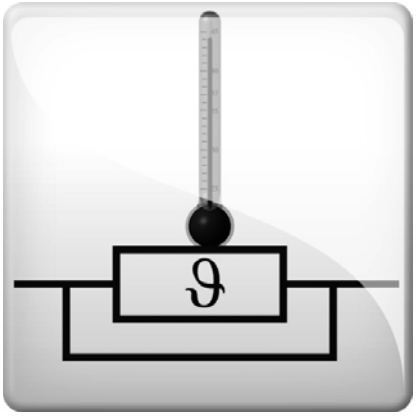
FAST Application Software



Technology module

Temperature Control _______________________

Reference Manual EN

13531746

Lenze

1 About this documentation

This documentation ...

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

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

More technical documentation for Lenze components

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

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

Symbols:

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

Tip!

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

www.lenze.com

Target group

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

1.1 Document history

Version			Description
2.2	05/2017	TD17	Content structure has been changed. New: ▸ Outputs (📖 14) xDiffPositiveOut and xDiffNegativeOut ▸ Parameters (📖 16) lrDiffCtrlActuatingTime ▸ Signal flow diagram: Correcting variable conditioning (📖 22) ▸ Minimum external wiring for the use of the DIFF module (📖 28)
2.1	04/2016	TD17	General revisions
2.0	10/2015	TD17	• General editorial revision • Modularisation of the contents for the »PLC Designer« online help
1.0	06/2015	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!

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).

Stop!

Temperature monitoring is deactivated

In the case of the Control with constant correcting variable (📖 35), all temperature monitoring functions are deactivated.

Possible consequences

Damages to the heating system or the medium to be heated

Protective measures

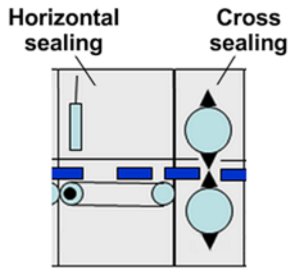
Due to previous process monitoring or know-how, select the correcting variable in such a way that the desired actual temperature is set by approximation.

3 "Temperature Control" functional description

The "Temperature Control" technology module can be used in applications in which the primary task is the control of the temperature of a (partial) system that is equipped with a heating element and a thermal sensor.

Typical application cases in packaging machines are, for instance, temperature control systems for sealing knifes or sealing bars which seal and/or separate packaging material by means of the effect of heat.

Another typical application case are extruders where material such as synthetic granules is melted and compressed.



[3-1] Sealing knife and sealing bar in packaging machines

3.1 Overview of the functions

In addition to the basic **Stop** function, the technology module offers the following functionalities:

- Parameter identification (📖 30)
  In this operating mode, the optimal values for the controller parameters of the heat section are identified.
- Temperature control (📖 33)
  In this operating mode, the heat section is controlled with the identified controller parameters.
  Before temperature control, the parameter identification has to be carried out once.
- Control with constant correcting variable (📖 35)
  In this operating mode, the heat section is controlled with a constant correcting variable defined in the parameters.
  This is required for maintaining an emergency operation when the thermal sensor or its evaluation unit is defective.
  All temperature monitoring modes are deactivated in this operating mode!

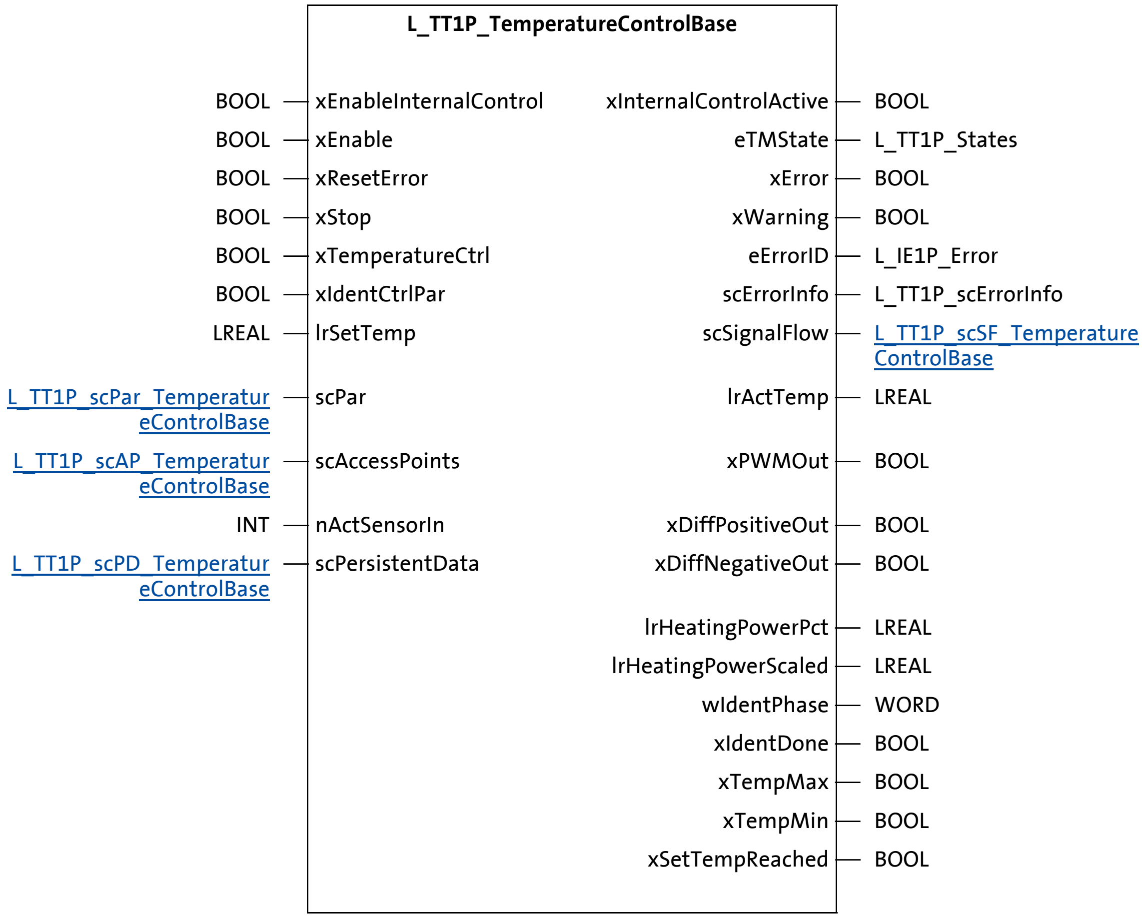
FAST technology modules reference manual

Here you can find detailed information on the **Stop function**.

3.2 L_TT1P_TemperatureControlBase function block

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

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



3.2.1 Inputs and outputs

Designator Data type	Description
scPersistentData L_TT1P_scPD_TemperatureControlBase	Persistent data (📖 19) contain the controller characteristic values calculated in the course of the parameter identification process (xIdentCtrlPar input = TRUE).

3.2.2 Inputs

Designator Data type	Description	
xEnableInternalControl BOOL	TRUE	In the visualisation, the internal control of the axis can be selected via the "Internal Control" axis.
xEnable BOOL	Execution of the function block	
	TRUE	The function block is executed.
	FALSE	The function block is not executed.
xResetError BOOL	TRUE	Reset axis error or software error.
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 = TRUE. • The input is also active with "Internal Control".
xTemperatureCtrl BOOL	TRUE	Activate the "Temperature control (33)" or "Control with constant correcting variable (35)" operating mode. The operating mode is selected via the eModeTempCtrl parameter of the L_TT1P_scPar_TemperatureControlBase (16) parameter structure
xIdentCtrlPar BOOL	TRUE	Activate the "Parameter identification (30)" operating mode. The values to be used are selected via the xIdentAuto parameter of the L_TT1P_scPar_TemperatureControlBase parameter structure (16)
lrSetTemp BOOL	Temperature setpoint for the "Temperature control" operating mode • Unit: °C or °F Depending on the temperature unit set in the xTempUnitIsCelsius parameter of the L_TT1P_scPar_TemperatureControlBase (16) parameter structure.	
scPar L_TT1P_scPar_TemperatureControlBase	The parameter structure contains the parameters of the technology module.	
scAccessPoints L_TT1P_scAP_TemperatureControlBase	Structure of the access points	
nActSensorIn INT	Input for coupling the thermal sensor (usually PT100 with interface connection via EPM module)	

3.2.3 Outputs

Designator Data type	Description	
xInternalControlActive BOOL	TRUE	The internal control of the axis is activated via the visualisation. (xEnableInternalControl input = TRUE)
eTMState L_TT1P_States	Current state of the technology module ▸ State machine (📖 20)	
	210	CONTROL
	211	REGULATION
	212	IDENTIFICATION
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	
scSignalFlow L_TT1P_scSF_TemperatureControlBase	Structure of the signal flow ▸ Signal flow diagrams (📖 21)	
lrActTemp LREAL	Actual temperature value • Unit: °C or °F Depending on the temperature unit set in the xTempUnitIsCelsius parameter of the L_TT1P_scPar_TemperatureControlBase (📖 16) parameter structure.	
xPWMOut BOOL	TRUE	Output signal of the internal PWM module for triggering a (solid state) relay
xDiffPositiveOut BOOL	Positive output signal for actuators (mixing valves) with a differential control	
	TRUE	Open mixing valve / increase heating power
xDiffPositiveOut BOOL	Negative output signal for actuators (mixing valves) with a differential control	
	TRUE	Close mixing valve / reduce heating power
lrHeatingPowerPct LREAL	Current value for the heating power • Unit: %	
lrHeatingPowerScaled LREAL	Current value for the heating power • Value range: 0.0 ... 1.0 (1.0 = 100 %)	
wIdentPhase LREAL	Display of the active phase during the parameter identification Possible values:	
	0	Parameter identification is inactive
	1	Parameter identification phase 1 active
	2	Parameter identification phase 2 active
	3	Parameter identification phase 3 active
	4	Parameter identification phase 4 active
	5	Parameter identification is completed
xIdentDone BOOL	TRUE	The parameter identification has been completed successfully.
xTempMax BOOL	TRUE	Warning signal "Overtemperature" The current actual temperature exceeds the setpoint temperature by more than the value set in the lrTempMaxWindow parameter (L_TT1P_scPar_TemperatureControlBase (📖 16) parameter structure).

Designator Data type	Description	
xTempMin BOOL	TRUE	Warning signal "Undertemperature" The current actual temperature falls below the setpoint temperature by more than the value set in the lrTempMinWindow parameter (L_TT1P_scPar_TemperatureControlBase (📖 16) parameter structure).
xSetTempReached BOOL	TRUE	Information signal "Setpoint temperature reached" • The temperature setpoint (lrSetTemp input) has been reached. • The deviation between the setpoint temperature and the actual temperature does not exceed the tolerance set in the lrSetTempReachedTolerance parameter (L_TT1P_scPar_TemperatureControlBase (📖 16) parameter structure).

3.2.4 Parameters

L_TT1P_scPar_TemperatureControlBase

The **L_TT1P_scPar_TemperatureControlBase** structure contains the parameters of the technology module.

Designator / Data type	Description
lrSensorValueGain LREAL	Gain factor for the sensor signal Set this value so that the sensor used provides a correct mapping to the temperature unit °C. • Initial value: 0.1
lrSensorValueOffset LREAL	Offset factor for the scaled sensor signal Set this value so that the sensor used and the set gain factor (lrSensorValueGain) provide a correct zero point mapping (absolute position) to the temperature unit °C. • Initial value: 0.0
xTempUnitIsCelsius BOOL	Unit for all temperature values (setpoint/actual value/parameter setting) • Initial value: TRUE TRUE: Celsius [°C] FALSE: Fahrenheit [°F]
lrTempMaxCriticalAbsolute LREAL	Absolute temperature limit for the system If this value is reached, the correcting variable is set to 0 % and an error is generated. • Unit: °C or °F • Initial value: 120.0
lrSetTempReachedTolerance LREAL	Permitted deviation (tolerance) between setpoint and actual temperature The xSetTempReached output is set to TRUE if the deviation between setpoint and actual temperature does exceed the value set here. • Unit: °C or °F • Initial value: 1.0
lrTempMinWindow LREAL	Difference in temperature for tripping the "undertemperature" warning signal The xTempMin output is set to TRUE if the current actual temperature falls below the setpoint temperature by more than the value set here. • Unit: °C or °F • Initial value: 5.0
lrTempMinHysteresis LREAL	Hysteresis for the "undertemperature" warning signal (xTempMin output = TRUE). • Unit: °C or °F • Initial value: 1.0
lrTempMaxWindow LREAL	Difference in temperature for tripping the "overtemperature" warning signal The xTempMax output is set to TRUE if the current actual temperature exceeds the setpoint temperature by more than the value set here. • Unit: °C or °F • Initial value: 5.0
lrTempMaxHysteresis LREAL	Hysteresis for the"overtemperature" warning signal (xTempMax output = TRUE). • Unit: °C or °F • Initial value: 1.0
lrSensorMonitoringValueMin LREAL	If the sensor provides a temperature value smaller than this parameter value, this is recognised as a probe error and an error message is displayed. • Unit: °C or °F • Initial value: -1000.0 (with this value, monitoring is deactivated.)
lrSensorMonitoringValueMax LREAL	If the sensor provides a temperature value greater than this parameter value, this is recognised as a probe error and an error message is displayed. • Unit: °C or °F • Initial value: 1000.0 (with this value, monitoring is deactivated.)
lrSensorMonitoringChangeMax LREAL	If the temperature change within one cycle is greater than the value set here, this is recognised as a probe error and an error message is displayed. • Unit: °C/s / °F/s • Initial value: 0.0 (with this value, monitoring is deactivated.)

Designator Data type	Description
xCtrlIntegralClamping BOOL	TRUE: As long as the correcting variable for the thermostat is limited, the added-up I component is held (frozen). • Initial value: TRUE
lrControlValueMinPct LREAL	Minimum value for the correcting variable limitation of the thermostat • Unit: % • Initial value: 0.0
lrControlValueMaxPct LREAL	Maximum value for the correcting variable limitation of the thermostat • Unit: % • Initial value: 100.0
lrPWMPeriod LREAL	Period of the PWM signal generated in the technology module. • Unit: s • Initial value: 1.0
lrDiffCtrlActuatingTime LREAL	Control time of an actuator connected (mixing valve) The control time is the time period required by the mixing valve from full closing to full opening with permanent activation. • Unit: s • Initial value: 60.0
xIdentAuto BOOL	Selection of the values to be used in the "parameter identification" operating mode (xIdentCtrlPar input = TRUE) • Initial value: TRUE TRUE: For the identification process, standard values and automatically calculated values are used. FALSE: For the identification process, the values stored in the following parameters are used.
lrIdentStep1TempChange Max LREAL	Max. permissible temperature change in order that reaching the stationary (ambience) temperature is detected in phase 1 of the parameter identification. If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: °C/min. / °F/min. • Initial value: 0.5
lrIdentStep2ControlStepPct LREAL	Initial correcting variable injection in phase 2 of the parameter identification If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: % • Initial value: 4.0
lrIdentStep2TempChange Max LREAL	Maximally permissible temperature change in order that reaching the stationary (ambience) temperature is detected in phase 2 of the parameter identification and phase 2 can be terminated. If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: °C/min. / °F/min. • Initial value: 0.5
lrIdentStep2TempDiffMin LREAL	Minimum value of the temperature increase to be reached in phase 2 of the parameter identification before switching over to phase 3. If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: °C or °F • Initial value: 10.0
lrIdentStep2WaitTimeMin LREAL	If the stationary temperature (lrIdent-Step2TempChangeMax parameter) has been reached after this time has elapsed but not the minimum temperature increase from the lrIdentStep2TempDiffMin parameter, the correcting variable will be doubled and the time will be restarted. If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: min. • Initial value: 10.0
lrIdentStep2ControlValue MaxPct LREAL	If the correcting variable exceeds this value without the minimum temperature increase set in the lrIdentStep2TempDiffMin parameter being reached, the parameter identification ends with an error. If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: % • Initial value: 50.0

Designator Data type	Description
lrIdentStep3ControlStepPct LREAL	Value of the correcting variable step in phase 3 of the parameter identification for accepting the step response of the heat section If the value '-1' is set, the correcting variable step for phase 3 is automatically calculated from the heat section of phase 2 If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: % • Initial value: -1.0
lrIdentStep3TempDiffMin LREAL	Minimum value of the temperature increase to be reached in phase 3 of the parameter identification in order that the detected parameters (values) are regarded as valid. If the xIdentAuto parameter is set to TRUE, this parameter is not used. • Unit: °C or °F • Initial value: 20.0
eModeTempCtrl ENUM	Operating mode for the temperature control (xTemperatureCtrl input = TRUE) • Initial value: REGULATION
	CONTROL • A constant correcting variable for the heat section is defined (lrControlValueFixedPct parameter). • All temperature monitoring modes are deactivated.
	REGULATION • The heat section is controlled and operated with the thermostat.
lrControlValueFixedPct LREAL	Constant correcting variable for the "CONTROL" operating mode. • Unit: % • Initial value: 10.0

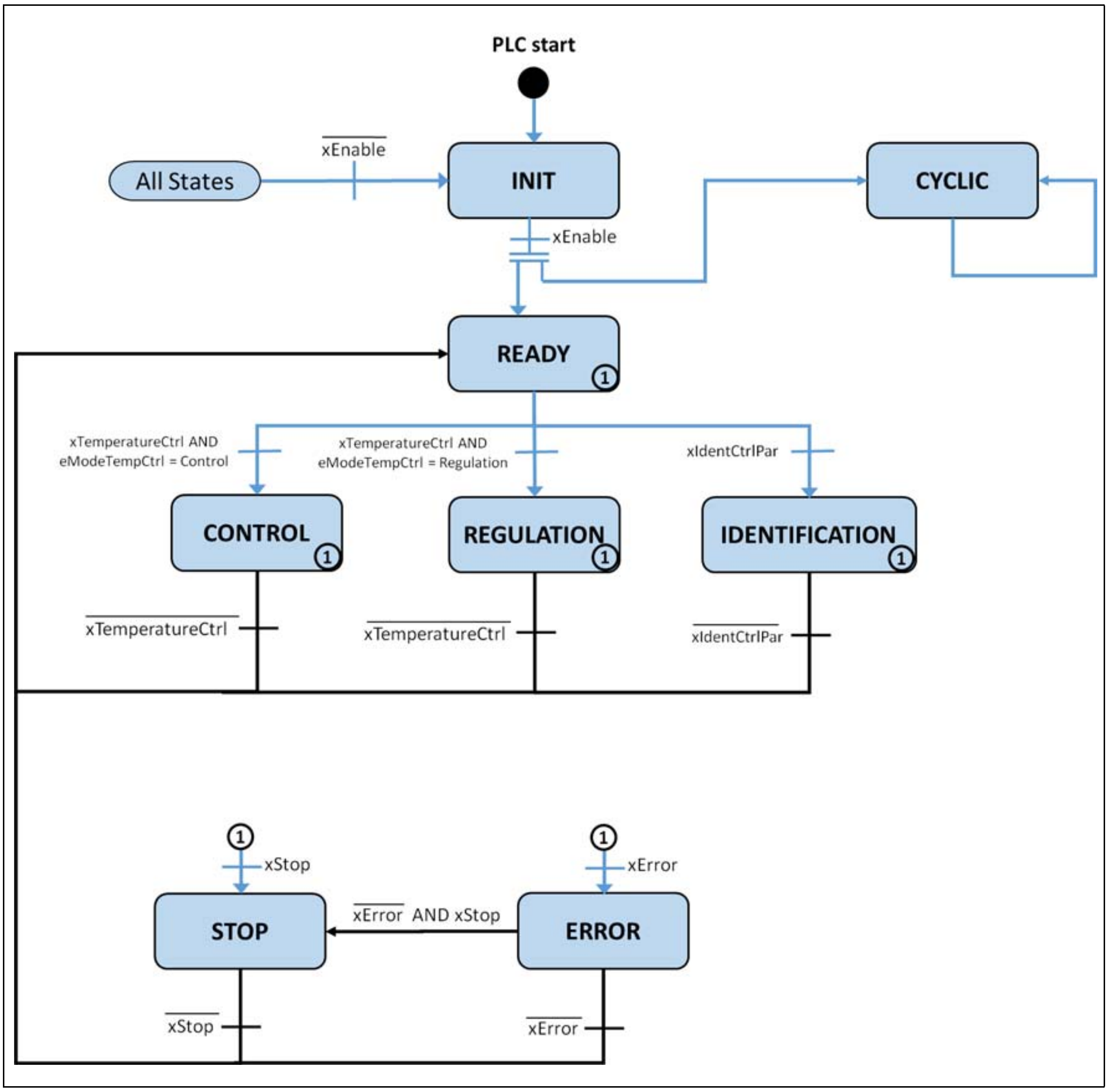
3.2.5 Persistent data

L_TT1P_scPD_TemperatureControlBase

The **L_TT1P_scPD_TemperatureControlBase** parameter structure contains the controller characteristic values calculated during the parameter identification (*xIdentCtrlPar* input = TRUE).

Designator Data type	Description
lrTempSetFilterTime LREAL	PT1 filter time constant for the temperature setpoint With the value '0', filtering is deactivated. • Unit: s • Initial value: 30.0
lrTempActFilterTime LREAL	PT1 filter time constant for the actual temperature value With the value '0', filtering is deactivated. • Unit: s • Initial value: 1.0 Example : A task cycle time of 10 ms and the common factor of 500 result in a filter time constant of 5 seconds.
lrTempCtrlGain BOOL	P parameter of the PID controller • Unit: %/°C / %/°F • Initial value: 0.0
lrTempCtrlResetTime LREAL	Reset time (I parameter) of the PID controller • Unit: s • Initial value: 100.0
lrTempCtrlRateTime LREAL	Rate time (D parameter) of the PID controller • Unit: s • Initial value: 100.0
dwIdentVersionInfo DWORD	Version identifier of the parameters detected by the parameter identification • This serves to detect whether valid parameters are available (dwIdentVersionInfo > 0). • Moreover, an external persistence mechanism can detect whether new parameters have been identified since the value of dwIdentVersionInfo is increased by 1 each time. • Initial value: 0

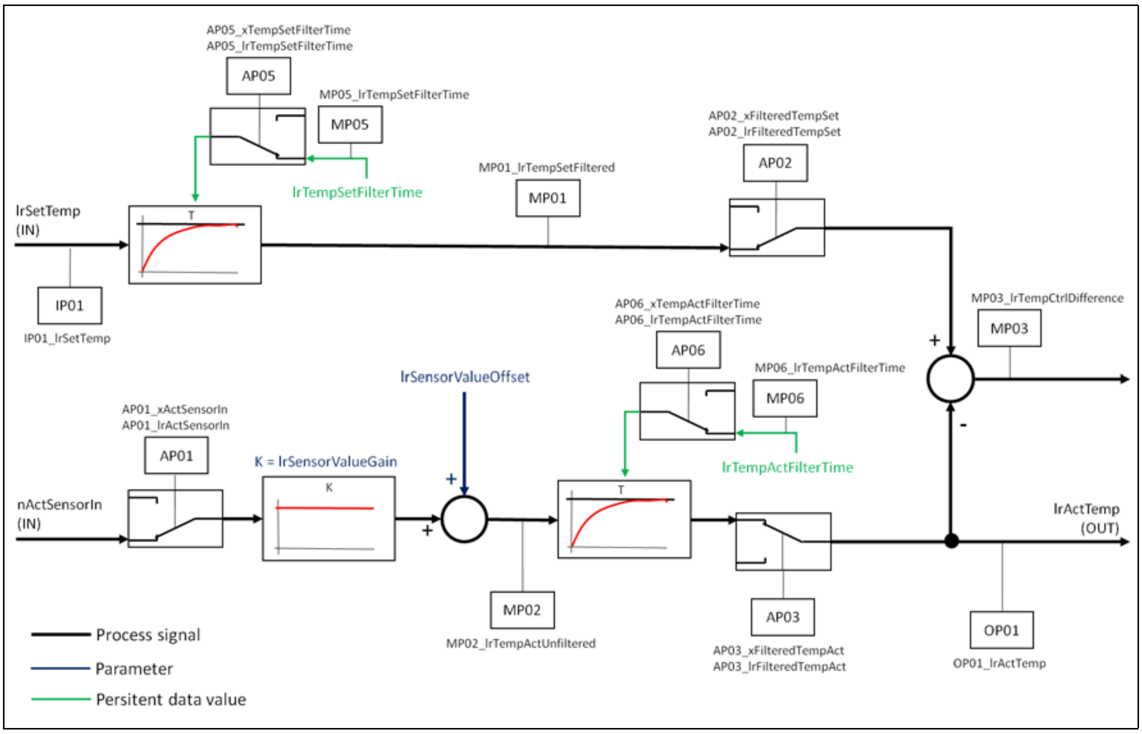
3.3 State machine



[3-2] State machine of the technology module

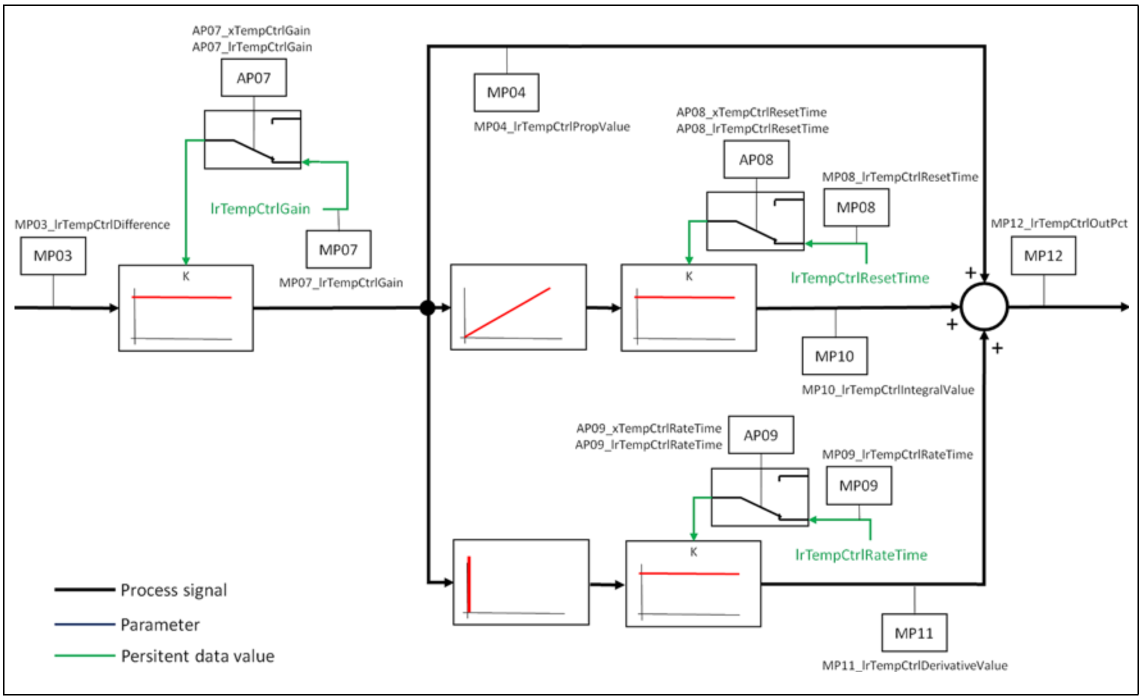
3.4 Signal flow diagrams

Setpoint/actual value conditioning



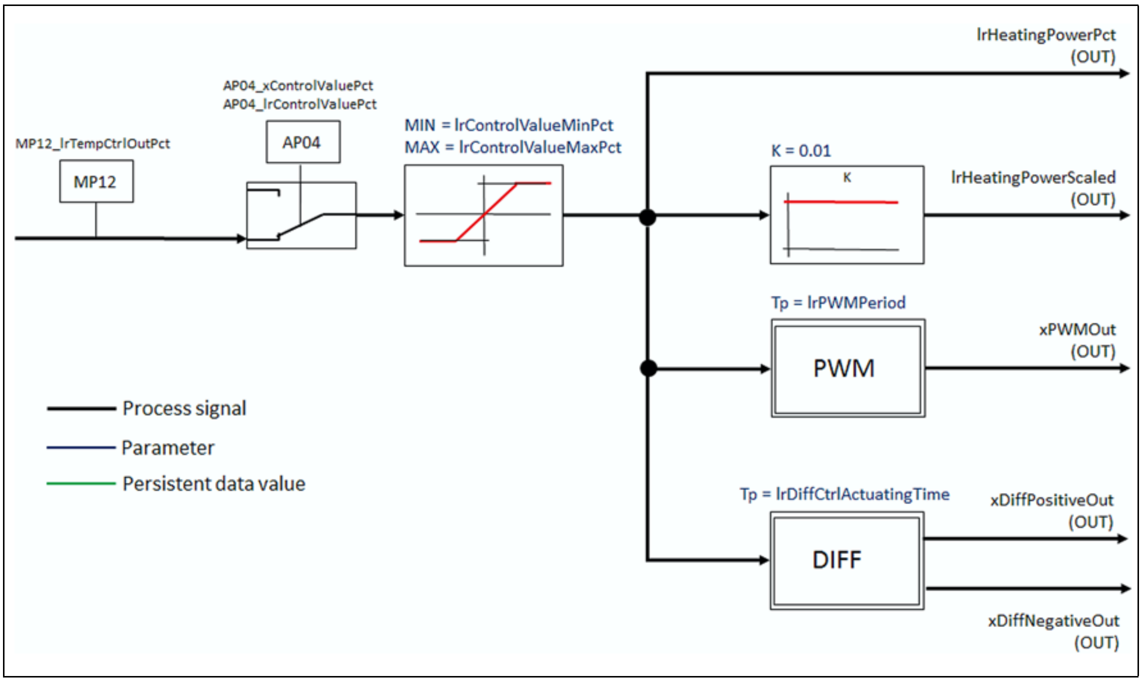
[3-3] Signal flow diagram: Setpoint/actual value conditioning

Thermostat



[3-4] Signal flow diagram: Thermostat

Correcting variable conditioning



[3-5] Signal flow diagram: Correcting variable conditioning

3.4.1 Structure of the signal flow

L_TT1P_scSF_TemperatureControlBase

The contents of the **L_TT1P_scSF_TemperatureControlBase** structure are read-only and offer a practical diagnostics option within the signal flow (Signal flow diagrams (📖 21)).

Designator Data type	Description
IP01_lrSetTemp LREAL	Temperature setpoint for the "temperature control" operating mode (eModeTempCtrl parameter = REGULATION) • Unit: °C or °F
MP01_lrTempSetFiltered LREAL	Filtered temperature setpoint • Unit: °C or °F
MP02_lrTempActUnfiltered LREAL	Unfiltered actual temperature value • Unit: °C or °F
MP03_lrTempCtrlDifference LREAL	System deviation between temperature setpoint and actual value • Unit: °C or °F
MP04_lrTempCtrlPropValue LREAL	Process value of the proportional component (P) of the controller
MP05_lrTempSetFilterTime LREAL	PT1 filter time constant for the temperature setpoint With the value '0', filtering is deactivated. • Unit: s
MP06_lrTempActFilterTime LREAL	PT1 filter time constant for the actual temperature value With the value '0', filtering is deactivated. • Unit: s Example : A task cycle time of 10 ms and the common factor of 500 result in a filter time constant of 5 seconds.
MP07_lrTempCtrlGain LREAL	P parameter of the PID controller • Unit: %/°C / %/°F
MP08_lrTempCtrlResetTime LREAL	Reset time (I parameter) of the PID controller • Unit: s
MP09_lrTempCtrlRateTime LREAL	Rate time (D parameter) of the PID controller • Unit: s
MP10_lrTempCtrlIntergral Value LREAL	Process value of the integral-action component (I) of the controller
MP11_lrTempCtrlDerivative Value LREAL	Process value of the differential component (D) of the controller
MP12_lrTempCtrlOutPct LREAL	Correcting variable of the controller • Unit: %
OP01_lrActTemp LREAL	Actual temperature value • Unit: °C or °F

3.4.2 Structure of the access points

L_TT1P_scAP_TemperatureControlBase

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.

<table>
<tr><th>Designator
Data type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="3">AP01_xActSensorIn
BOOL</td><td colspan="2">Selection of the sensor input signal for the temperature detection
• Initial value: FALSE</td></tr>
<tr><td>TRUE</td><td>The value of the AP01_lrActSensorIn access point is used for the temperature detection.</td></tr>
<tr><td>FALSE</td><td>The sensor input signal of the technology module (nActSensorIn input) is used for the temperature detection.</td></tr>
<tr><td>AP01_lrActSensorIn
LREAL</td><td colspan="2">Alternative value for the sensor input signal, activation with AP01_xActSensorIn = TRUE
• Unit: °C or °F
• Initial value: -</td></tr>
<tr><td rowspan="3">AP02_xFilteredTempSet
BOOL</td><td colspan="2">Selection of the temperature setpoint signal for the control
• Initial value: FALSE</td></tr>
<tr><td>TRUE</td><td>The value of the AP02_lrFilteredTempSet access point is used for the control.</td></tr>
<tr><td>FALSE</td><td>The original temperature setpoint from the technology module is used for the control.</td></tr>
<tr><td>AP02_lrFilteredTempSet
LREAL</td><td colspan="2">Alternative value for the temperature setpoint signal, activation with AP02_xFilteredTempSet = TRUE
• Unit: °C or °F
• Initial value: -</td></tr>
<tr><td rowspan="3">AP03_xFilteredTempAct
BOOL</td><td colspan="2">Selection of the actual temperature value signal for the control
• Initial value: FALSE</td></tr>
<tr><td>TRUE</td><td>The value of the AP03_lrFilteredTempAct access point is used for the control.</td></tr>
<tr><td>FALSE</td><td>The original actual temperature value from the technology module is used for the control.</td></tr>
<tr><td>AP03_lrFilteredTempAct
LREAL</td><td colspan="2">Alternative value for the actual temperature value signal, activation with AP03_xFilteredTempAct = TRUE
• Unit: °C or °F
• Initial value: -</td></tr>
<tr><td rowspan="3">AP04_xControlValuePct
BOOL</td><td colspan="2">Selection of the correcting variable signal for the control
• Initial value: FALSE</td></tr>
<tr><td>TRUE</td><td>The value of the AP04_lrControlValuePct access point is used for the control.</td></tr>
<tr><td>FALSE</td><td>The original correcting variable value from the technology module is used for the control.</td></tr>
<tr><td>AP04_lrControlValuePct
LREAL</td><td colspan="2">Alternative value for the correcting variable signal, activation with AP04_xControlValuePct = TRUE
• Unit: %
• Initial value: -</td></tr>
</table>

Designator / Data type	Description	
AP05_xTempSetFilterTime BOOL	Selection of the PT1 filter time constant for the temperature setpoint • Initial value: FALSE	
	TRUE	The value of the AP05_lrTempSetFilterTime access point is used for the control.
	FALSE	The PT1 filter time constant from the technology module (lrTempSetFilterTime from L_TT1P_scPD_TemperatureControlBase (📖 19)) is used for the control.
AP05_lrTempSetFilterTime LREAL	Alternative value for the PT1 filter time constant, activation with AP05_xTempSetFilterTime = TRUE • Unit: s • Initial value: -	
AP06_xTempActFilterTime BOOL	Selection of the PT1 filter time constant for the actual temperature value • Initial value: FALSE	
	TRUE	The value of the AP06_lrTempActFilterTime access point is used for the control.
	FALSE	The PT1 filter time constant from the technology module (lrTempActFilterTime from L_TT1P_scPD_TemperatureControlBase (📖 19)) is used for the control.
AP06_lrTempActFilterTime LREAL	Alternative value for the PT1 filter time constant, activation with AP06_xTempActFilterTime = TRUE • Unit: s • Initial value: -	
AP07_xTempCtrlGain BOOL	Selection of the P parameter of the PID controller • Initial value: FALSE	
	TRUE	The value of the AP07_lrTempCtrlGain access point is used for the control.
	FALSE	The P parameter of the PID controller from the technology module (lrTempCtrlGain from L_TT1P_scPD_TemperatureControlBase (📖 19)) is used for the control.
AP07_lrTempCtrlGain LREAL	Alternative value for the P parameter of the PID controller, activation with AP07_xTempCtrlGain = TRUE • Unit: %/°C / %/°F • Initial value: -	
AP08_xTempCtrlResetTime BOOL	Selection of the reset time (I parameter) of the PID controller • Initial value: FALSE	
	TRUE	The value of the AP08_lrTempCtrlResetTime access point is used for the control.
	FALSE	The reset time (I parameter) of the PID controller from the technology module (lrTempCtrlResetTime from L_TT1P_scPD_TemperatureControlBase (📖 19)) is used for the control.
AP08_lrTempCtrlResetTime LREAL	Alternative value for the reset time (I parameter) of the PID controller, activation with AP08_xTempCtrlResetTime = TRUE • Unit: s • Initial value: -	
AP09_xTempCtrlRateTime BOOL	Selection of the rate time (D parameter) of the PID controller • Initial value: FALSE	
	TRUE	The value of the AP09_lrTempCtrlRateTime access point is used for the control.
	FALSE	The rate time (D parameter) of the PID controller from the technology module (lrTempCtrlRateTime from L_TT1P_scPD_TemperatureControlBase (📖 19)) is used for the control.
AP09_lrTempCtrlRateTime LREAL	Alternative value for the rate time (D parameter) of the PID controller, activation with AP09_xTempCtrlRateTime = TRUE • Unit: s • Initial value: -	

3.4.3 Use of the access points

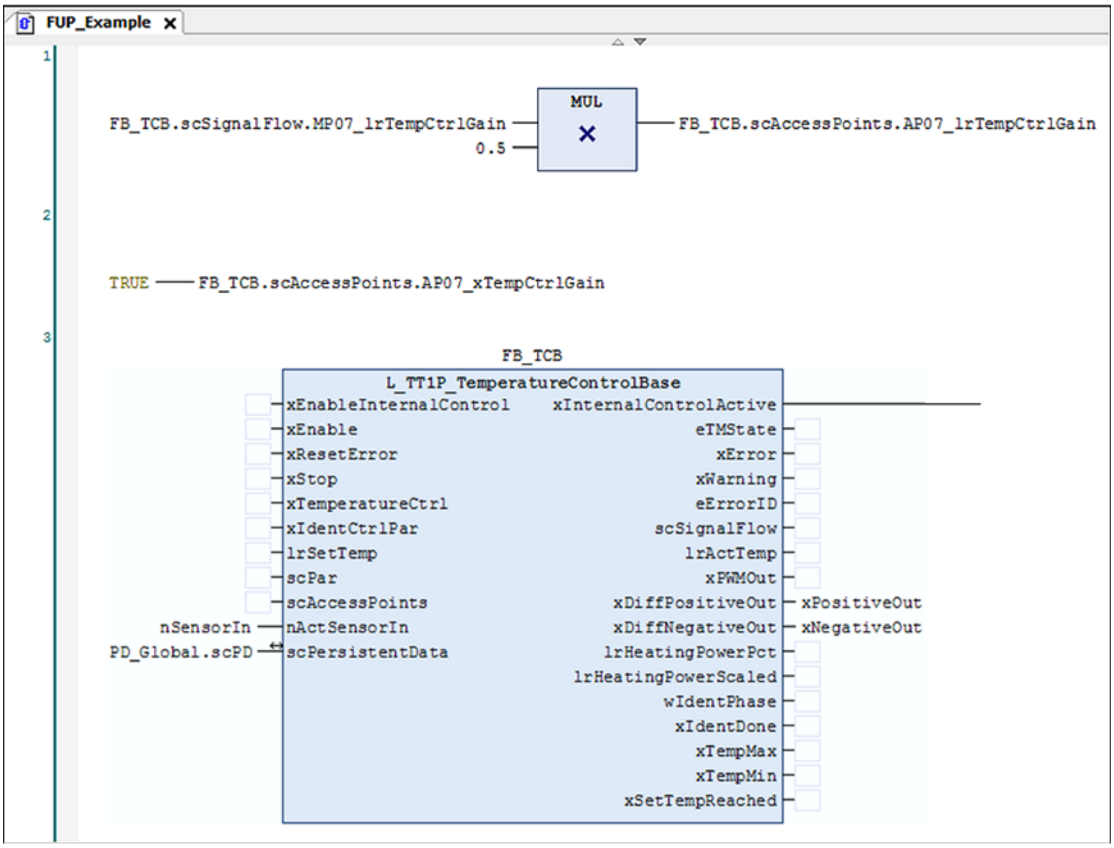
The use of the access points is only required in exceptional cases.

The following application example explains the use of the access points for correcting values of the controller parameters.

Application example

Due to path-specific features, the value of the *lrTempCtrlGain* controller parameter detected by the parameter identification shall not be used directly for the heat section but shall be reduced by the factor '2' (scaling to 50 %).

The required additional external wiring of the function block **L_TT1P_TemperatureControlBase** is shown here:



- In network 1, the controller parameter value is adapted. Here, the *MP07_lrTempCtrlGain* measuring point which provides the value detected by the parameter identification (see Signal flow diagram: Thermostat (📖 21)) is multiplied by the factor '0.5' and is switched to the *AP07_lrTempCtrlGain* access point.
- In network 2, the *AP07_lrTempCtrlGain* is activated by setting the *AP07_xTempCtrlGain* signal to TRUE.
- In network 3, the implementation of the "FB_TCB" module instance is included with minimum circuitry.

3.5 Basic settings

The maximum absolute temperature limit has to be adapted via the *lrTempMaxCriticalAbsolute* parameter (initial value 120 °C) for the heating system and/or the medium to be heated.

The standard values of the other parameters are selected such that the **L_TT1P_TemperatureControlBase** function block can be used without adapting the parameter values.

3.6 Task cycle time

The cycle time of the task in which the **L_TT1P_TemperatureControlBase** function block is used has to be adjusted to the PWM period (*lrPWMPeriod* parameter). Here, the ratio of the times has to be at least 1:100.

Example: In case of a standard PWM period of 1.0 s, the task cycle time must be ≤ 10 ms.

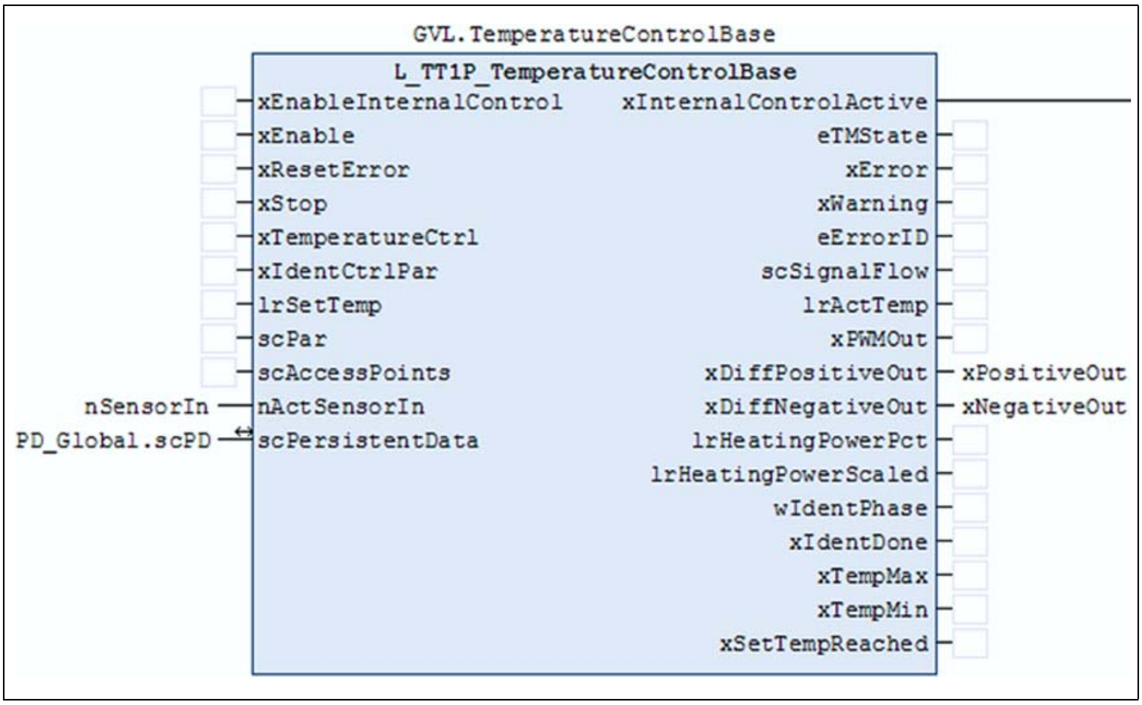
If the ratio of the PWM period to the task cycle time is less than 1:50, the *eErrorID* output displays the warning "17134" (TaskCyclePWMPeriodMismatch).

"FAST technology modules" reference manual

Here you can find information on error and warning messages.

3.7 Minimum external wiring for the use of the DIFF module

The minimum external wiring of the technology module for the use with the DIFF module is shown in the following illustration.



[3-6] Minimum external wiring of the function block

Connection of the thermal sensor

The *nActSensorIn* sensor input, for instance, can be directly connected to the output signal of an EPM-S405 module (I/O system 1000).

The *lrSensorValueGain* parameter serves to scale the sensor signal. (The EPM-S405 module provides an integer temperature value with the resolution 0.1 °C). The initial value ’0.1’ of the parameter ensures that in this case the temperature will be detected correctly without any further adaptations. The temperature can be checked by the *lrActTemp* output.

Connecting the mixing valve for the heating control

Usually the mixing valve is controlled via an actuator which is provided with two input signals:

- One signal for opening the valve (generally CCW rotation of the motor), which is connected to the *xDiffPositiveOut* output.
- One signal for closing the valve (generally CW rotation of the motor), which is connected to the *xDiffNegativeOut* output.

The *lrDiffCtrlActuatingTime* parameter is used to set the control time (opening time of the valve), in order to achieve better control results of the mixing valve.

Persistent data

Generally, the controller parameters detected in the course of the parameter identification process are to be maintained even if the controller is restarted. For this case, an instance that is part of the persistent variable list has to be connected to the *scPersistentData* input/output. For the wiring example shown in figure [3-6] this would be as follows:

```
VAR_GLOBAL PERSISTENT RETAIN
   scPD : L_TT1P_scPD_TemperatureControlBase;
END_VAR
```

3.8 Parameter identification

Before using the **L_TT1P_TemperatureControlBase** function block as thermostat, the parameter identification has to be carried out once. In this process, the system behaviour of the heat section is detected and the optimum values for the controller parameters are derived.

3.8.1 Activation

How to activate the parameter identification:

1. Set the *xEnable* input = TRUE.
   - The function block is switched on.
2. Set the *xIdentCtrlPar* input = TRUE.
   - The identification is started.
   - The function block changes to the @IDENTIFICATION" state (212).

Procedure

- **Phase 1:** Waiting for the ambient temperature

  The identification needs to be started at ambient temperature.

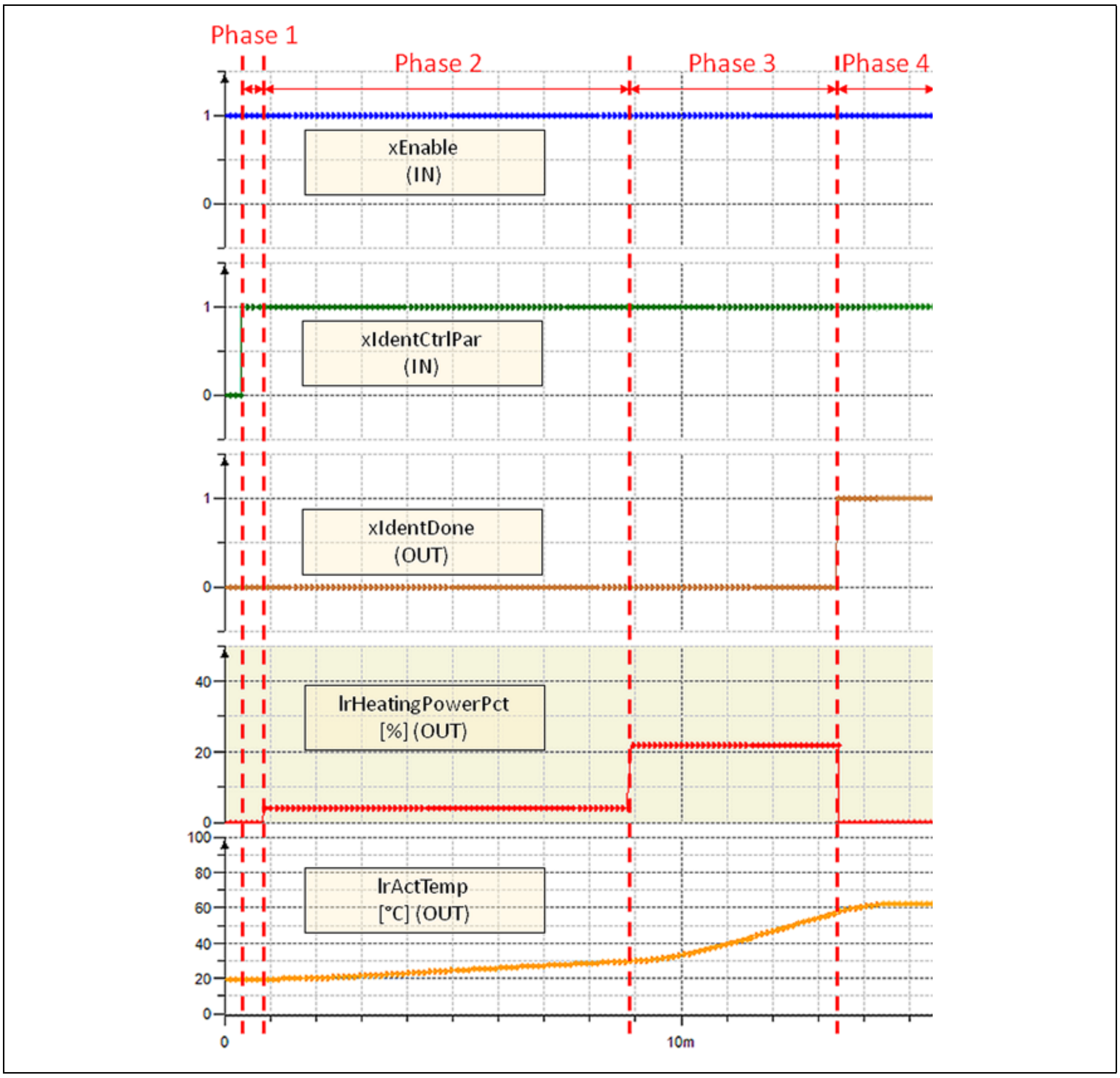
  For this purpose, the change of the current temperature is monitored while the heating element is switched off (correcting variable = 0%). Only when the change rate falls below a limit value (stationary ambient temperature is reached), phase 1 is completed.
- **Phase 2:** Applying a small correcting variable

  The heating element is applied with a small correcting variable and is it waited until a stationary temperature has been reached. This is required to detect the fundamental heating capacity of the heat section and calculate the correcting variable step that is optimal for phase 3.

  The current correcting variable for the heating element can be read at the *lrHeatingPowerPct* output (scaled in percent).
- **Phase 3:** Determining the behaviour of the heat section
  - The heating element is applied with the optimal correcting variable calculated in phase 2.
  - The relevant characteristic values of the heat section are detected.
- **Phase 4:** Calculating the values for the controller parameters and terminating the identification
  - The heating element is switched off (correcting variable = 0 %)
  - The characteristic values detected in phase 3 serve to calculate the optimum values for the controller parameters which are saved in the Persistent data (🕮 32).
  - The termination of the parameter identification is signalled: *xIdentDone* output = TRUE

3.8.2 Signal characteristic

The diagram shows the signal characteristic during the previously described Parameter identification (📖 30).



[3-7] Signal characteristic during parameter identification

3.8.3 Persistent data

The persistent data in the L_TT1P_scPD_TemperatureControlBase (📖 19) include the following elements:

```
TYPE _scPD_TM_TemperatureControlBase EXTENDS _scPD_TM_Base :
STRUCT
   lrTempSetFilterTime : LREAL := 30.0;
   lrTempActFilterTime : LREAL := 1.0;
   lrTempCtrlGain : LREAL := 0.0;
   lrTempCtrlResetTime : LREAL := 100.0;
   lrTempCtrlRateTime : LREAL := 100.0;
   dwIdentVersionInfo : DWORD := 0;
END_STRUCT
END_TYPE
```

The *dwIdentVersionInfo* parameter has a special meaning. The initial value '0' indicates that so far no Parameter identification (📖 30) has been carried out. With each parameter identification that has been completed, the values identified for the controller parameters are entered into the LREAL data elements. Afterwards the value of *dwIdentVersionInfo* is increased by '1'.

3.9 Temperature control

If valid values for the controller parameters are available (Persistent data (32)), the temperature control function can be activated.

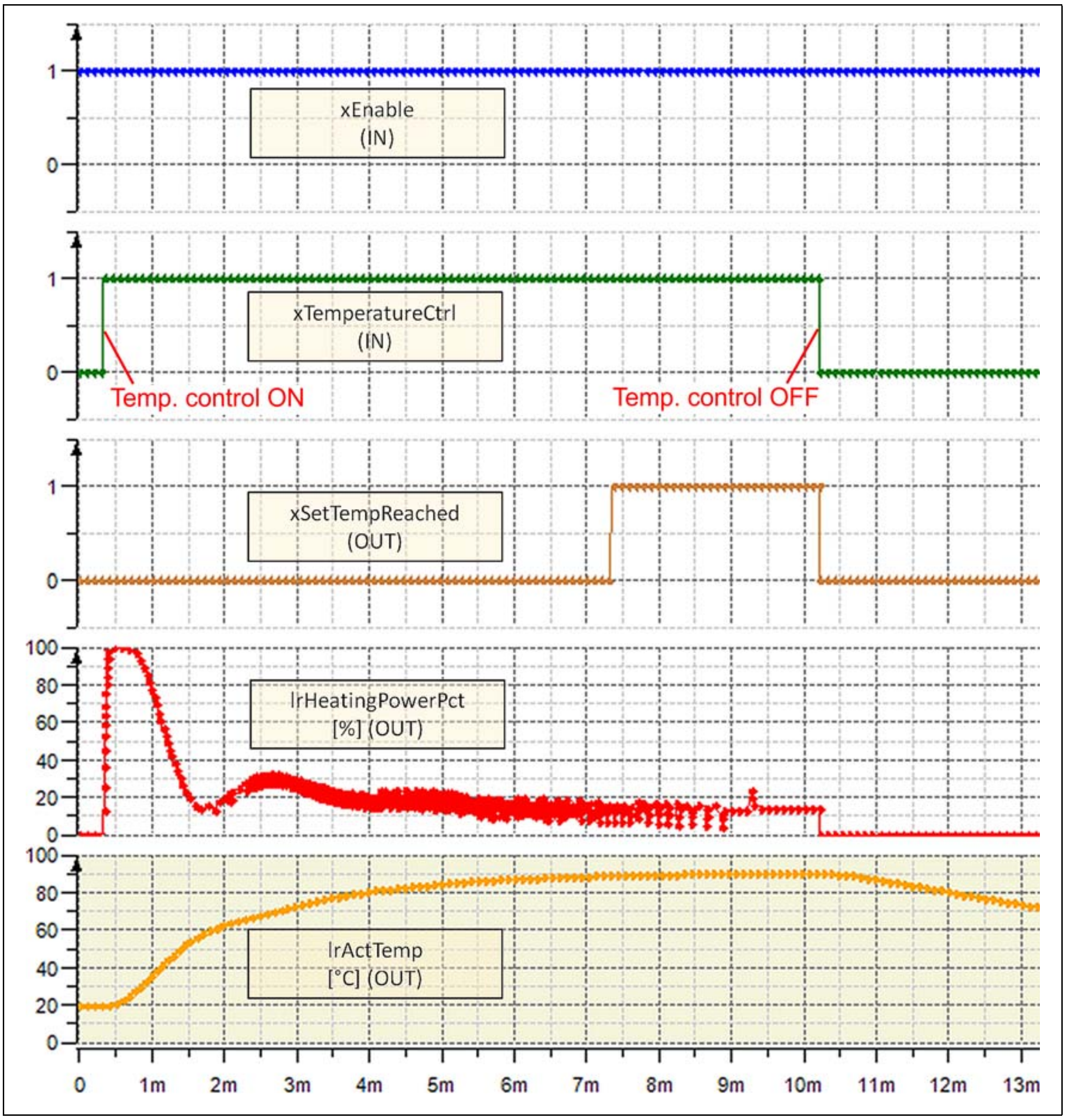
3.9.1 Activation

How to activate the temperature control:

1. Set the *xEnable* input = TRUE.
   - The function block is switched on.
2. Set the *eModeTempCtrl* parameter = REGULATION.
3. Define the temperature setpoint to be reached via the *lrSetTemp* input.
4. Set the *xTemperatureCtrl* input = TRUE.
   - The temperature control is started.
   - The function block changes to the "REGULATION" state (211).

3.9.2 Signal characteristic

The sample diagram shows the signal characteristic of the temperature control with a setpoint temperature of 90 °C.



[3-8] Signal characteristic during the temperature control

3.10 Control with constant correcting variable

The special operating mode "control with constant correcting variable" is intended for maintaining an emergency operation if the thermal sensor or its evaluation unit is defective and thus no information on the current actual temperature is available. In this case, a constant manipulating variable serves to control the heating element.

Stop!

Temperature monitoring is deactivated

In case of control with constant correcting variable, all temperature monitoring modes are deactivated.

Possible consequences

Damages to the heating system or the medium to be heated

Protective measures

Due to previous process monitoring or know-how, select the correcting variable in such a way that the desired actual temperature is set by approximation.

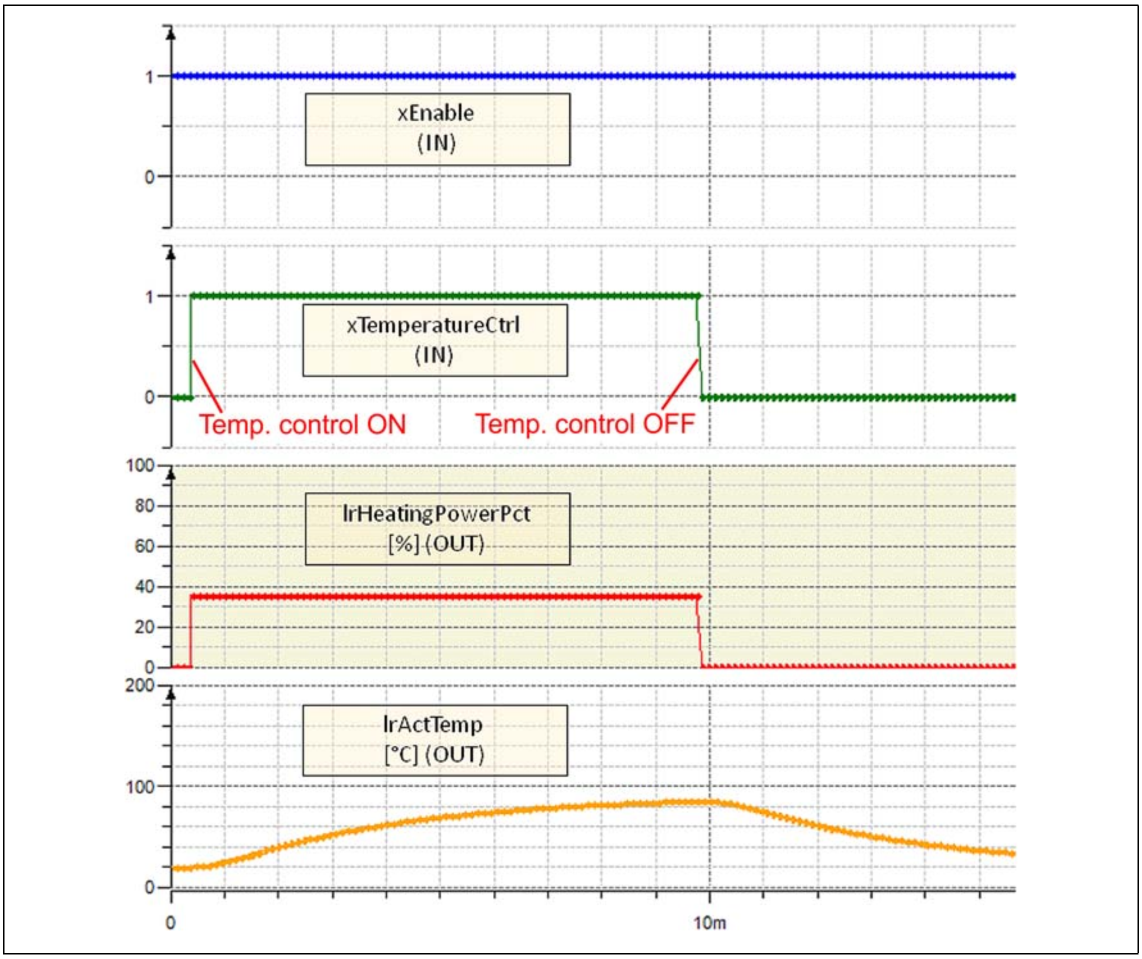
3.10.1 Activation

How to activate the control with constant correcting variable:

1. Set the *xEnable* input = TRUE.
   - The function block is switched on.
2. Set *eModeTempCtrl* parameter = CONTROL.
3. Set the value of the correcting variable in the *lrControlValueFixedPct* parameter (initial value is 10 %).
4. Set the *xTemperatureCtrl* input = TRUE.
   - The control is started.
   - The function block changes to the "CONTROL" state (210).

3.10.2 Signal characteristic

The sample diagram shows the signal characteristic of the control with constant correcting variable with a value of 35 %.



[3-9] Signal characteristic during control with constant correcting variable

Index

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