parameter *IrCrossCutterCircumference*, the speed of the master axis and the cycle length of the sealing roller.

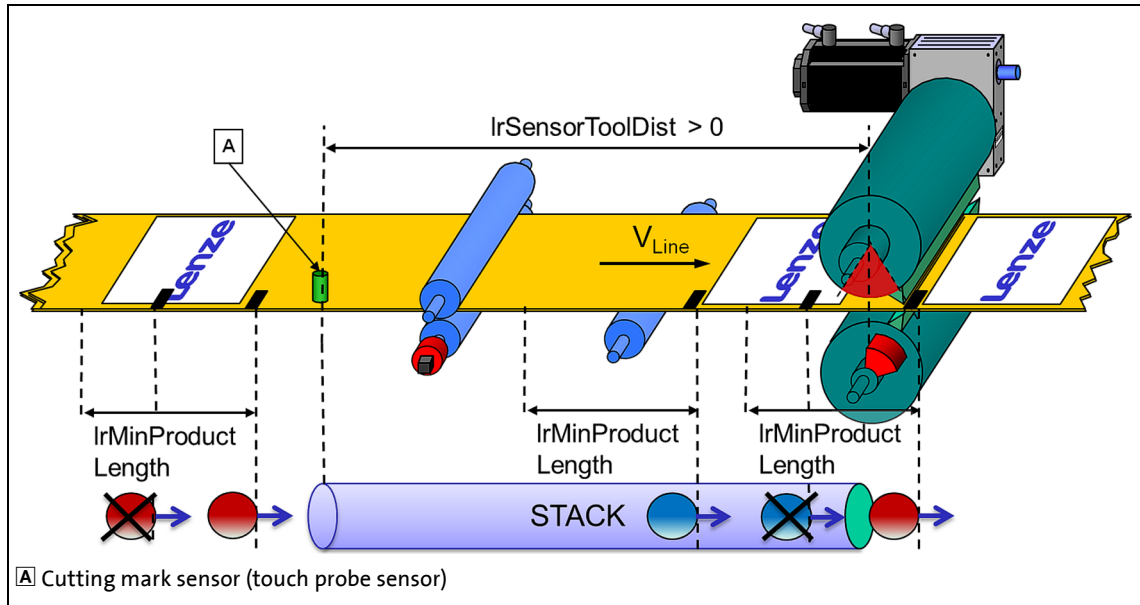
Select the touch probe source used to detect the marks via parameter *eTpMode*.

► [Selecting the touch probe source](#) (□ 74)

The input *xSyncCrossCutter* = TRUE serves to calculate the start cam from the current position in the synchronous phase. The cutting process starts as soon as a valid mark (detected touch probe with the current position) is present. The cross cutter makes the cuts on the detected marks. If no other valid marks are detected, the cross cutter is moved to the parking position. The cross cutter remains at standstill until another valid mark is detected. A valid mark serves to re-initiate the cutting process.

All valid marks are stored in a register. The register can keep a maximum of 100 marks. After each cut, the next mark from the register is processed for the next step.

Several marks can be stored and processed consecutively during the cutting process. In order to hide marks, you must set a minimum distance between two marks using parameter *lrMinProductLength*. After a valid mark, the following marks within the minimum distance are masked out by the technology module.

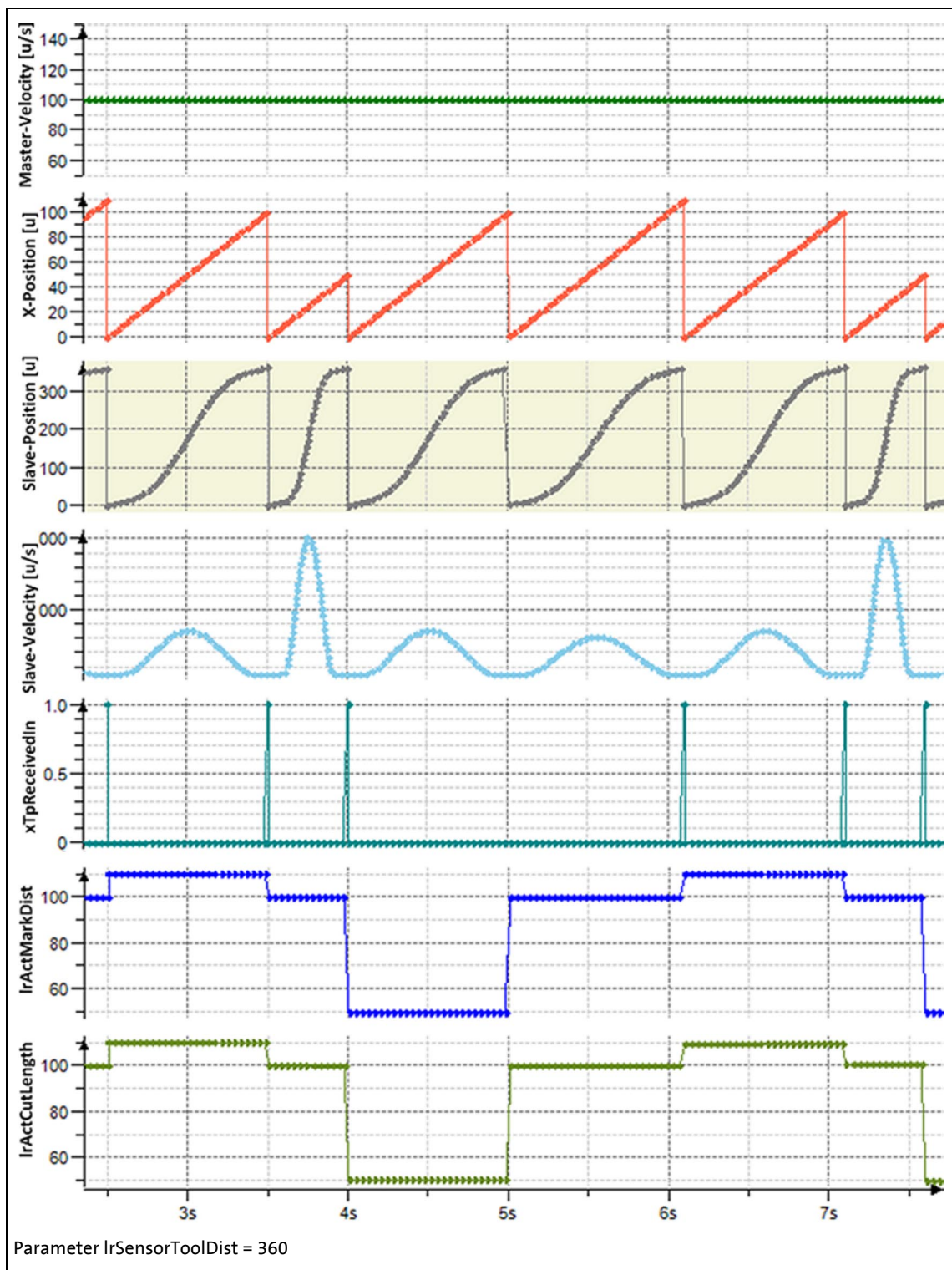


[3-36] Cutting on a mark with the minimum product length (parameter *lrMinProductLength*)

Parameters to be set

The parameters for the "cutting on a mark (event-controlled)" operating mode are located in the [L_TT1P_scPar_CrossCutter\[Base/State\]](#) (23) parameter structure.

```
xMasterAbsolute : BOOL := FALSE;
lrStartXPosRelative : LREAL := 0;
eCuttingMode : L_TT1P_CuttingModeState := 3;
eTpMode : L_TT1P_TpMode := 0;
lrSensorToolDist : LREAL := 0;
lrMinProductLength : LREAL := 50;
lrCrossCutterCircumference : LREAL := 360;
lrCuttingAngle : LREAL := 10;
byNumOfTools : BYTE := 1;
byStartToolNo : BYTE := 0;
```



[3-37] History: Event-controlled cutting process on a detected mark

3.30 Torque feedforward control

The torque feedforward control is activated by parameter *lrTrqCtrlGain* > 0 and the output *xTrqCtrlActive* is set to TRUE. The *lrTrqCtrlGain* parameter is the multiplicative gain factor of the feedforward control value. The feedforward control value is calculated in the axis by multiplying the mass inertia with the acceleration/deceleration of the sealing roller. The mass inertia of the sealing roller is defined by the *lrMInertia* parameter in kg/cm². The feedforward control value is put out via the *lrSetMotorPreCtrlTrqOut* output.

The acceleration/deceleration of the sealing roller is pre-estimated from the cam information. The time difference at which the acceleration/deceleration is pre-estimated is defined by the *lrTrqCtrlRateTime* parameter in seconds. If the parameter *lrTrqCtrlRateTime* is set to 0, the current acceleration/deceleration setpoints of the sealing roller are used.

As the feedforward control values are calculated in real time there is the possibility to make flexible changes to the parameters for the torque feedforward control during the cutting process. Any changes are immediately effective. In this way, the feedforward control values can be adapted and optimised during the cutting process.

Parameters to be set

The parameters to be set for torque feedforward control are located in the [L TT1P_scPar_CrossCutter\[Base/State\]](#) (□ 23) parameter structure.

```
lrTrqCtrlGain : LREAL := 0;  
lrTrqCtrlRateTime : LREAL := 0;  
lrMInertia : LREAL := 0;           // Constant MInertia J [kg/cm^2]
```

3.31 CPU utilisation (example Controller 3231 C)

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

Versions	Wiring of the technology module	CPU utilisation	
		Average	Maximum peak
Base	xEnable := TRUE; xRegulatorOn := TRUE; xSyncCrossCutter := TRUE;	90 µs	155 µs
State	xEnable := TRUE; xRegulatorOn := TRUE; xSyncCrossCutter := TRUE;	120 µs	172 µs

Symbole

"Cutting on a mark (event-controlled)" operating mode [80](#)
"Flow packer" operating mode [45](#)
"Length cutting with mark correction" operating mode [69](#)
"Length cutting" operating mode [42](#)

A

Absolute positioning of the cutting position [62](#)
Access points [37](#)
Application notes [7](#)

C

Calculating extreme values of a cam [68](#)
Calculating the cutting length [68](#)
Clutching the cross cutter to moving material [51](#)
Clutching to moving material [51](#)
Controlled start of the axes [14](#)
Conventions used [6](#)
CPU utilisation (example Controller 3231 C) [84](#)
Cross Cutter (functional description) [10](#)
Cutting mark register [72](#)
Cutting on a mark (operating mode) [80](#)
Cutting position via master offset (absolute positioning) [62](#)

D

Declutching (cross cutter) [53](#)
Declutching the cross cutter [53](#)
Defining partial cutting lengths in product length (print/cut format) [71](#)
Distribution of tools on the sealing roller [39](#)
Document history [5](#)

E

E-mail to Lenze [87](#)
Emergency opening operation [55](#)
eTMState [30](#)
Extreme values of a cam (calculation) [68](#)

F

Feedback to Lenze [87](#)
Flow packer (operating mode) [45](#)
Function block L_TT1P_CrossCutterBase/State [15](#)
Functional description of "Cross Cutter" [10](#)

G

Gearbox factor correction [77](#)

H

Homing [41](#)

I

Increasing/decreasing the speed (overspeed) [66](#)
Inputs [17](#)
Inputs and outputs [16](#)

L

L_TT1P_CrossCutterBase [15](#)
L_TT1P_CrossCutterState [15](#)
L_TT1P_scAP_CrossCutterBase [37](#)
L_TT1P_scAP_CrossCutterState [37](#)
L_TT1P_scPar_CrossCutterBase [23](#)
L_TT1P_scPar_CrossCutterState [23](#)
L_TT1P_scSF_CrossCutterBase [33](#)
L_TT1P_scSF_CrossCutterState [33](#)
Layout of the safety instructions [7](#)
Length cutting (operating mode) [42](#)
Length cutting with mark correction (operating mode) [69](#)

M

Manual jog (jogging) [41](#)
Master offset (absolute positioning) [62](#)
Method of elimination [64](#)

N

Notes on how to operate the technology module [13](#)

O

Offset of the cutting position [57](#)
Operating mode [13](#)
Outputs [20](#)
Overspeed (increasing/decreasing the speed) [66](#)

P

Parameter structure L_TT1P_scPar_CrossCutterBase/State [23](#)
Parameterisation of the sealing roller [38](#)
Parameterise the sealing roller [38](#)
Parking position (positioning) [39](#)
Position cut trimming [58](#)
Positioning to static material [49](#)
Positioning to the parking position [39](#)
Positive opening operation [55](#)
Print/cut format (partial cutting length in product length) [71](#)

S

Safety instructions [7](#), [8](#)
Selecting the touch probe source [74](#)
Signal flow diagrams [31](#)
Speed cut trimming [60](#)
Speed synchronism of master and register axis [76](#)
Start of the axes [14](#)
State machine [28](#)
States [28](#)
States of the eTMState output [30](#)
Structure of the access points L_TT1P_scAP_CrossCutterBase/State [37](#)
Structure of the signal flow L_TT1P_scSF_CrossCutterBase/State [33](#)

Index

T

Target group [4](#)

Technology module functions (overview) [12](#)

Tip trimming [60](#)

Top cut [44](#)

Torque feedforward control [83](#)

Touch probe mark (referencing) [75](#)

Trimming of the cutting position [57](#)

V

Variable names [6](#)



Your opinion is important to us

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

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

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, 31763 Hameln
Hans-Lenze-Straße 1, 31855 Aerzen
Germany
HR Hannover B 205381
☎ +49 5154 82-0
📠 +49 5154 82-2800
✉ sales.de@lenze.com
🌐 www.lenze.com

Service

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