FAST Application Software

Technology module

Winder Tension-controlled __________________

Reference Manual EN

13531757

Lenze

Contents

1 About this documentation

This documentation ...

- contains detailed information on the functionalities of the "Winder Tension-controlled" 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	
PDF	**Product catalogues** • Controller-based Automation • Controllers • Inverter Drives/Servo Drives
Mounting and wiring	
Printed	**Mounting instructions** • Controllers • Communication cards (MC-xxx) • I/O system 1000 (EPM-Sxxx) • Inverter Drives/Servo Drives • Communication modules
PDF	**Hardware manuals** • Inverter Drives/Servo Drives
Parameter setting / configuration / commissioning	
PDF	**Online help/reference manuals** • Controllers • Inverter Drives/Servo Drives • I/O system 1000 (EPM-Sxxx)
PDF	**Online help/communication manuals** • Bus systems • Communication modules
Sample applications and templates	
PDF	**Online help / software and reference manuals** • i700 application sample • Application Samples 8400/9400 • FAST Application Template Lenze/PackML • FAST technology modules

Symbols:

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

Tip!

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

www.lenze.com

Target group

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

1.1 Document history

Version			Description
4.3	05/2017	TD17	• Content structure has been changed. • General revisions • Figure Signal flow of the technology module (📖 29) corrected. New: • "MaterialCounterAxis" input (AXIS_REF) • Sources for the material length counting (📖 41)
4.2	11/2016	TD17	• General revisions • Parameter L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) supplemented.
4.1	04/2016	TD17	• General revisions • Figure Signal flow of the technology module (📖 29) corrected. • Access points L_TT1P_scAP_WinderTensionCtrl[Base/State/High] (📖 32) added.
4.0	11/2015	TD17	• General revisions • New: System deviation within the reduced sensitivity (📖 61) • Content structure has been changed.
3.0	05/2015	TD17	• General revisions • New: Material length counter (📖 40)
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

1.2 Conventions used

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

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

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

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

Variable names

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

1.3 Definition of the notes used

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

Safety instructions

Layout of the safety instructions:

Pictograph and signal word!

(characterise the type and severity of danger)

Note

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

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

Stop!

Damage or destruction of machine parts

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

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

Protective measures

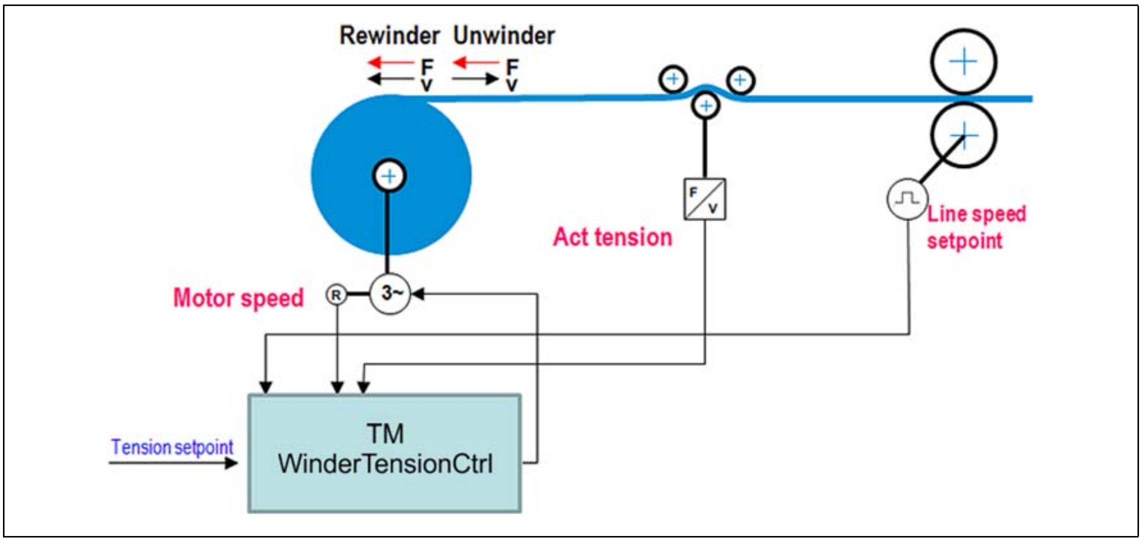
- Always switch off the voltage supply before working on the components of the automation system.
- Do not touch electronic components and contacts unless ESD measures were taken beforehand.
- If required, provide systems with installed inverters with additional monitoring and protective devices according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

3 Functional description for "Winder Tension-controlled"

In many technological processes, winding drives are a key component of an entire system. Depending on the material and the winding process, different open and closed-loop control methods are used:

- Dancer position control
- Open-loop tension control
- Closed-loop tension control

This technology module can be used to design a tension-controlled (open loop)/tension-controlled (closed-loop) winding drive.



[3-1] Structure of a tension-controlled (open loop)/tension-controlled (open loop) winder

The material is rewound or unwound by a centre-driven winding shaft. The winding motor torque directly determines the tensile force at the material. The winder speed changes reciprocally to the diameter (n~1/d); the torque increases proportionally with the diameter (M~d).

The speed in the machine module is not determined by the winder. The material must have a fixed point so that a web tension can be built via the winder.

- The "Base" version provides open loop tension control with acceleration compensation and a linear friction compensation. The open loop tension control can be set via a linear characteristic function. For diameter calculation, the line velocity and winder speed are included.
- The "State" version provides an extended functional range of the "Base" version. Here, three characteristics are available for open loop tension control:
  - Characteristic for a linear tensile force profile
  - Characteristic for a linear torque profile
  - User-definable characteristic with 64 grid points

  For closed loop tension control a PI controller can be used. In addition, the friction characteristic (as a function of the winder speed) can be identified and used to compensate the friction.
- The "High" version additionally offers the option to identify the moment of inertia of the winder axis and to integrate this value into the parameterisation of the technology module. Furthermore, the speed controller gain can be adapted during operation as a function of the current moment of inertia.

Speed control

A higher-level speed control only takes effect in the event of a web break to limit the speed of the drive. In order to avoid that the setpoint torque is influenced by the speed limitation during normal operation, a speed offset must be added to the speed setpoint calculated from the current line speed and the current diameter.

The torque setpoint is composed of the tensile force setpoint multiplied by the current radius, the correcting signal for compensating the mechanical friction, and the correcting signal for compensating the acceleration torque.

Closed-loop tension control

In order to achieve a good winding result, the friction and acceleration compensation must not substantially exceed the lowest load torque. If too great deviations with regard to the tensile force in spite of friction and acceleration compensation are to be expected or are identified, the tensile force setpoint can be corrected correspondingly by recording and closed-loop control of the tensile force.

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 that are assigned to the "Base", "State", and "High" versions:

Functionality	Versions		
	Base	State	High
Defining the winding direction (winding/unwinding) (🕮 34)	●	●	●
Automatic detection of the winding direction (🕮 34)	●	●	●
Defining the material feed to the winder (🕮 35)	●	●	●
Master value source for diameter calculation (🕮 36)	●	●	●
Speed feedforward control (🕮 36)	●	●	●
Calculation of the diameter (🕮 37)	●	●	●
Holding the diameter (🕮 38)	●	●	●
Defining the diameter / signal from the diameter sensor (🕮 39)	●	●	●
Material length counter (🕮 40)	●	●	●
Sources for the material length counting (🕮 41)	●	●	●
Manual jog (jogging) (🕮 43)	●	●	●
Synchronisation to the line velocity (🕮 44)	●	●	●
Trimming (🕮 45)	●	●	●
Tension control via characteristic function (Base version) (🕮 46)	●	●	●
Acceleration compensation (🕮 49)	●	●	●
Web break monitoring (🕮 51)	●	●	●
Persistent variables (🕮 52)	●	●	●
Tension control via characteristic function/winding characteristic (🕮 49)		●	●
Friction identification and compensation (🕮 54)		●	●
PI controller for tension control (🕮 57)		●	●
Identification of the moments of inertia (🕮 58)			●
Adaptation of the speed controller gain (🕮 60)			●
System deviation within the reduced sensitivity			●

»PLC Designer« Online help

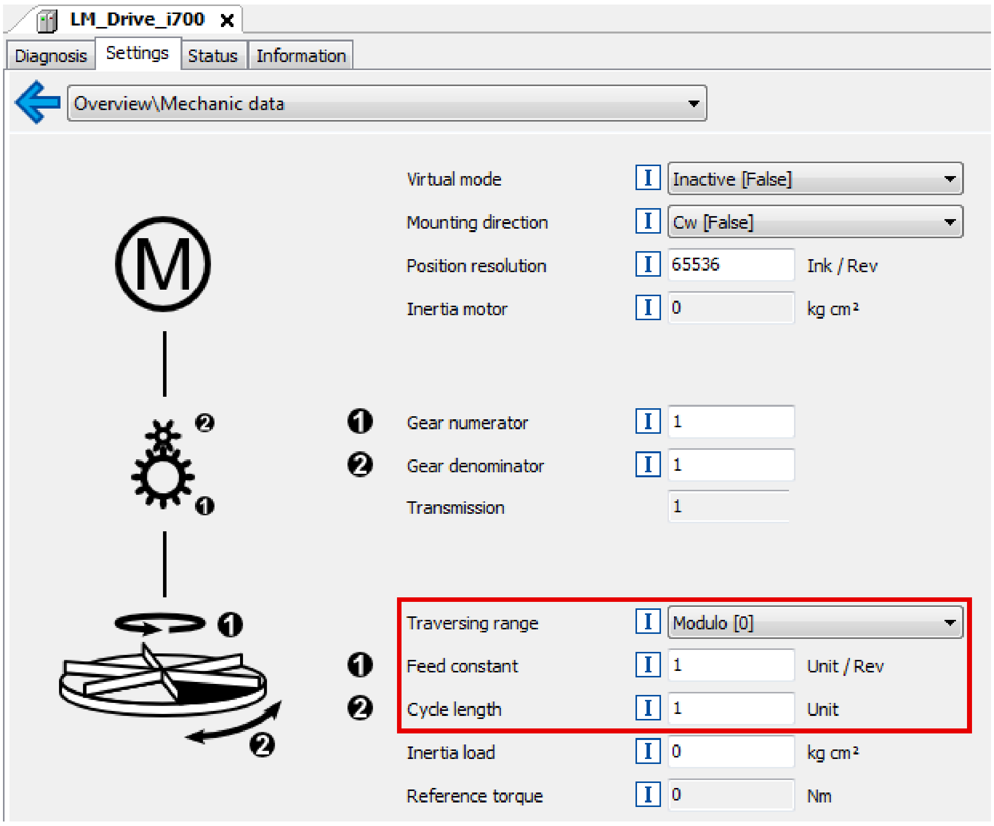
Here you will find detailed information on the **L_MC1P_AxisBasicControl** function block, the **stop function** and the **holding function**.

3.2 Important notes on how to operate the technology module

The technology module ...

- does not support the simulation mode in the »PLC Designer«;
- does not support any virtual axes;
- only supports rotary winder axes.

Go to the »PLC Designer« fand set the following parameters for each axis under the **Settings** tab:



- The feed of the winder axis is parameterised in the unit [revs/s].
- The line velocity is parameterised in the unit [mm/s].

Setting of the operating mode

The operating mode for the winder axis has to be set to "cyclically synchronous position" (csp) because the axis is led via the master position, master velocity and master torque 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↗TRUE edge.

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

Example Manual jog (jogging) (📖 43)**:**

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

3.3 L_TT1P_WinderTensionCtrl[Base/State/High] function block

The figure shows the relation of the inputs and outputs to the "Base", "State" and "High" versions. The additional inputs and outputs of the "State" and "High" versions are shaded.

L_TT1P_WinderTensionCtrl[Base/State/High]

Type	Input	Output	Type
Base			
BOOL	xEnableInternalControl	xInternalControlActive	BOOL
BOOL	xEnable	eTMState	L_TT1P_States
scCtrl_ABC	scCtrlABC	scStatusABC	scStatus_ABC
BOOL	xResetError	xError	BOOL
BOOL	xRegulatorOn	xWarning	BOOL
BOOL	xStop	eErrorID	L_IE1P_Error
BOOL	xHalt	scErrorInfo	L_TT1P_scErrorInfo
L_TT1P_scPar_WinderTensionCtrl[Base/State/High]	scPar	xAxisEnabled	BOOL
AXIS_REF	Axis	scSignalFlow	L_TT1P_scSF_WinderTensionCtrl[Base/State/High]
L_TT1P_PersistentVarWinder	PersistentVar	xDone	BOOL
L_TT1P_scAP_WinderTensionCtrl[Base/State/High]	scAccessPoints	xBusy	BOOL
AXIS_REF	MaterialCounterAxis	xSynchronised	BOOL
BOOL	xMaterialFeeding	xAccDecSync	BOOL
BOOL	xWindingDirection	xUnwind	BOOL
BOOL	xWindingDirectionStandstill	xWebBreak	BOOL
BOOL	xLoadDiam	xHoldDiamActive	BOOL
LREAL	lrSetDiam	xDiamMax	BOOL
BOOL	xHoldDiam	xDiamMin	BOOL
BOOL	xDiamCalcReduced	lrSetDiamOut	LREAL
BOOL	xSetMaterialCounter	lrSetDiamScaledOut	LREAL
BOOL	xWebBreakMonit	lrMaterialCounter	LREAL
BOOL	xJogLinePos	lrSetLineVelScaledOut	LREAL
BOOL	xJogLineNeg	lrActTotalTrqScaled	LREAL
BOOL	xTrimLinePos	lrWndSpdRef	LREAL
BOOL	xTrimLineNeg	xWndSpdLimit	BOOL
BOOL	xSyncLineVel	lrSetMInertiaOut	LREAL
BOOL	xTensCurve	lrSetTensScaledOut	LREAL
LREAL	lrSetTens	lrSetTensTrqScaledOut	LREAL
LREAL	lrSetLineVel	lrSetFricTrqScaledOut	LREAL
BOOL	xTensCtrl	lrSetAccTrqScaledOut	LREAL
BOOL	xFricCmps		
BOOL	xAccCmps		
LREAL	lrMInertiaAdapt		
State			
ARRAY [1..66] OF LREAL	alrFrictionCurve1	alrFrictionCurve1	ARRAY [1..66] OF LREAL

Input type	L_TT1P_WinderTensionCtrl[Base/State/High]		Output type
ARRAY [1..66] OF LREAL	alrFrictionCurve2	alrFrictionCurve2	ARRAY [1..66] OF LREAL
ARRAY [1..66] OF LREAL	alrFrictionCurve3	alrFrictionCurve3	ARRAY [1..66] OF LREAL
ARRAY [1..66] OF LREAL	alrFrictionCurve4	alrFrictionCurve4	ARRAY [1..66] OF LREAL
LREAL	lrActTensIn	lrActTensScaled	LREAL
LREAL	lrSetFricAdapt	lrSetTensCtrlScaledOut	LREAL
LREAL	lrTensCtrlInfluence		
BOOL	xResetICtrl		
BOOL	xExecuteIdentFric		
High			
BOOL	xExecuteIdentMInertia	lrIdentMInertia	LREAL
BOOL	xAdaptSpdCtrlGain	lrSetSpdCtrlGainAdaptOut	LREAL

3.3.1 Inputs and outputs

Designator Data type	Description	Available in version		
		Base	State	High
Axis AXIS_REF	Reference to the axis	●	●	●
PersistentVar L_TT1P_PersistentVarWinder	Reference to persistent variables In the reference, the following data are managed: • Diameter calculated	●	●	●
alrFrictionCurve1 ARRAY [1..66] OF LREAL	Friction characteristics 1 ... 4 consisting of 65 interpolation points in each case The interpolation points are entered automatically in the course of the friction identification process.		●	●
alrFrictionCurve1 ARRAY [1..66] OF LREAL				
alrFrictionCurve1 ARRAY [1..66] OF LREAL				
alrFrictionCurve1 ARRAY [1..66] OF LREAL				

3.3.2 Inputs

Designator Data type	Description		Available in version Base	State	High
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 FALSE	 The function block is executed. The function block is not executed.	●	●	●
scCtrlABC scCtrl_ABC	Input structure for the **L_MC1P_AxisBasicControl** function block • scCtrlABC can be used in "Ready" state. • If there is a request, the state changes to "Service". • The state change from "Service" back to "Ready" takes place if there are no more requests.		●	●	●
xResetError BOOL	TRUE	Reset axis error or software error.	●	●	●
xRegulatorOn BOOL	TRUE	Activate controller enable of the axis (via the **MC_Power** function block).	●	●	●
xStop BOOL	TRUE	Cancel the active movement and brake the axis to a standstill with the deceleration defined via the lrStopDec parameter. • The state changes to "Stop". • The technology module remains in the "Stop" state as long as xStop is set to TRUE (or xHalt = TRUE). • The input is also active with "Internal Control".	●	●	●
xHalt BOOL	TRUE	Cancel the active movement and brake the axis to a standstill with the deceleration defined via the lrHaltDec parameter. • The state changes to "Stop". • The technology module remains in the "Stop" state as long as xStop is set to TRUE (or xHalt = TRUE).	●	●	●
scPar L_TT1P_scPar_WinderTensionCtrl[Base/State/High]	The parameter structure contains the parameters of the technology module. The data type depends on the version used (Base/State/High).		●	●	●
scAccessPoints L_TT1P_scAP_WinderTensionCtrl[Base/State/High]	Structure of the access points The data type depends on the version used (Base/State/High).		●	●	●
MaterialCounterAxis AXIS_REF	Here a modulo axis of a measuring wheel can be connected to the material. If an axis is connected to the input, the material length is increased on the basis of the data from the reference axis. This procedure is also suitable for noisy signals. If no axis is connected here, the material length is determined from the integration of the material speed (lrSetLineVel or lrSetLineVelDiamCalc input). ▸ Material length counter (🕮 40) ▸ Sources for the material length counting (🕮 41)		●	●	●
xMaterialFeeding BOOL	Material feeding at the reels from the top or from the bottom TRUE FALSE	 Material feeding from the top Material feeding from the bottom	●	●	●
xWindingDirection BOOL	Winder function at positive line velocity (lrSetLineVel input > 0) TRUE FALSE	 Unwinder Rewinder	●	●	●

Designator / Data type	Description		Available in version		
			Base	State	High
xWindingDirectionStandstill BOOL	Function of the winder with idle line speed (lrSetLineVel input = 0)		●	●	●
	TRUE	Unwinder			
	FALSE	Rewinder			
xLoadDiam BOOL	TRUE	Load the (start) diameter from the lrSetDiam input. • Unit: mm	●	●	●
lrSetDiam LREAL	Defining a (start) diameter The diameter is loaded cyclically when the xLoadDiam is set to TRUE. • Unit: mm		●	●	●
xHoldDiam BOOL	TRUE	Current diameter is held.	●	●	●
xDiamCalcReduced BOOL	Change-over of the diameter calculation mode between long/ short distance		●	●	●
	TRUE	Diameter is updated after the short distance.			
	FALSE	Diameter is updated after the long distance.			
xSetMaterialCounter BOOL	The input is edge-controlled and evaluates the FALSE↗TRUE edge.		●	●	●
	TRUE	Sets the material length counter (lrMaterialCounter output) to the value set under the lrSetMaterialPos parameter.			
xWebBreakMonit BOOL	TRUE	Activate web break monitoring.	●	●	●
xJogLinePos BOOL	TRUE	Drive axis in positive material flow direction (manual jog). If xJogLineNeg is also TRUE, the traversing direction selected first remains set.	●	●	●
xJogLineNeg BOOL	TRUE	Drive axis in negative material flow direction (manual jog). If xJogLinePos is also TRUE, the traversing direction selected first remains set.	●	●	●
xTrimLinePos BOOL	TRUE	Enable the velocity offset in positive material flow direction when the winder axis is synchronised with the line (xSyncLineVel = TRUE)	●	●	●
xTrimLineNeg BOOL	TRUE	Enable the velocity offset in negative material flow direction when the winder axis is synchronised with the line (xSyncLineVel = TRUE)	●	●	●
xSyncLineVel BOOL	TRUE	Synchronise winder axis with the line.	●	●	●
xTensCurve BOOL	TRUE	Enable tensile force characteristic.	●	●	●
lrSetTens LREAL	Tensile force setpoint • Unit: N		●	●	●
lrSetLineVel LREAL	Current line velocity • Unit: mm/s		●	●	●
xTensCtrl BOOL	TRUE	Activate open loop/closed loop tension control.	●	●	●
xFricCmps BOOL	TRUE	Activate friction compensation during the open loop/ closed loop tension control process.	●	●	●
xAccCmps BOOL	TRUE	Activate acceleration compensation during the open loop/closed loop tension control process.	●	●	●

Designator Data type	Description		Available in version Base	 State	 High
lrMInertiaAdapt LREAL	Multiplier for the current moment of inertia		●	●	●
lrActTensIn LREAL	Current actual tensile force value • Unit: N			●	●
lrSetFricAdapt LREAL	Multiplier for the friction characteristic			●	●
lrTensCtrlInfluence LREAL	Influence of the tensile force process controller • Unit: x 100 % (1 = 100 %) • Initial value: 0			●	●
xResetICtrl BOOL	TRUE	The I component of the PI controller is switched off and the correcting variable (output of the controller) from the I component is led to '0' via the ramp function. The correcting variable from the P component is not affected.		●	●
xExecuteIdentFric BOOL	The input is edge-controlled and evaluates the rising edge.			●	●
	FALSE↗TRUE	Activate identification of the friction characteristic.			
xExecuteIdentMInertia BOOL	The input is edge-controlled and evaluates the rising edge.				●
	FALSE↗TRUE	The moment of inertia at the winder shaft is detected. The lrIdentMInertia output displays the detected moment of inertia in $kgcm^2$.			
xAdaptSpdCtrlGain BOOL	TRUE	Switch on adaptation of the speed controller gain.			●

3.3.3 Outputs

Designator / Data type	Description		Available in version: Base	State	High
xInternalControlActive BOOL	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 (📖 27)		●	●	●
scStatusABC scStatus_ABC	Structure of the status data of the **L_MC1P_AxisBasicControl** function block		●	●	●
xError BOOL	TRUE	There is an error in the technology module.	●	●	●
xWarning BOOL	TRUE	There is a warning in the technology module.	●	●	●
eErrorID L_IE1P_Error	ID of the error or warning message if xError = TRUE or xWarning = TRUE. **"FAST technology modules" reference manual:** Here you can find information on error or warning messages.		●	●	●
scErrorInfo L_TT1P_scErrorInfo	Error information structure for a more detailed analysis of the error cause		●	●	●
xAxisEnabled BOOL	TRUE	The axis is enabled.	●	●	●
scSignalFlow L_TT1P_scSF_WinderTensionCtrl[Base/State/High]	Structure of the signal flow The data type depends on the version used (Base/State/High). ▸ Signal flow diagrams (📖 28)		●	●	●
xDone BOOL	TRUE	The request/action has been completed successfully.	●	●	●
xBusy BOOL	TRUE	The request/action is currently being executed.	●	●	●
xSynchronised BOOL	TRUE	The winder is synchronised with the line speed.	●	●	●
xAccDecSync BOOL	TRUE	The synchronisation function is active. Synchronisation of the winder is carried out or cancelled.	●	●	●
xUnwind BOOL	Status bit for unwinder and rewinder		●	●	●
	TRUE	Unwinder			
	FALSE	Rewinder			
xWebBreak BOOL	TRUE	A web break has occurred.	●	●	●
xHoldDiamActive BOOL	TRUE	Current diameter is held.	●	●	●
xDiamMax BOOL	TRUE	The maximum diameter has been reached.	●	●	●
xDiamMin BOOL	TRUE	The minimum diameter has been reached.	●	●	●
lrSetDiamOut LREAL	Current diameter calculated • Unit: mm		●	●	●
lrSetDiamScaledOut LREAL	Current diameter calculated and scaled • Unit: x 100 % • 1 = 100 % = parameter lrMaxDiam		●	●	●

Designator Data type	Description		Available in version		
			Base	State	High
lrMaterialCounter LREAL	Display of the material length counter content on the winder Depending on the Defining the winding direction (winding/unwinding) (34), the material length counter is incremented or decremented. • Unit: mm		●	●	●
lrSetLineVelScaledOut LREAL	Current line velocity scaled • Unit: x 100 % • 1 = 100 % = parameter lrLineVelRef		●	●	●
lrActTotalTrqScaled LREAL	Current scaled torque of the winder shaft • Reference variable: Rated/reference torque of the motor. • Unit: x 100 % (1 = 100 %)		●	●	●
lrWndSpdRef LREAL	Reference of the winder speed at minimum diameter and maximum line velocity.		●	●	●
xWndSpdLimit BOOL	TRUE	The winder speed has reached the speed limitation.	●	●	●
lrSetMInertiaOut LREAL	Current moment of inertia at the winder shaft • Unit: $kgcm^2$		●	●	●
lrSetTensScaledOut LREAL	Current scaled tensile force • Unit: x 100 % • 1 = 100 % = lrTensRef parameter		●	●	●
lrSetTensTrqScaledOut LREAL	Resulting torque proportion from the tensile force scaled to the rated motor torque • Unit: Nm		●	●	●
lrSetFricTrqScaledOut LREAL	Resulting torque proportion from the friction compensation scaled to the rated motor torque • Unit: Nm		●	●	●
lrSetAccTrqScaledOut LREAL	Resulting torque proportion from the acceleration compensation scaled to the rated motor torque • Unit: Nm		●	●	●
lrActTensScaled LREAL	Current scaled tensile force • Unit: x 100 % • 1 = 100 % = lrTensRef parameter			●	●
lrSetTensCtrlScaledOut LREAL	Current scaled manipulating variable of tension control • Unit: x 100 % • 1 = 100 % = lrTensRef parameter			●	●
lrIdentMInertia LREAL	Identified moment of inertia at the winder shaft • Unit: $kgcm^2$				●
lrSetSpdCtrlGainAdaptOut LREAL	Adaptation of the speed controller gain • Unit: x 100 % (1 = 100 %)				●

3.3.4 Parameters

L_TT1P_scPar_WinderTensionCtrl[Base/State/High]

The **L_TT1P_scPar_WinderTensionCtrl[Base/State/High]** structure contains the parameters of the technology module.

Designator Data type	Description	Available in version		
		Base	State	High
lrStopDec LREAL	Deceleration for the stop function and when hardware/software limit switches and the following error monitoring function are triggered • Unit: revs/s • Initial value: 10000	●	●	●
lrStopJerk LREAL	Jerk for the stop function and for the triggering of the hardware limit switches, software limit positions, and the following error monitoring function • Unit: revs/s^3 • Initial value: 100000	●	●	●
lrHaltDec LREAL	Deceleration for the holding function Specification of the maximum speed variation which is to be used for deceleration to standstill. • Unit: revs/s^2 • Initial value: 3600 • Only positive values are permissible.	●	●	●
lrJerk LREAL	Jerk for compensation of a holding function • Unit: revs/s^3 • Initial value: 100000	●	●	●
lrLineJerk LREAL	Jerk for manual jog and compensation of a trimming or clutch function • Unit: mm/s^3 • Initial value: 10000	●	●	●
lrJogLineAcc LREAL	Acceleration for manual jog Specification of the maximum speed variation which is to be used for acceleration. • Unit: mm/s^2 • Initial value: 100	●	●	●
lrJogLineDec LREAL	Deceleration for manual jog Specification of the maximum speed variation which is to be used for deceleration to standstill. • Unit: mm/s^2 • Initial value: 100	●	●	●
lrJogLineVel LREAL	Maximum speed to be used for manual jog. • Unit: mm/s • Initial value: 10	●	●	●
lrTrimLineAcc LREAL	Acceleration for trimming Selection of the speed change relative to the line speed to be used for accelerating. The acceleration acting on the drive is the sum of line and trimming acceleration. • Unit: mm/s^2 • Initial value: 100	●	●	●
lrTrimLineDec LREAL	Deceleration for trimming Selection of the speed change relative to the line speed to be used for decelerating. The deceleration acting on the drive is the sum of line and trimming acceleration. • Unit: mm/s^2 • Initial value: 100	●	●	●

Designator Data type	Description	Available in version		
		Base	State	High
lrTrimLineVel LREAL	Velocity for trimming Selection of the velocity used for trimming. • Unit: mm/s • Initial value: 10	●	●	●
lrSyncLineAcc LREAL	Acceleration for synchronising to line velocity • Unit: mm/s^2 • Initial value: 100	●	●	●
lrSyncLineDec LREAL	Deceleration for synchronising to line velocity • Unit: mm/s^2 • Initial value: 100	●	●	●
lrWebBreakWindow LREAL	Web break window The current diameter is compared with the previous diameter across the web break window. • Unit: x 100 % (1.0 = 100 %) • Initial value: 0.1 (10 %)	●	●	●
lrMaxDiam LREAL	Maximum diameter • Unit: mm • Initial value: 180	●	●	●
lrMinDiam LREAL	Minimum diameter • Unit: mm • Initial value: 50	●	●	●
rFiltTimeDiam REAL	PT1 filter time for the current diameter (lrSetDiamOut) • Unit: s • Initial value: 0.05	●	●	●
lrDiamCalcRegularDist LREAL	Regular calculation distance for the diameter • Unit: rev • Initial value: 1	●	●	●
lrDiamCalcReducedDist LREAL	Reduced calculation distance for the diameter • Unit: rev • Initial value: 0.1	●	●	●
alrAdaptDiamX ARRAY [1...9] OF LREAL	Grid points of the curve function for diameter loading • Values that may be applied to the analog input lrSetDiam. • Unit: mm • Initial values: 0, 100, 200, 300, 400, 500, 600, 700, 800	●	●	●
alrAdaptDiamY ARRAY [1...9] OF LREAL	Grid points of the curve function for diameter loading • Function values for the diameter • Unit: mm • Initial values: 0, 100, 200, 300, 400, 500, 600, 700, 800	●	●	●
lrTensCurveCtrlScaled LREAL	Gradient of the characteristic for the tension control open loop • Unit: x 100 % (1 = 100 %) • Initial value: 0 • The value '1' causes a constant tensile force profile and thus a proportionally increasing setpoint for the diameter.	●	●	●
lrTensCurveStartDiamScaled LREAL	Initial point of the characteristic for the tension control open loop • Unit: x 100 % • 1 = 100 % = parameter lrMaxDiam • Initial value: 0 (0 %)	●	●	●
lrLineVelRef LREAL	Maximum line velocity • Unit: mm/s • Initial value: 1000	●	●	●

Designator Data type	Description	Available in version		
		Base	State	High
lrMinLineVel LREAL	Minimum line velocity Up to this velocity, the diameter is held. • Unit: mm/s • Initial value: 1	●	●	●
rFiltTimeMaterialCounter REAL	Filter time constant for the material length counter (lrMaterialCounter output) • Initial value: 0 (filter is deactivated.)	●	●	●
lrSetMaterialPos LREAL	Position of the material length counter With a FALSE↗TRUE edge at the xSetMaterialCounter input, the material length counter (lrMaterialCounter output) is set to the value in lrSetMaterialPos. • Unit: mm	●	●	●
lrLineVelOffsetScaled LREAL	Line speed offset for speed limitation • Unit: x 100 % (1 = 100 % = lrLineVelRef parameter) • Initial value: 0.1	●	●	●
lrTensRef LREAL	Maximum permissible tensile force • Unit: N • Initial value: 2	●	●	●
lrTensRamp LREAL	Acceleration ramp for the tensile force setpoint • Unit: N/s • Initial value: 1	●	●	●
rFiltTimeWndSpd REAL	PT1 filter time for the winder shaft speed • Unit: s • Initial value: 0.01	●	●	●
rFiltTimeAccSpd REAL	PT1 filter time for the winder shaft speed for acceleration compensation • Unit: s • Initial value: 0.005	●	●	●
lrAccCmpsDeadBandTrq Scaled LREAL	Lagging range (dead band) for the current acceleration torque • Unit: Nm • Initial value: 0.1	●	●	●
lrAccCmpsGainAcc LREAL	Gain factor for the acceleration torque in positive direction • Unit: x 100 % (1.00 = 100 %) • Initial value: 1.05 (105 %)	●	●	●
lrAccCmpsGainDec LREAL	Gain factor for the acceleration torque in negative direction • Unit: x 100 % (1.00 = 100 %) • Initial value: 0.95 (95 %)	●	●	●
lrConstMInertia LREAL	Constant moment of inertia at the winder shaft • Unit: $kgcm^2$ • Initial value: 9	●	●	●
lrMaxMInertia LREAL	Maximally permissible moment of inertia at the winder shaft • Unit: $kgcm^2$ • Initial value: 50	●	●	●
lrFricCurveStartTrq LREAL	Initial friction for the linear friction compensation if the winder shaft is at a standstill. • Unit: Nm • Initial value: 0	●	●	●
lrFricCurveEndTrq LREAL	Final friction for the linear friction compensation if the winder shaft is at a standstill. • Unit: Nm • Initial value: 0	●	●	●

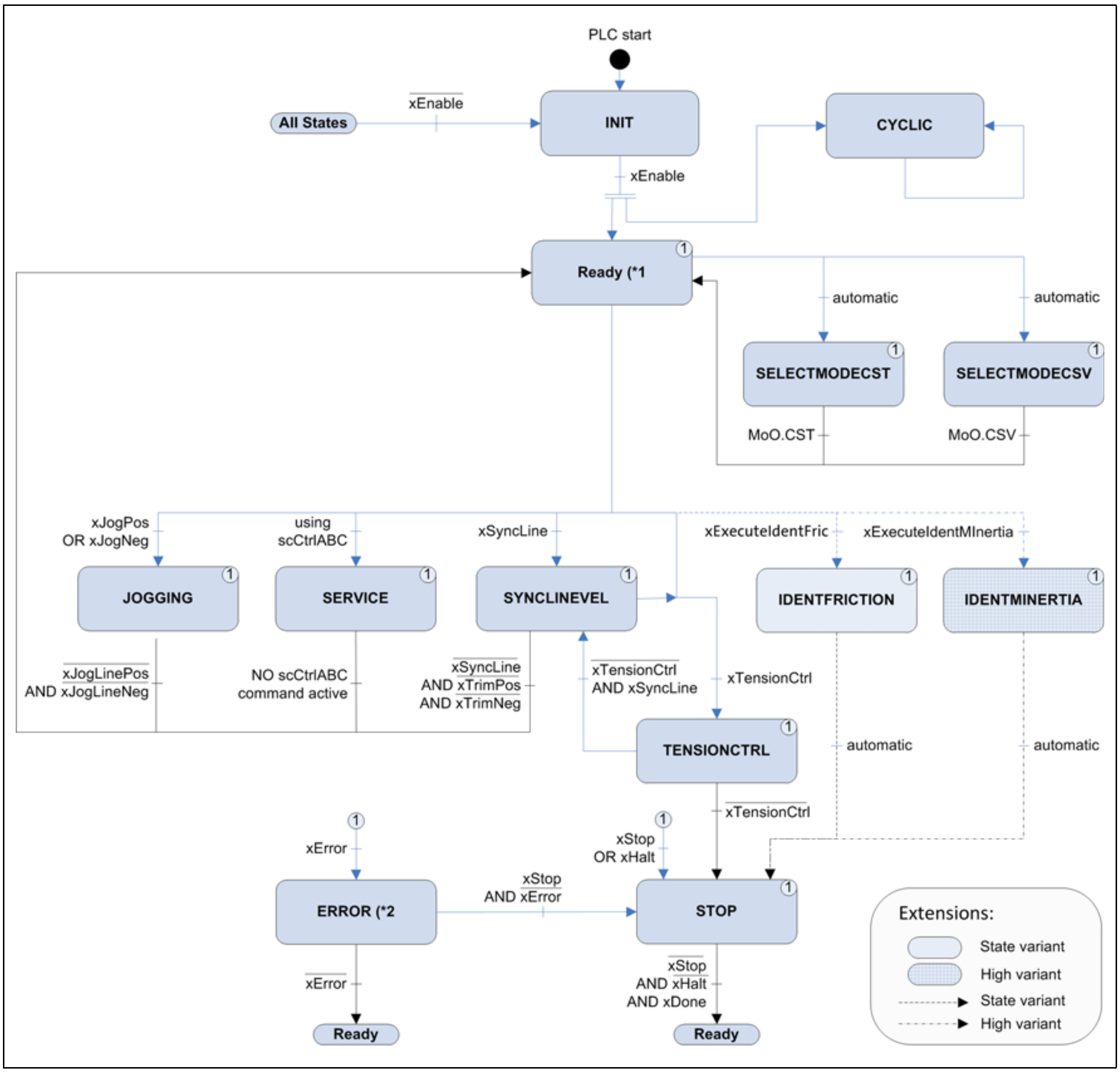
Designator Data type	Description		Available in version		
			Base	State	High
rFiltTimeFricSetSpd REAL	PT1 filter time for the winder shaft speed for friction compensation • Unit: s • Initial value: 0.01		●	●	●
dwSelectTensCurve DWORD	Selection of the characteristic for tension control • Initial value: 0			●	●
	0	Linear tensile force profile			
	1	Linear torque profile			
	2	Tensile force profile according to predefined characteristic			
alrTensCurve ARRAY [1...65] OF LREAL	Characteristic for tension control open loop consisting of 65 values.			●	●
lrTensCtrlGain LREAL	Controller gain • Initial value: 0			●	●
lrTensCtrlResetTime LREAL	Controller reset time • Unit: s • Initial value: 0 (reset time deactivated)			●	●
lrIdentFricMaxSpdScaled LREAL	Speed for friction identification • Unit: x 100 % (1 = 100 % = max. speed at the lrWndSpdRef output) • Initial value: 0.9			●	●
lrIdentFricAccDec LREAL	Acceleration for friction identification • Unit: revs/s^2 • Initial value: 1			●	●
rFiltTimeIdentFricSpd REAL	PT1 filter time for the winder shaft speed for friction identification • Unit: s • Initial value: 0.0			●	●
rFiltTimeIdentFricTrq REAL	PT1 filter time for the winder shaft friction torque for friction identification • Unit: s • Initial value: 0.05			●	●
dwSelectFricCurve DWORD	**For friction identification (xFricCmps input = TRUE):** Selection of the memory area in which the friction values identified are stored. A maximum of 4 characteristics can be stored.			●	●
	1 ... 4	Memory area 1 ... 4			
	For friction compensation (xExecuteIdentFric input = FALSE↗TRUE): Selection of the characteristic for friction compensation				
	0	Linear friction characteristic (adjustable via the lrFricCurveStartTrq parameter)			
	1 ... 4	Friction characteristic identified (memory area 1 ... 4)			
rFiltTimeActTensIn REAL	PT1 filter time for the current tensile force (lrActTensIn input) • Unit: s • Initial value: 0.005			●	●
lrActTensInGain LREAL	Gain factor for the current tensile force (lrActTensIn input) • Initial value: 1			●	●
lrActTensInOffset LREAL	Offset for the current tensile force (lrActTensIn input) • Initial value: 0			●	●

Designator Data type	Description	Available in version		
		Base	State	High
rFiltTimeIdentMInertiaSpd REAL	PT1 filter time for the speed at the winder shaft during the identification of the moment of inertia • Unit: s • Initial value: 0.01			●
rFiltTimeIdentMInertiaTrq REAL	PT1 filter time for the torque at the winder shaft during the identification of the moment of inertia • Unit: s • Initial value: 0.005			●
lrIdentMInertiaMaxSpd Scaled LREAL	Maximum speed of the winder shaft during the identification of the moment of inertia • Unit: x 100 % (1.0 = 100 % = lrWndSpdRef) • Initial value: 0.2 (20 %)			●
lrIdentMInertiaMaxTrq Scaled LREAL	Maximum torque of the winder shaft during the identification of the moment of inertia • Unit: x 100 % (1.0 = 100 %) • Initial value: 0.2 (20 %)			●
alrSpdCtrlGainAdaptX ARRAY [1...9] OF LREAL	Characteristic function for the speed control gain The X axis corresponds to the scaled moment of inertia. • Unit: x 100 % (1 = 100 % = parameter lrMaxMInertia) • Initial values: [0.0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6]			●

alrSpdCtrlGainAdaptY
ARRAY [1...9] OF LREAL
	Characteristic function for the speed control gain
	The Y axis corresponds to the gain factor of the speed controller.
	• Unit: x 100 % (1 = 100 %)
	• Initial values:
	  • [0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 0.98, 0.95, 0.95]
	  • Lower limitation: 0.5 = 50 %
	  • Upper limitation: 1.0 = 100 %
	Linear increase of the gain up to 100 % of the moment of inertia
				●

lrReducedGainWindow LREAL	Range of system deviation with reduced gain/sensitivity • Initial value: 0.0			●
lrReducedGain LREAL	Gain of system deviation within the reduced sensitivity • Initial value: 0.0			●
lrTensPosCtrlLimPos LREAL	Limitation of the tensile force controller correcting variable (output of the controller) in positive direction • Unit: [N] • Initial value: 2147483648 (0.5 x 2^{32})		●	●
lrTensPosCtrlLimNeg LREAL	Limitation of the tensile force controller correcting variable (output of the controller) in negative direction • Unit: [N] • Initial value: -2147483648 (-0.5 x 2^{32})		●	●

3.4 State machine



[3-2] State machine of the technology module

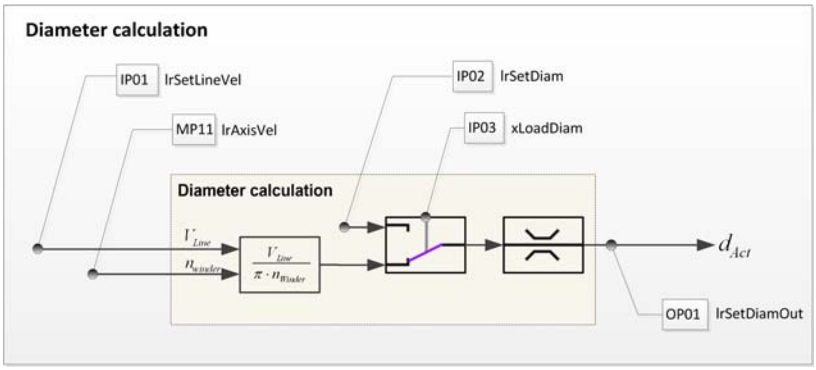
(*1 In the "Ready" state, xRegulatorOn has to be set to TRUE.

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

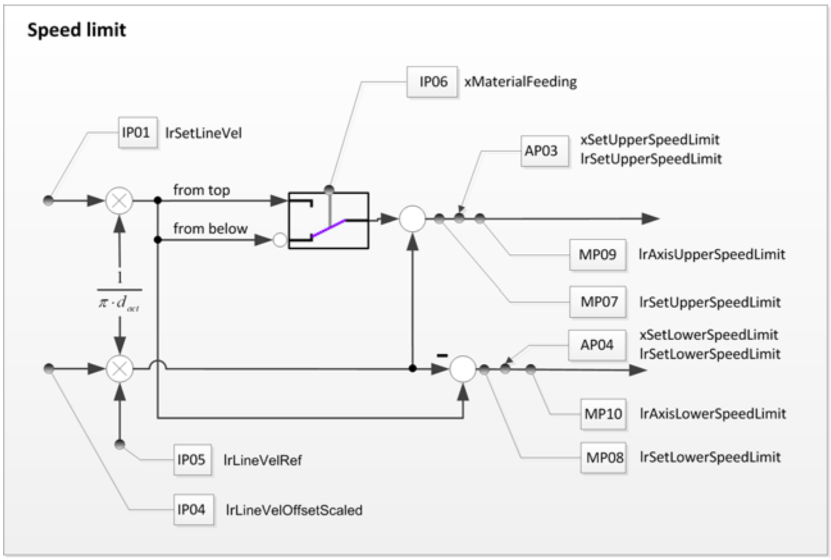
3.5 Signal flow diagrams

The illustrations [3-3] and [3-5] show the main signal flow of the implemented functions.

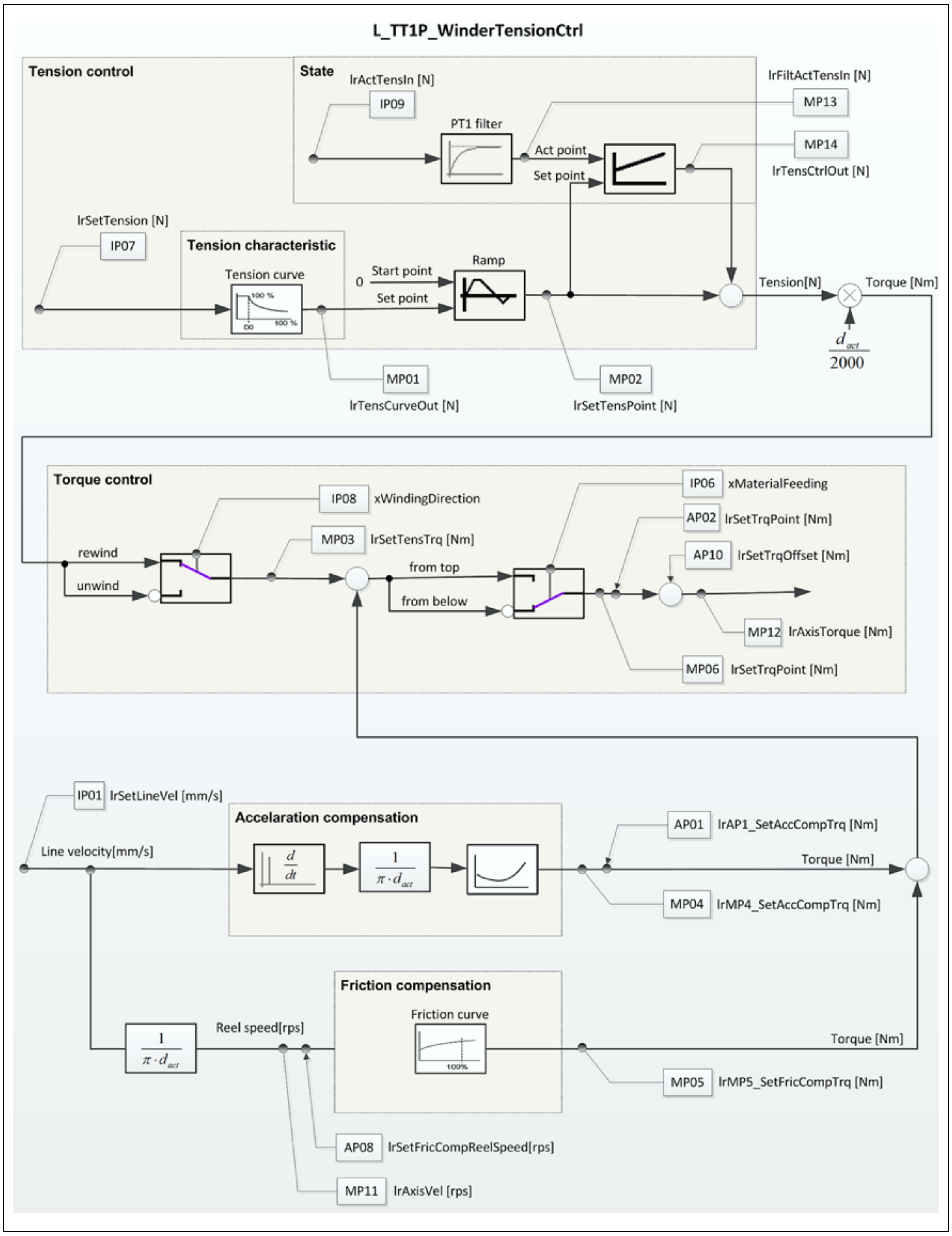
The signal flow of the additional functions such as "manual jog" is not displayed here.



[3-3] Signal flow for calculating the diameter



[3-4] Signal flow for speed limitations in the winding process



[3-5] Signal flow of the technology module

3.5.1 Structure of the signal flow

L_TT1P_scSF_WinderTensionCtrl[Base/State/High]

The contents of the **L_TT1P_scSF_WinderTensionCtrl[Base/State/High]** structure are read-only and offer a practical diagnostics option within the signal flow (Signal flow diagrams (📖 28)).

Designator / Data type	Description	Available in version: Base	State	High
IP01_lrSetLineVel LREAL	Current line velocity • Unit: mm/s	●	●	●
IP02_lrSetDiam LREAL	Defining a (start) diameter The diameter is loaded cyclically when the xLoadDiam is set to TRUE. • Unit: mm	●	●	●
IP03_xLoadDiam BOOL	TRUE: Load the (start) diameter from the lrSetDiam input. • Unit: mm	●	●	●
IP04_lrLineVelOffsetScaled LREAL	Line speed offset for speed limitation • Unit: x 100 % • 1 = 100 % = parameter lrLineVelRef • Initial value: 0.1	●	●	●
IP05_lrLineVelRef LREAL	Maximum line velocity • Unit: mm/s • Initial value: 1000	●	●	●
IP06_xMaterialFeeding BOOL	Material feeding at the reels from the top or from the bottom TRUE: Material feeding from the top FALSE: Material feeding from the bottom	●	●	●
IP07_lrSetTens LREAL	Tensile force setpoint • Unit: N	●	●	●
IP08_xWindingDirection BOOL	Winder function at positive line velocity (lrSetLineVel input > 0) TRUE: Unwinder FALSE: Rewinder	●	●	●
IP09_lrActTensIn LREAL	Current actual tensile force value • Unit: N		●	●
MP01_lrTensCurveOut LREAL	The tensile force evaluated with the tensile force characteristic. • Unit: N	●	●	●
MP02_lrSetTens LREAL	Tensile force setpoint • Unit: N	●	●	●
MP03_lrSetTensTrq LREAL	Resulting torque setpoint from the tensile force • Unit: Nm	●	●	●
MP04_lrSetAccCompTrq LREAL	Resulting torque setpoint from the acceleration compensation of the winding drive • Unit: Nm	●	●	●
MP05_lrSetFricCompTrq LREAL	Resulting torque setpoint from the friction compensation of the winding drive • Unit: Nm	●	●	●
MP06_lrSetTrqPoint LREAL	Resulting torque setpoint from the torque feedforward control • Unit: Nm	●	●	●
MP07_lrSetUpperSpeedLimit LREAL	Upper limit value for the speed limitation of the winding drive • Unit: revs/s	●	●	●
MP08_lrSetLowerSpeedLimit LREAL	Lower limit value for the speed limitation of the winding drive • Unit: revs/s	●	●	●

Designator Data type	Description	Available in version		
		Base	State	High
MP09_lrAxisUpperSpeed Limit LREAL	Upper speed limit value of the winding drive • Unit: revs/s	●	●	●
MP10_lrAxisLowerSpeed Limit LREAL	Lower speed limit value of the winding drive • Unit: revs/s	●	●	●
MP11_lrAxisVel LREAL	Speed of the winding drive • Unit: revs/s	●	●	●
MP12_lrAxisTroque LREAL	Torque of the winding drive • Unit: Nm	●	●	●
MP13_rFiltActTensIn REAL	Filtered current tensile force from the lrActTensIn input • Unit: N		●	●
MP14_lrTensCtrlOut LREAL	Setpoint tensile force for the winding drive • Unit: N		●	●
MP15_lrTensCtrlOutGain LREAL	Correcting variable of the proportional component (P component) of the tensile force controller • Unit: N		●	●
MP16_lrTensCtrlOutReset Time LREAL	Correcting variable of the integral-action component (I component) of the tensile force controller • Unit: N		●	●
MP17_lrTensCtrlOutRate Time LREAL	Correcting variable of the differential component (D component) of the tensile force controller • Unit: N		●	●
OP01_lrSetDiamOut LREAL	Current diameter calculated • Unit: mm	●	●	●

3.5.2 Structure of the access points

L_TT1P_scAP_WinderTensionCtrl[Base/State/High]

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	High
AP01_xSetAccCompTrq BOOL	Enable of the AP01_lrSetAccCompTrq access point		●	●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP01_lrSetAccCompTrq LREAL	Resulting torque setpoint from the acceleration compensation of the winding drive • Unit: Nm				
AP02_xSetTrqPoint BOOL	Enable of the AP02_lrSetTrqPoint access point		●	●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP02_lrSetTrqPoint LREAL	Resulting torque setpoint from the torque feedforward control • Unit: Nm				
AP03_xSetUpperSpeedLimit BOOL	Enable of the AP03_lrSetUpperSpeedLimit access point		●	●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP03_lrSetUpperSpeedLimit LREAL	Upper limit value for the speed limitation of the winding drive • Unit: revs/s				
AP04_xSetLowerSpeedLimit BOOL	Enable of the AP04_lrSetLowerSpeedLimit access point		●	●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP04_lrSetLowerSpeedLimit LREAL	Lower limit value for the speed limitation of the winding drive • Unit: revs/s				
AP05_xSetTensionCtrlOut Gain BOOL	Enable of the AP05_lrSetTensionCtrlOutGain access point			●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP05_lrSetTensionCtrlOut Gain LREAL	Cyclic loading of the correcting variable of the proportional component (P component) of the tensile force controller • Unit: N				
AP06_xSetTensionCtrlOut ResetTime BOOL	Enable of the AP06_lrSetTensionCtrlOutResetTime access point			●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP06_lrSetTensionCtrlOut ResetTime LREAL	Cyclic loading of the correcting variable of the integral-action component (I component) of the tensile force controller • Unit: N				
AP07: xSetTensionCtrlOutRate Time BOOL	Enable of the access point AP07: lrSetTensionCtrlOutRateTime			●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP07: lrSetTensionCtrlOutRate Time LREAL	Cyclic loading of the correcting variable of the differential component (D component) of the tensile force controller • Unit: N				

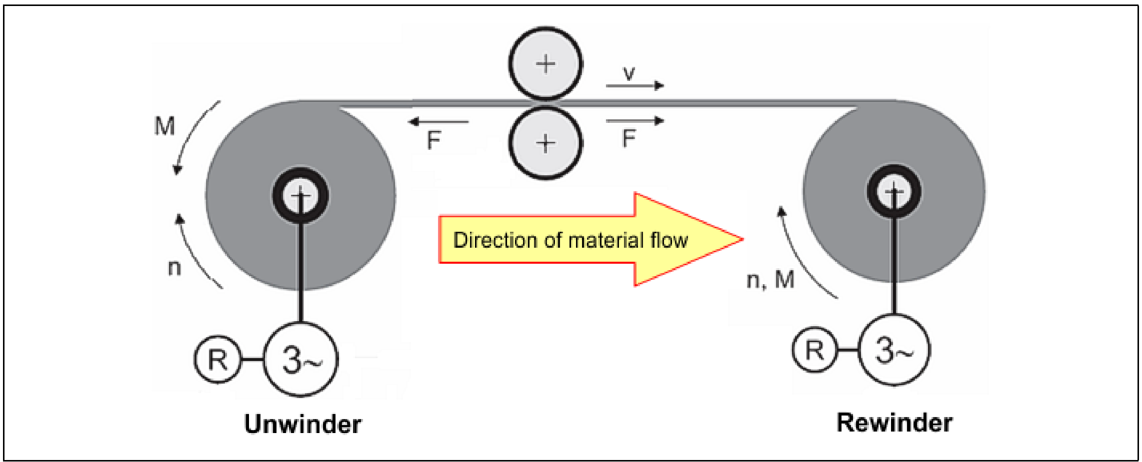
Designator / Data type	Description		Available in version: Base	State	High
AP08: xSetFricCompReelSpeed / BOOL	Enable of the access point AP08: lrSetFricCompReelSpeed			●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP08: lrSetFricCompReelSpeed / LREAL	Selection of the speed for the friction compensation • Unit: revs/s Based on the speed AP08: lrSetFricCompReelSpeed, the corresponding torque for friction compensation is set in the unit [Nm].				
AP09: xSetVelOffset / BOOL	Enable of the access point AP09: lrSetVelOffset		●	●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP09: lrSetVelOffset / LREAL	Cyclic specification of the offset for the velocity of the winder axis with regard to the winding shaft (gearbox output side) • Unit: units/s **The offset value is set immediately and abruptly without ramp generator!**				
AP10: xSetTrqOffset / BOOL	Enable of the access point AP10: lrSetTrqOffset		●	●	●
	TRUE	The access point overwrites the values at the access point in the signal flow.			
AP10: lrSetTrqOffset / LREAL	Cyclic specification of the offset for the torque of the winder axis with regard to the winding shaft (gearbox output side) • Unit: Nm **The offset value is set immediately and abruptly without ramp generator!**				

3.6 Defining the winding direction (winding/unwinding)

In order that the feedforward control values, the disturbance compensation and the correcting signal of the position controller always act in the required direction, the "normal winding direction" must be defined once.

The *xWindingDirection* input serves to set whether the winding drive is to operate as unwinder or rewinder with regard to the normal material flow direction with positive line velocity.

- *xWindingDirection* = TRUE: Unwinder (material is unwound.)
- *xWindingDirection* = FALSE: Rewinder (material is rewound.)



[3-6] Effective direction of speed and torque as a function of the material flow

3.7 Automatic detection of the winding direction

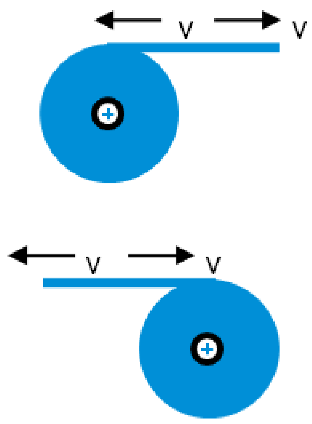
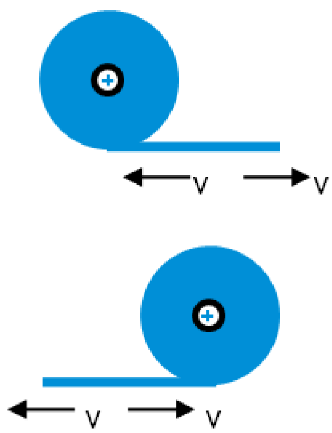
After Defining the winding direction (winding/unwinding) (📖 34), the winding drives can also be operated in the opposite direction with a negative line velocity. No intervention in the signal flow is necessary when the material flow direction is reversed. The current winding direction is provided at the *xUnwind* output.

Exception:

If the line speed is stationary, the winding direction cannot be identified. For this case, set the *xWindingDirectionStandstill* input to TRUE for an unwinding process. If the material is to be rewound at stationary line speed, set the *xWindingDirectionStandstill* input to FALSE.

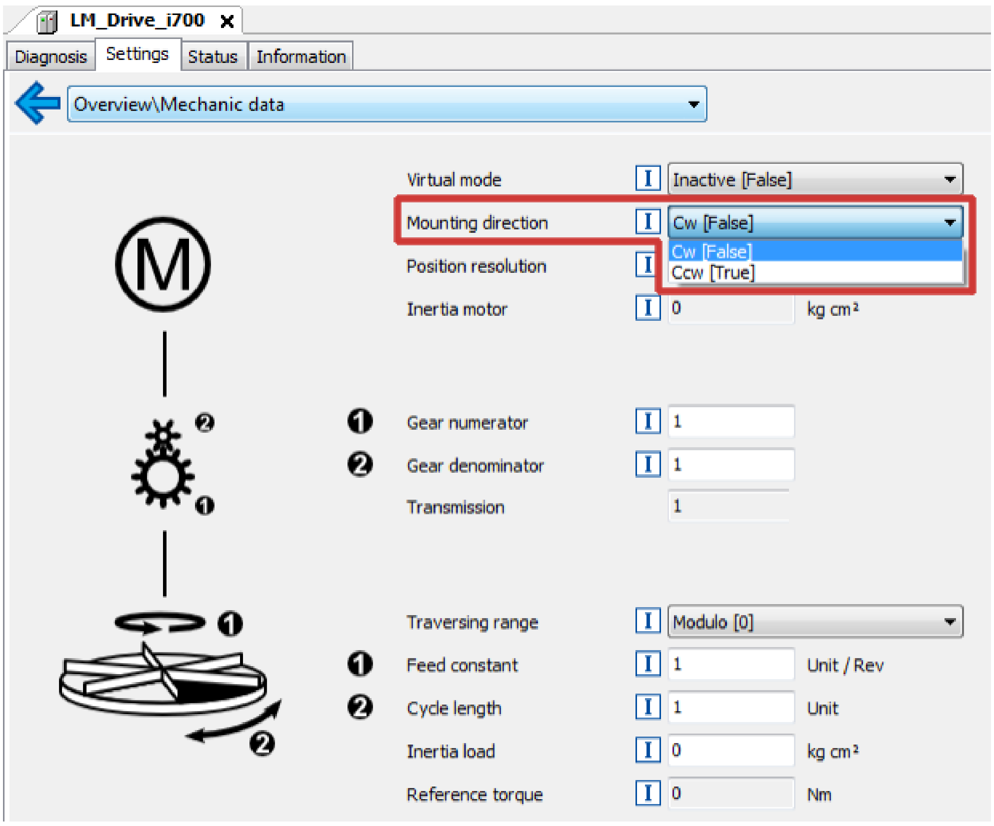
3.8 Defining the material feed to the winder

The *xMaterialFeeding* input serves to define whether the material is fed to the winder from the top or the bottom.

Material from the top (xMaterialFeeding = TRUE)	Material from the bottom (xMaterialFeeding = FALSE)
	

In principle, the rotation direction of the winder shaft is adapted to the material flow through the mounting direction of the motor.

The direction of rotation of the axis can be set in the »PLC Designer« under the **Settings** tab:



3.9 Master value source for diameter calculation

The technology module always operates with the line velocity at the *lrSetLineVel* input.

3.10 Speed feedforward control

The speed feedforward control provides the speed limits for open/closed loop tension-controlled operation as well as the setpoint for the "Follow line speed" set-up.

The setpoint speed for the speed feedforward control is calculated by dividing the line velocity at the *lrSetLineVel* input by the current diameter and the figure π:

Calculation of the setpoint speed for speed feedforward control

$$n_{Set} = \frac{v_{Line}}{d_{act} \cdot \pi}$$

Symbol	Description	Dimension unit
n_{Set}	Setpoint speed for speed feedforward control	revs/s
v_{Line}	Line velocity at the lrSetLineVel input	mm/s
d_{act}	Current diameter	mm

In order that the winder speed complies with the motor setpoint speed and the line velocity signal, the suitable setting for the motor reference speed is mandatory. For this purpose, the calculation and parameterisation are carried out automatically and not by the user.

The scaled winder setpoint speed at the *lrWndSpdRef* output refers to the motor speed required at minimum diameter (d_{min}) in order to reach the reference line velocity at the circumference of the reel.

Check speed feedforward control

- Load the diameter calculator with the minimum diameter (d_{min}):
  Input *lrSetDiam* = 0 (or ≤ d_{min})
  Input *xLoadDiam* = TRUE
- With Synchronisation to the line velocity (📖 44) with the *xSyncLineVel* input = TRUE, the winder axis follows the line setpoint speed in a purely speed-controlled fashion.
  Start the line velocity master and increase the speed up to e.g. 50%. Now the winder should rotate with a speed corresponding to half the reference speed calculated at the *lrWndSpdRef* output.
- Now, the circumferential speed of the winder has to correspond to half the *lrLineVelRef* reference. The current line velocity signal is displayed at the output of the technology module *lrSetLineVelScaledOut* = 0.5 [x 100 %] = 50 %.

If the speed or the direction of rotation is not correct, check the definition of the system data described above.

3.11 Calculation of the diameter

The current diameter is calculated by dividing the line velocity by the winder speed and the figure π:

Calculation of the current diameter		
$d_{act} = \frac{v_{Line}}{n_{Winder} \cdot \pi}$		
Symbol	Description	Dimension unit
d_{act}	Current diameter	mm
v_{Line}	Line velocity	mm/s
n_{Winder}	Winder speed	revs/s

In fact, however, for the calculation integrated speed values are used instead of instantaneous values. This causes a average determination. The number of revolutions causing a recalculation of the diameter is specified via the parameter *lrDiamCalcRegularDist*. The initial value of this parameter is set to 1 winding shaft revolution.

For quick diameter changes of *lrDiamCalcRegularDist*, it can be switched to the fast calculation mode by setting the *xDiamCalcReduced* input = TRUE. The lower calculation distance is set with the *lrDiamCalcReducedDist* parameter. Here, the initial value 1/10 is defined for the winder shaft revolution.

The smaller calculation distance is also activated automatically when a starting diameter is loaded. This state remains until a new diameter has been calculated. The function is required if the real diameter of the reel may deviate significantly from the loaded diameter. Thus, the winder shaft only rotates for a short distance with the "wrong" diameter. After the diameter has been calculated, a suitable value is available again.

Parameters to be set

The parameters for diameter calculation are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrDiamCalcRegularDist : LREAL := 1;
lrDiamCalcReducedDist : LREAL := 0.1;
```

3.12 Holding the diameter

For some operating states of the winder, in which the line velocity does not correspond to the circumferential velocity of the reel, the current diameter cannot be calculated from the line velocity and the motor speed. In this case, the calculation of new values must be prevented and the diameter must be held at the old value.

This is done automatically under the following conditions:

- Line velocity < minimum line velocity (*lrMinLineVel* [mm/s] from the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22)) parameter structure;
- Winder speed < *lrMinLineVel* [mm/s] / (π x d [mm]);
- In the states STOP, ERROR, READY, JOGGING and SYNCLINEVEL.

For the user holding of the diameter, set the *xHoldDiam* input to TRUE.

3.13 Defining the diameter / signal from the diameter sensor

At the start of a winding process, it may be required to define a start diameter or use the signal of a diameter sensor.

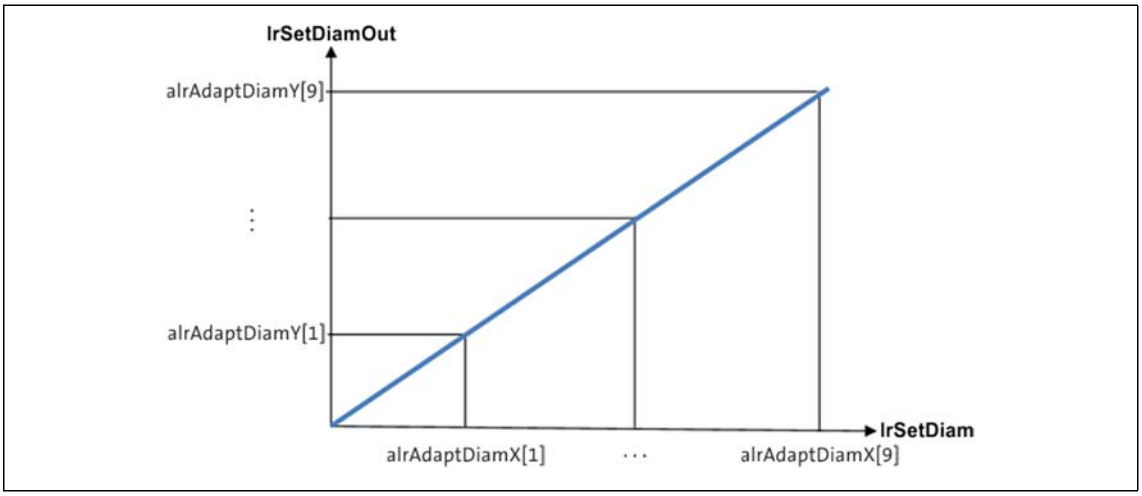
The *lrSetDiam* input serves to define a start diameter that is accepted with the highest priority and loaded cyclically when *xLoadDiam* = TRUE.

An external diameter value, e.g. of an ultrasonic sensor, can also be connected to the *lrSetDiam* input. This analog value can be adapted via a curve function Y = f(x). The curve function is set with nine grid points via the parameters *alrAdaptDiamX[1...9]* and *alrAdaptDiamY[1...9]*. In order that the analog value is used as start diameter, the adapted curve progression is initialised with *alrAdaptDiamY* = *alrAdaptDiamX*. The sensor signal can also be loaded permanently.

Parameters to be set

The parameters for the curve function are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
alrAdaptDiamX : ARRAY[1...9] OF LREAL := [0,100,200,300,400,500,600,700,800];
alrAdaptDiamY : ARRAY[1...9] OF LREAL := [0,100,200,300,400,500,600,700,800]
```



[3-7] Loading of a diameter via a curve function

3.14 Material length counter

The material length counter is activated with the *xEnable* input = TRUE.

The material length is calculated by integrating the line speed at the *lrSetLineVel* input and is shown at the *lrMaterialCounter* output (in millimetres). Depending on the Defining the winding direction (winding/unwinding) (📖 34), the material length is incremented or decremented.

For the analog signal of the line speed, the current value of the material length can be filtered with a PT1 characteristic. The filter time is set with the *rFiltTimeMaterialCounter* parameter (the default setting is '0 ms').

The current value of the material length is stored in the persistent data in the *PersistentVar* structure.

For the initialisation of the material length, an initial material length can be set via the *lrSetMaterialPos* parameter. With a FALSE↗TRUE edge at the *xSetMaterialCounter* input, the initial material length is accepted with the highest priority.

Parameters to be set

The parameters for the material length counter are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
rFiltTimeMaterialCounter : LREAL := 0;
lrSetMaterialPos : REAL := 0;
```

3.15 Sources for the material length counting

The material length can be counted from one of three different sources, and the counting procedure can be carried out in two different ways.

3.15.1 Source: "lrSetLineVel" input

Conditions

- No reference axis is connected to the *MaterialCounterAxis* input.
- *xLineVelDiamCalc* parameter = FALSE

Operating principle

In addition to the position (*lrSetMaterialPos* parameter), the line speed is integrated into the material length counting process at the *lrSetLineVel* input. The resulting value is shown as material length at the *lrMaterialCounter* output and is stored persistently.

With a FALSE↗TRUE edge at the *xSetMaterialCounter* input, the material length is loaded from the *lrSetMaterialPos* parameter, directly setting the material length to the value of *lrSetMaterialPos* at the *lrMaterialCounter* output. The values from the continuing counting process are added to the value of the material length set at the output.

Note!

In the case of a noisy signal, the material counting process is falsified by integration of the line speed. In this process, the material length counter shows a drift, even if the line is stable.

3.15.2 Source: "lrSetLineVelDiamCalc" input

Conditions

- No reference axis is connected to the *MaterialCounterAxis* input.
- *xLineVelDiamCalc* parameter = TRUE

Operating principle

In addition to the position (*lrSetMaterialPos* parameter), the line speed for the Calculation of the diameter (🕮 37) is integrated into the material length counting process at the *lrSetLineVelDiamCalc* input. The resulting value is shown as material length at the *lrMaterialCounter* output and is stored persistently.

With a FALSE↗TRUE edge at the *xSetMaterialCounter* input, the material length is loaded from the *lrSetMaterialPos* parameter, directly setting the material length to the value of *lrSetMaterialPos* at the *lrMaterialCounter* output. The values from the continuing counting process are added to the value of the material length set at the output.

Note!

In the case of a noisy signal, the material counting process is falsified by integration of the line speed. In this process, the material length counter shows a drift, even if the line is stable.

3.15.3 Source: "MaterialCounterAxis" input (reference axis)

Conditions

- A reference axis (modulo axis) is connected to the *MaterialCounterAxis* input.
- The loss-free number of revolutions determined is used as a basis for the material length counting. – This procedure is suitable for noisy signals!

Operating principle

Via the feed constant of the reference axis (modulo axis), the material length is shown at the *lrMaterialCounter* output.

The number of revolutions counted can be read out via the *MP20_liRevCounter* measuring point. The fractional amount of a revolution is shown via the *MP21_lrRevCounterResidual* measuring point. The values of these measuring points are stored persistently.

With a FALSE↗TRUE edge at the *xSetMaterialCounter* input, the material length is loaded from the *lrSetMaterialPos* parameter. In this process, the material length is converted to the number of revolutions and stored via the feed constant of the axis.

The material length from the *lrSetMaterialPos* parameter is shown at the *lrMaterialCounter* output.

Note!

An exact material length counting process can only be carried out with a slip-free measuring wheel. A measuring wheel that is prone to slip on the material causes errors in the material length counting.

3.16 Manual jog (jogging)

For the manual jog of the winder, the *lrJogLineVel* manual jog velocity is used.

When the *xJogLinePos* = TRUE, the line is driven in positive direction and when the *xJogLineNeg* = TRUE, it is driven in negative direction. The line is driven as long as the input remains set to TRUE. The running travel command cannot be replaced by the other jog command.

The parameterisable setpoints *lrJogLineVel, lrJogLineAcc* and *lrJogLineDec* for manual jog refer to the circumferential velocity or line velocity and not to the motor speed.

Parameters to be set

The parameters for the manual jog are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrLineJerk   : LREAL := 10000; // Jerk [mm/s^3]
lrJogLineVel : LREAL := 100;   // Velocity [mm/s]
lrJogLineAcc : LREAL := 100;   // Acceleration [mm/s^2]
lrJogLineDec : LREAL := 10;    // Deceleration [mm/s^2]
```

The parameter values can be changed during operation. They are accepted when resetting the inputs *xJogLinePos* = TRUE or *xJogLineNeg* = TRUE.

3.17 Synchronisation to the line velocity

The winder axis is synchronised to the line velocity when the *xSyncLineVel* input is set to TRUE.

The parameters *lrSyncLineAcc* and *lrSyncLineDec* refer to the circumferential velocity or line velocity and not to the motor speed.

Parameters to be set

The parameters for synchronising to the line velocity are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrLineJerk    : LREAL := 10000; // Jerk [mm/s^3]
lrSyncLineAcc : LREAL := 100;   // Acceleration [mm/s^2]
lrSyncLineDec : LREAL := 100;   // Deceleration [mm/s^2]
```

3.18 Trimming

Note!

Trimming can only be used if the winder axis is synchronised to the line velocity.

▸ Synchronisation to the line velocity (📖 44)

When the *xTrimLinePos* input = TRUE, the line is trimmed in positive direction and when the *xTrimLineNeg* input = TRUE, the line is trimmed in negative direction.

For trimming, the *lrTrimLineVel* trimming velocity is added to the *lrSetLineVel* line velocity. In the trimming process, the overall setpoint can exceed the trimming setpoint maximally by the value of the minimum line velocity.

The parameterisable setpoints *lrTrimLineVel, lrTrimLineAcc* and *lrTrimLineDec* for the positive and negative trimming operation refer to the circumferential velocity or line velocity and not to the motor speed.

Parameters to be set

The parameters for trimming are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrLineJerk    : LREAL := 10000; // Jerk [mm/s^3]
lrTrimLineVel : LREAL := 100;   // Velocity [mm/s]
lrTrimLineAcc : LREAL := 100;   // Acceleration [mm/s^2]
lrTrimLineDec : LREAL := 10;    // Deceleration [mm/s^2]
```

3.19 Tension control via characteristic function (Base version)

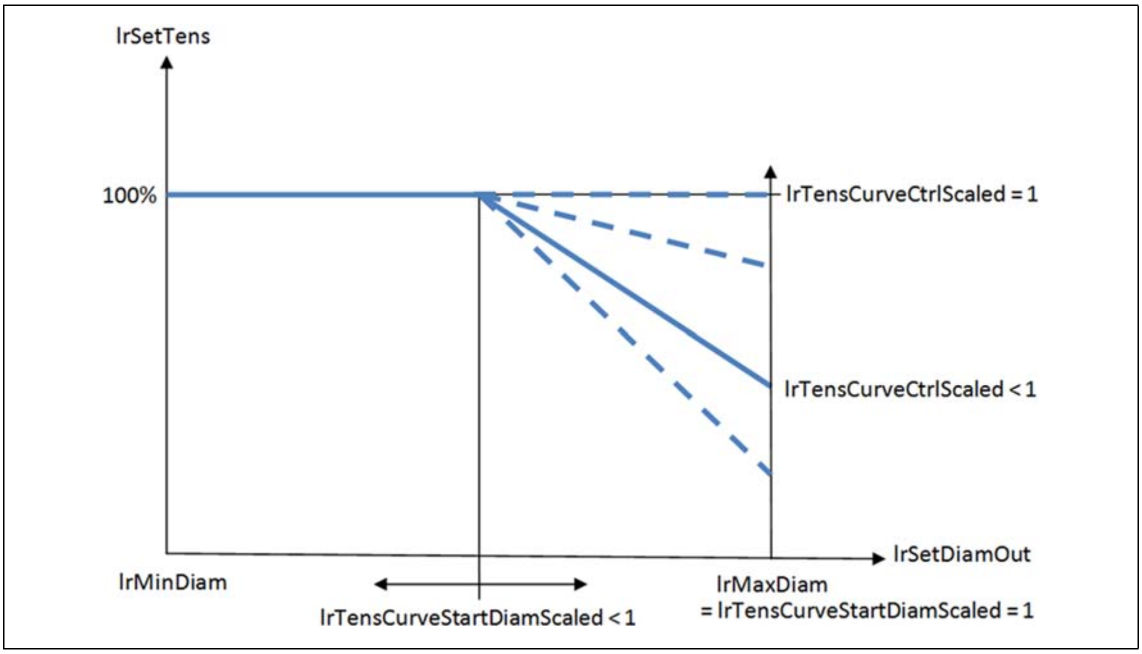
Depending on the surface and the type of winding material, many rewinders require that the tensile force is reduced with an increasing diameter, avoiding that the reel is shifted. This is referred to as the winding characteristic or tensile force characteristic.

Influencing the tensile force in the winder control is a common practice in order to, for example, apply the setpoint adapted to a pneumatic actuator.

In order that the material-dependent characteristic is reached, the real tensile force setpoint from the *lrSetTens* input is evaluated dependent on the diameter via a linear characteristic function.

The characteristic is marked by an initial range with constant evaluation (100 %) and a second range where the tensile force is adapted to the diameter.

The *lrTensCurveStartDiamScaled* parameter serves to define the diameter from which on the tensile force is to be reduced. The *lrTensCurveCtrlScaled* parameter is used to evaluate the tensile force at the maximum diameter.



[3-8] Characteristic for a linear tensile force profile

Parameters to be set

The parameters for the "tensile force via characteristic function" are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (🕮 22) parameter structure.

```
lrTensCurveStartDiamScaled : LREAL := 0; // Initial point of the characteristic line
                                                  [x100 %]
lrTensCurveCtrlScaled : LREAL := 0;      // Gapping of the characteristic line [x100
%]
lrMaxDiam : LREAL := 0;                   // Maximal diameter [mm]
lrMinDiam : LREAL := 0;                   // minimum diameter [mm]
```

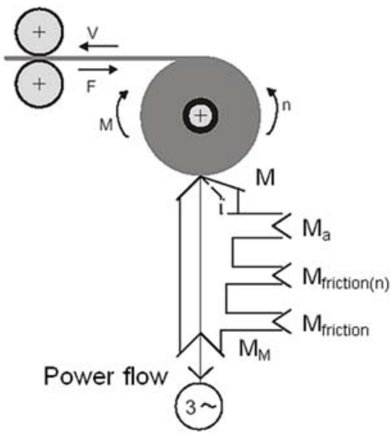
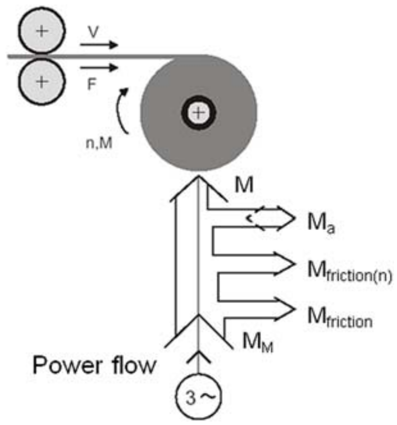
Diameter evaluation / conversion to torque setpoint

The resulting tensile force setpoint from the characteristic evaluation and open loop tension control in a final step must be converted to a motor setpoint torque for the selection via the basic function "Torque follower".

The motor setpoint torque results from the addition of the total tensile force setpoint to the correction values from the friction compensation and the acceleration compensation.

▶ Acceleration compensation (📖 49)

▶ Friction identification and compensation (📖 56)

Unwinder	Rewinder
	

Calculation of the motor setpoint torque

$$M_M = M + M_{friction} + M_{friction(n)} + M_a$$

with

$$M = F \cdot \frac{d}{2}$$

Symbol	Description	Dimension unit
M_M	Motor setpoint torque	Nm
M	Torque at the reel	
$M_{friction}$	Static friction torque	
$M_{friction(n)}$	Speed-dependent friction torque	
M_a	Acceleration torque	
F	Acting force	N
d	Diameter	mm

Ramp generator for the tensile force setpoint

After having activated open loop tension control with the *xTensCtrl* input = TRUE, the tensile force must first reach the setpoint from the *lrSetTens* input. In order that the tensile force is increased in a controlled fashion, the ramp generator for the tensile force setpoint is loaded with the value 'zero' first. The ramp is set using the *lrTensRamp* parameter in unit [N/s]. The standard setting of the ramp is preselected with *lrTensRamp* = 1. If the ramp generator reaches the setpoint, the *xDone* output is set to TRUE.

The parameters for the ramp generator are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrTensRef : LREAL := 2;    // 100% of line tension [N]
lrTensRamp : LREAL := 1;   // Ramp of tension values [N/s]
```

3.20 Acceleration compensation

The acceleration in the line velocity setpoint is a disturbance in the winding process. The torque consumed for acceleration is lacking for the tensile force.

Therefore the acceleration torque has to be calculated and pilot-controlled as an additional torque.

Calculation of the acceleration torque

$$M_a = 2 \cdot \pi \cdot \left(\frac{\partial n}{\partial t}\right) \cdot (J_{const} + J_{var})$$

with

$$J_{var} = (J_{max} - J_{const}) \cdot \left(\frac{d_{act}^4 - d_{min}^4}{d_{max}^4 - d_{min}^4}\right) \cdot B$$

Symbol	Description	Dimension unit
M_a	Acceleration torque	Nm
∂n	(Delta) motor speed δ	revs/s
∂t	(Delta) time	s
J_{const}	Constant moment of inertia	kgm^2
J_{var}	Variable (diameter-dependent) moment of inertia	
J_{max}	Maximum moment of inertia	
d_{act}	Current diameter	mm
d_{min}	Minimum diameter (sleeve diameter)	
d_{max}	Maximum diameter	
B	Material width	mm

In this connection, the change of the speed value (new value – old value) corresponds to the acceleration of the winder. The winder speed is calculated from the line velocity.

In practice, it must be expected that the line velocity signal will not be ideally constant. The parameters *lrAccCmpsGainAcc* and *lrAccCmpsGainDec* can be used to set the resolution of the signal to be differentiated. Additionally, the signal first can be smoothed via a PT1 functionality. The PT1 time constant is set via the *rFiltTimeAccSpd* parameter. For noise suppression, a lagging range can be interconnected via the calculated acceleration torque. The lagging range is set via the *lrAccCmpsDeadBandTrqScaled* parameter in the unit [x 100 %].

The acceleration compensation is enabled when the *xAccCmp* input = TRUE.

In order to generate acceleration, the line velocity must be differentiated. Depending on the resolution and the stability of the signal, it may be required to reduce the sensitivity of the differentiation to prevent that master value fluctuations cause sudden acceleration changes. Different material widths or material densities can be taken into account as percentage via the *lrMInertiaAdapt* input.

Defining moments of inertia

Note!

The defined mass inertia has to refer to the winder shaft and not to the motor shaft.

The inertia (J) of motor shaft to the winder shaft can be calculated using the following equation:

Calculation of the inertia (J) from the motor shaft to the winder shaft

$$J_{Winder} = i^2 \cdot J_{Motor}$$

with

$$i = \frac{n_{Motor}}{n_{Winder}}$$

Symbol	Description	Dimension unit
J_{Winder}	Moment of inertia of the winder shaft	$kgcm^2$
J_{Motor}	Moment of inertia of the motor shaft	$kgcm^2$
i	Gearbox factor	
n_{Motor}	Motor speed	revs/s
n_{Winder}	Winder speed	revs/s

The moment of inertia consists of a constant part and a diameter-dependent part. The constant part is mainly determined by the moment of the inertia of the motor (code C00273/1 for Lenze motors). The variable part is determined by the diameter as well as the maximum and constant moment of inertia in the technology module.

The constant moment of inertia is set using the *lrConstMInertia* parameter.

The maximum moment of inertia (full reel) is set via the *lrMaxMInertia* parameter.

Parameters to be set

The parameters for the acceleration compensation are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
rFiltTimeAccSpd : REAL := 0.005;  // Filtertime ActReelSpeed during AccComp [s]
lrAccCmpsDeadBandTrqScaled : LREAL := 0.10;  // Dead-band of winder torque [Nm]
lrAccCmpsGainAcc : LREAL := 1.05;  // [x100 %]
lrAccCmpsGainDec : LREAL := 0.95;  // [x100 %]
lrConstMInertia : LREAL := 9;  // Constant MInertia J_min [kgcm^2]
lrMaxMInertia : LREAL := 50;   // Maximum MInertia J_max [kgcm^2]
```

3.21 Web break monitoring

For the diameter calculation, the technology module provides a web break monitoring function.

In case of a web break, the calculated diameter changes contrary to the winding direction (unwinding or rewinding).

The monitoring function is activated with the *xWebBreakMonit* input = TRUE, making a diameter change opposed to the winding direction only permissible within the window set in the *lrWebBreakWindow* parameter.

Rewinding or unwinding is automatically detected via the sign of the line velocity and the winding direction set via the *xWindingDirection* input.

Note!

Web break monitoring may only be activated when the calculated diameter corresponds to the real diameter.

If web break monitoring is activated (*xWebBreakMonit* = TRUE), a diameter change contrary to the winding direction specified via the *xUnwind* output.

After loading a starting diameter which deviates considerably from the real diameter contrary to the winding direction, this may lead to an unintentional triggering of the monitoring function. If, for instance, a starting diameter of 50% is loaded for a rewinder and the real diameter only is 45%, then a change of the diameter value to the real 45% is prevented when web break monitoring is activated.

Parameters to be set

The parameters for web break monitoring are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrWebBreakWindow : LREAL :=  0.1; // Window for web break 0..1 [x100 %]
```

3.22 Persistent variables

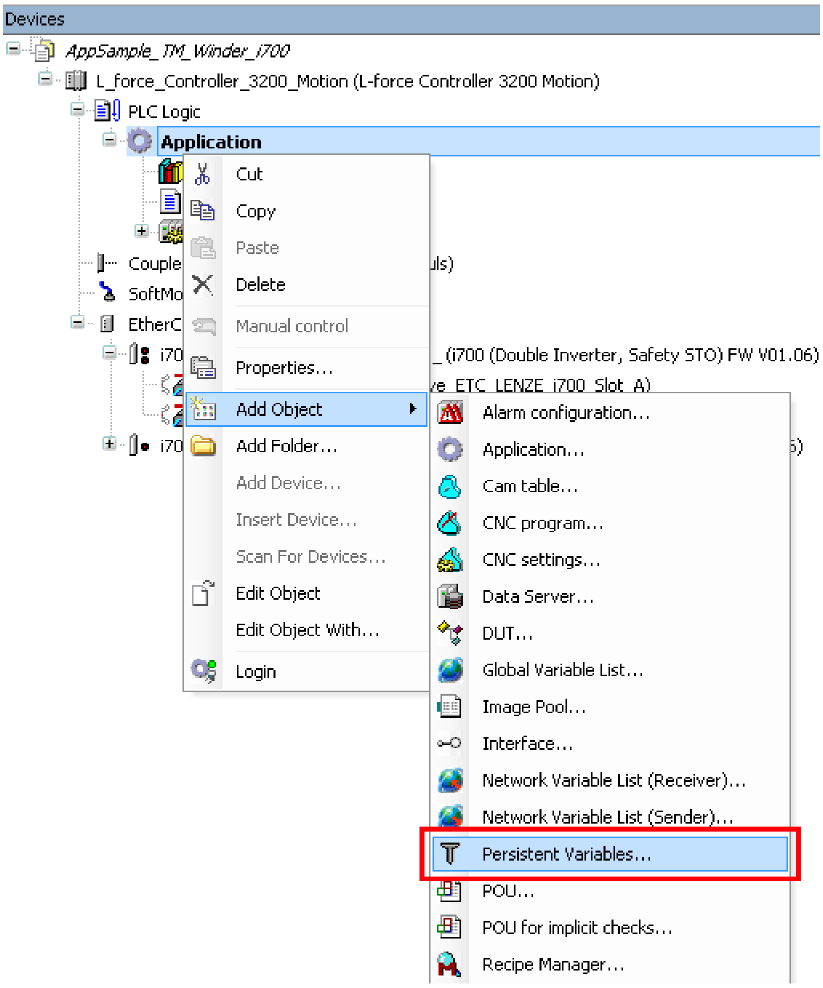
The technology module provides the possibility of persistently saving the parameters determined such as the winding diameter. For this purpose, the following setting has to be made in the »PLC Designer« for the technology module.

How to create persistent variables in the »PLC Designer«:

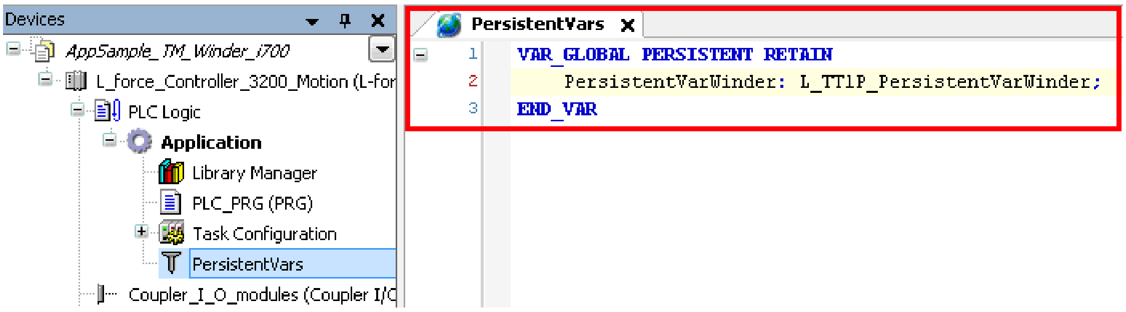
Note!

This procedure does not apply to the Application Template, since it already provides structures for persistent data of the machine modules.

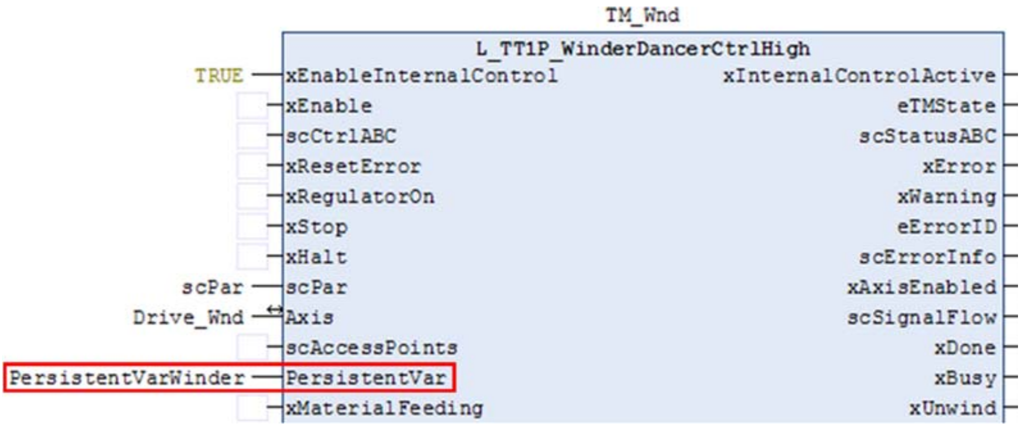
1. Add the global list for managing persistent variables in the context menu to **Application** with the **Add object → Persistent variables...** command.



2. Instance the reference of the persistent variables "L_TT1P_PersistentVarWinder" in the global structure of the persistent variables.



3. Interconnect the instance of the persistent variables with the *PersistentVar* input.



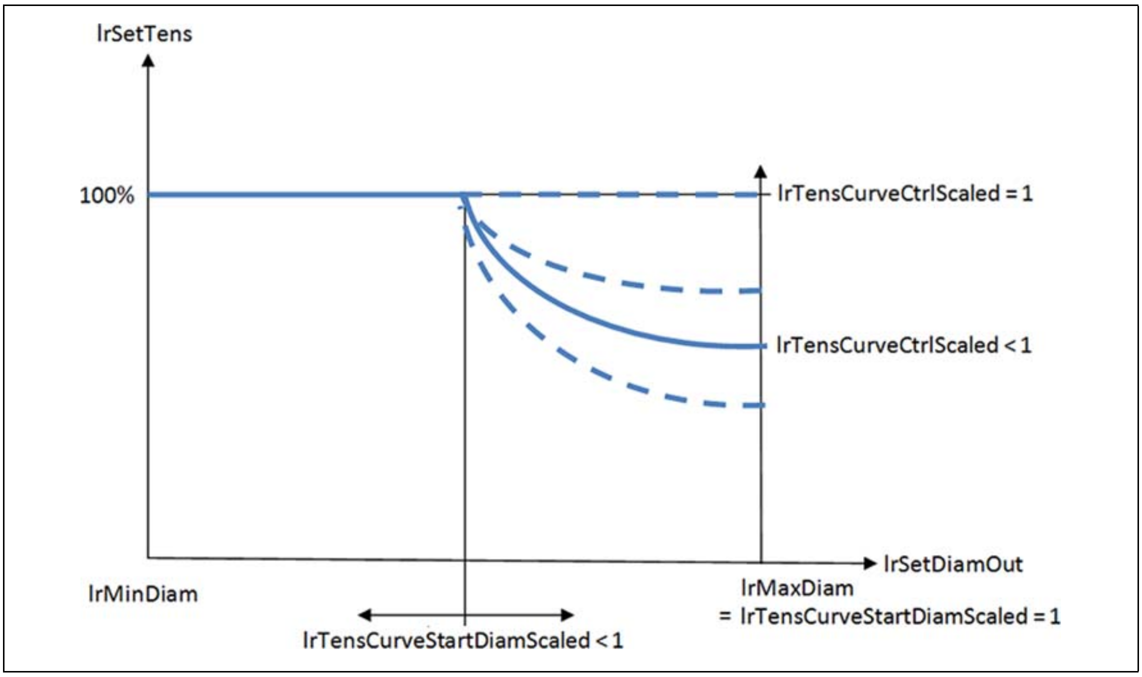
3.23 Tension control via characteristic function/winding characteristic

The characteristic function for tensile control is extended in the State version. In order that the material-dependent characteristic is reached, the tensile force setpoint from the *lrSetTens* input is evaluated as a function of the diameter via a characteristic function.

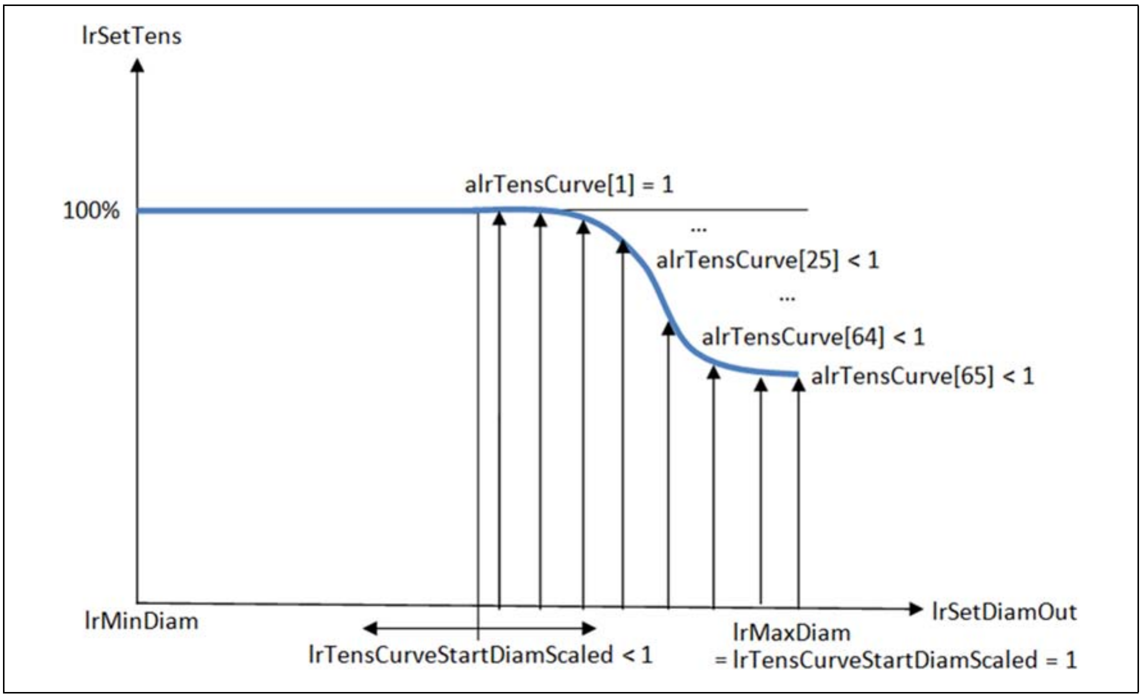
For the adaptation, different methods are available:

- Characteristic for a linear tensile force profile (*dwSelectTensCurve* = 0)
- Characteristic for a linear torque profile (*dwSelectTensCurve* = 1)
- User-definable characteristic with 64 grid points (*dwSelectTensCurve* = 2)

The characteristic has an initial area with constant evaluation (100 %) and a second area in which the tensile force is adapted to the diameter. The *lrTensCurveStartDiamScaled* parameter is used to specify the diameter from which on the tensile force will be reduced. The *lrTensCurveCtrlScaled* parameter serves to specify the percentage of the tensile force at the maximum diameter.



[3-9] Characteristic for a linear torque profile



[3-10] Characteristic with user-definable grid points

Parameters to be set

The parameters for the characteristic function are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
dwSelectTensCurve : DWORD := 0;              // 0..3
alrTensCurve : ARRAY[1..65] OF LREAL;        // Tension curve
lrTensCurveCtrlScaled : LREAL:= 0;           //  0 .. 1  [x100 %]
lrTensCurveStartDiamScaled : LREAL := 0;     // [x100 %]
lrMaxDiam : LREAL := 0;                      // Maximum diameter [mm]
lrMinDiam : LREAL := 0;                      // Minimum diameter [mm]
```

3.24 Friction identification and compensation

Note!

If the winder is controlled via an oil-filled gearbox, the friction very much depends on the temperature of the oil. The friction identification should be carried out using gearbox oil that is as warm as possible. For this it is sufficient to previously actuate the motor in a speed-controlled manner for some minutes.

To identify the friction, the winder shaft must be empty (without containing material).

Use the setting *lrIdentFricMaxSpdScaled* parameter = 1 to define the maximum motor speed (100 %) relating to the maximum winder speed at the *lrWndSpdRef* output, which may be reached during the identification run. (Usually the motor will also reach the *lrWndSpdRef* speed with the minimum diameter and at maximum line speed.)

Adapt the acceleration and deceleration time for the identification run to the basic conditions given: The maximally attainable torque must suffice to accelerate the mass inertia at the ramp to the *lrIdentFricMaxSpdScaled* parameter.

Optionally up to 4 friction characteristics can be recorded and stored. A change-over between different friction characteristics is for instance required when variable speed drives are used.

The friction characteristic is detected with a rising edge (FALSE↗TRUE) at the *xExecuteIdentFric* input.

Termination of the identification

The identification is completed as soon as the motor has reached standstill again, no errors have been reported and *xDone* has been set to TRUE.

The detected torque setpoints can be checked for plausibility and can be corrected if required.

The friction compensation is activated with *xFricCmps* = TRUE.

Parameters to be set

The parameters for friction identification and compensation are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrFricCurveStartTrq : LREAL := 0;      // Start friction for linear compensation [Nm]
lrFricCurveEndTrq : LREAL := 0;        // End friction for linear compensation [Nm]
rFiltTimeFricSetSpd : REAL := 0.01;     // Filter time SetReelSpeed 1 = 1[s]
lrIdentFricMaxSpdScaled : LREAL := 0.9; // Max. speed for ident friction [x100%]
lrIdentFricAccDec : LREAL := 1;         // Acceleration/Deceleration for ident
                                        // friction [rev/s^2]
rFiltTimeIdentFricSpd : REAL := 0.0;    // Filter time ident ActReelSpeed 1 = 1[s]
rFiltTimeIdentFricTrq : REAL := 0.05;   // Filter time 1 = 1[s]
dwSelectFricCurve : DWORD:=1;           // Friction curve 0..4
```

3.25 PI controller for tension control

Activation of the tension control with the *xTensCtrl* input = TRUE initiates closed-loop control to an actual tensile force value.

Use the *lrTensCtrlInfluence* input to define the influence of the PI controller on the motor control (standard setting: 0 %).

After having activated open loop tension control, the tensile force must first reach the setpoint from the *lrSetTens* input. In order that the tensile force is increased in a controlled fashion, the ramp generator for the tensile force setpoint is loaded with the current actual tensile force value first. The ramp is set using the *lrTensRamp* parameter in the unit [N/s]. The standard setting of the ramp is preselected with *lrTensRamp* = 1. If the ramp generator reaches the setpoint, the *xDone* output is set to TRUE.

It is not possible to filter and/or adapt the actual tensile force value. A signal (actual tensile force value) at the *lrActTensIn* input is expected.

The I component of the PI controller can be set with the *lrTensCtrlResetTime* parameter. In the default setting, *lrTensCtrlResetTime* is set to 0 (deactivated).

The controller gain is set with the *lrTensCtrlGain* parameter.

Parameters to be set

The parameters for the PI controller and for tension control are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrTensRamp : LREAL := 1;            // Ramp of lrSetTens [N/s]
lrTensCtrlGain : LREAL := 1;        // Gain of tension control
lrTensCtrlResetTime : LREAL := 0;   // Reset time of tension control
```

3.26 Identification of the moments of inertia

In order to compensate the acceleration torque, a parameterisation or identification of the constant moment of inertia (motor + gearbox + winder shaft) and the maximum moment of inertia (with full reel) is required.

Identification of the constant moment of inertia

The winder shaft is empty (no material).

The *lrIdentMInertiaMaxSpdScaled* parameter serves to define the maximum motor speed in [x 100 %] regarding the maximally reachable winder speed *lrWndSpdRef*. Typically, speeds between 50 ... 60 % are sufficient.

The *lrIdentMInertiaMaxTrqScaled* parameter serves to define the acceleration torque. This value always has to be higher than the maximally occurring friction – recommendation: 25 %.

The mass moment of inertia is detected with a rising edge (FALSE↗TRUE) at the *xExecuteIdentMInertia* input. The *lrIdentMInertia* output displays the detected moment of inertia.

Identification of the maximum moment of inertia

The winder is loaded with the maximum possible reel (maximum diameter and maximum width).

The maximum motor speed *lrIdentMInertiaMaxSpdScaled* has to be parameterised so that the maximally permissible circumferential velocity of the winder will not be exceeded (e.g. *lrIdentMInertiaMaxSpdScaled* = 10 % at d_{max}/d_{min} = 10) – recommended: 25 %.

The mass moment of inertia is detected with a rising edge (FALSE↗TRUE) at the *xExecuteIdentMInertia* input.

Termination of the identification

The identification is completed when the motor is at standstill again no errors have been reported. The identified inertia of the winder shaft (J_{Winder}) is displayed at the *lrIdentMInertia* output and has to be checked for plausibility.

The identification of both types of inertia should be executed several times. In doing so, the filter time for the *rFiltTimeIdentMInertiaSpd* speed can be altered.

Note!

A significant non-linear friction in the system has a negative impact on the calculation of the moment of inertia in the technology module.

Transfer the values of the identified moments of inertia into the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

Parameters to be set

The parameters for the identification of the moments of inertia are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
rFiltTimeIdentMInertiaSpd : REAL := 0.01;    // Filter time ActReelSpeed during ident
                                              // MInertia 1 = 1[s]
rFiltTimeIdentMInertiaTrq : REAL := 0.005;   // Filter time ActTorque during ident
                                              // MInertia 1 = 1[s]
lrIdentMInertiaMaxSpdScaled : LREAL := 0.2;  // Max ident speed [x 100%]
lrIdentMInertiaMaxTrqScaled : LREAL := 0.2;  // Max ident torque [x 100%]
```

Calculation of the maximum moment of inertia

If no reel is available, the maximum moment of inertia can be calculated as follows:

Calculation of the maximum moment of inertia

The density of the winding material is known:

$$J_{MaxWinder} = i^2 \cdot J_{Motor} + \left(\frac{\pi}{32 \cdot 10^8}\right) \cdot B \cdot \rho \cdot (d_{max}{}^4 - d_{min}{}^4)$$

The mass of the winding material is known:

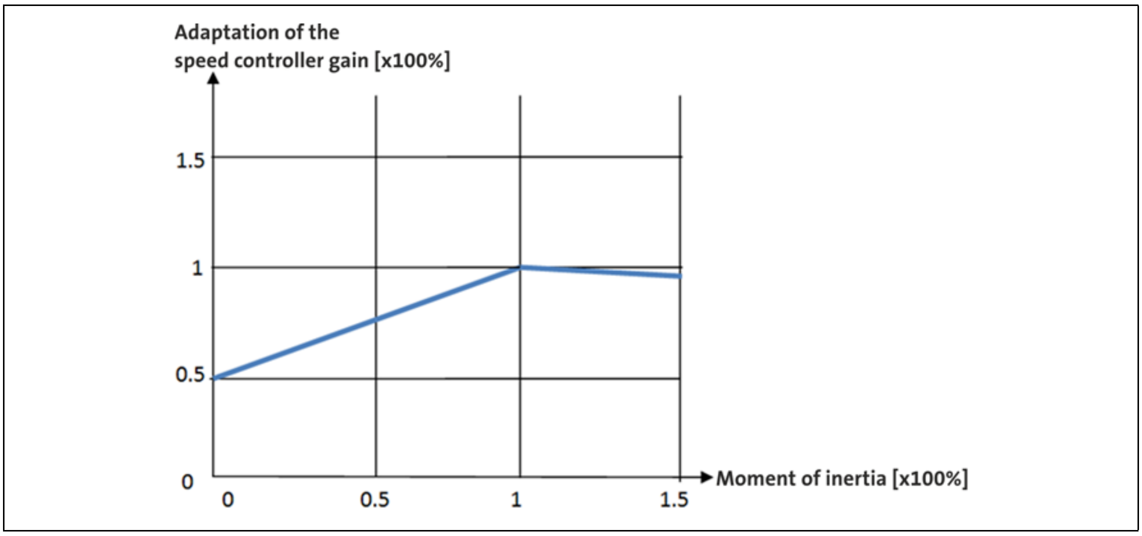
$$J_{MaxWinder} = i^2 \cdot J_{Motor} + \frac{m \cdot d_{max}{}^2}{800}$$

Symbol	Description	Dimension unit
$J_{MaxWinder}$	Maximum moment of inertia of the winder shaft	$kgcm^2$
J_{Motor}	Moment of inertia of the motor shaft	$kgcm^2$
i	Gearbox factor	
B	Material width	mm
r	Material density	kg/dm^3
d_{max}	Maximum diameter	mm
d_{min}	Minimum diameter (sleeve diameter)	mm
m	Mass	kg

3.27 Adaptation of the speed controller gain

Looking at the motor and the reel as a rigid one-mass system, the optimum gain of the speed controller is directly proportional to the moment of inertia.

Since the moment of inertia usually changes significantly during the winding process, for a good control response it may be necessary to continuously adapt the gain of the speed controller to the moment of inertia.



[3-11] Preset characteristic function for adapting the speed controller gain

The standard setting of this characteristic function is based on the following functions:

- Lower limitation of the adaptation: 50 %
- Upper limitation of the adaptation: 100 %
- Linear increase of the gain up to 100 % of the moment of inertia

Parameters to be set

The parameters for adapting the speed controller gain are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
alrSpdCtrlGainAdaptX : ARRAY[1..9] OF LREAL :=
                       [0.0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6];
alrSpdCtrlGainAdaptY : ARRAY[1..9] OF LREAL :=
                       [0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 0.98, 0.95, 0.95];
```

3.28 System deviation within the reduced sensitivity

A reduced controller dynamics at low system deviations serves to have a positive effect on the damping behaviour of the control loop.

The system deviation results from the difference of the values from input *lrSetDancerPosScaled* and output *lrActDancerPosScaled*.

The *lrReducedGainWindow* parameter serves to set a tolerance zone in which the system deviation is transmitted to the controller with a lower gain. The tolerance zone is set above and below around the setpoint (*lrSetDancerPosScaled* input).

The *lrReducedGain* parameter serves to set the value to which the gain is to be reduced in the defined tolerance zone. This means, in the tolerance zone, the reduced gain is active (*lrReducedGain*).

Parameters to be set

The parameters for the characteristic function are located in the L_TT1P_scPar_WinderTensionCtrl[Base/State/High] (📖 22) parameter structure.

```
lrReducedGain : LREAL := 0;
lrReducedGainWindow : LREAL := 0;
```

3.29 CPU utilisation (example Controller 3231 C)

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

Versions	Interconnection of the technology module	CPU utilisation	
		Average	Maximum peak

Base	xEnable := TRUE;
	xRegulatorOn := TRUE;
	xSyncLineVel := TRUE;
		75 µs	110 µs

State	xEnable := TRUE;
	xRegulatorOn := TRUE;
	xSyncLineVel := TRUE;
		85 µs	119 µs

High	xEnable := TRUE;
	xRegulatorOn := TRUE;
	xSyncLineVel := TRUE;
		95 µs	122 µs

Index

S

T

V

W

FEEDBACK

Your opinion is important to us

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

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

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

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

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