SIEMENS

Library description • 03/2017

Library for Controlled System Simulation with STEP 7 (TIA Portal)

STEP 7 V14

https://support.industry.siemens.com/cs/ww/en/view/79047707

Warranty and liability

Note

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are used correctly. These Application Examples do not relieve you of the responsibility to use safe practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice.
If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications − e.g. Catalogs − the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.
Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act (“Produkthaftungsgesetz”), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract (“wesentliche Vertragspflichten”). The damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.
Any form of duplication or distribution of these Application Examples or excerpts hereof is prohibited without the expressed consent of the Siemens AG.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.
In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens’ products and solutions only form one element of such a concept.
Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.
Additionally, Siemens’ guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity.

Siemens’ products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer’s exposure to cyber threats.
To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under http://www.siemens.com/industrialsecurity.

Table of contents

1 Library Overview

What you get

This document describes the “LSim” block library. The block library provides you with the tested code with clearly defined interfaces. They can be used as a basis for your task to be implemented.

A key concern of the document is to describe

- all blocks of the block library
- the functionality implemented through these blocks.

Furthermore, this documentation shows possible fields of application and helps you integrate the library into your STEP 7 project using step-by-step instructions.

1.1 Different user scenarios

Possible application(s) for the “LSim” library

The “LSim” library allows you to simulate controlled systems in an S7 CPU.

The blocks of the library simulate various simple controlling elements through linear transfer elements. Suitable serial or parallel interconnection of the individual system elements allows you to simulate even very complex controlled systems.

The library can be used, for example, in the following scenarios:

- Optimizing a controller: Simulation of the controlled system where commissioning on the real process is difficult or hardly possible.
- For training purposes: Simulation of individual controlled system elements for a clear representation of control engineering processes and demonstration of software controllers.

1.2 Hardware and software requirements

Requirements for this library

To be able to use the functionality of the library described in this document, the following hardware and software requirements must be met:

Hardware

The library was tested with the following SIMATIC CPUs:

- S7-1200 firmware v4.2
- S7-1500 firmware v2.0

Software

STEP 7 (TIA Portal) Basic or Professional from V14 Update 2 or higher

1.3 Library resources

What will you find in this section?

The following section gives you an overview of the size of the blocks of the “LSim” library in the main memory.

Size of the individual blocks

Table 1-1 : Memory usage (CPU 1211 DC/DC/DC V4.2, CPU 1511-1 PN V2.0)

Block	Version	S7-1200 allocation (in bytes)		S7-1500 allocation (in bytes)	
		Load memory	Main memory	Load memory	Main memory
LSim_PT1	2.0.1	8954	284	8984	355
LSim_PT1asym	2.0.1	11266	429	11377	500
LSim_PT2osc	2.0.1	13528	624	13599	695
LSim_PT2aper	2.0.1	12644	584	12687	655
LSim_PT3	2.0.1	11353	558	11359	624
LSim_PDT1	2.0.1	11040	419	11077	491
LSim_I	2.0.1	9070	284	9150	363
LSim_IT1	2.0.1	11457	496	11526	567
LSim_TempProcess	2.0.1	10912	400	11024	475
LSim_Lagging	2.0.2	7061	205	7078	269
LSim_DT1	2.0.1	10136	355	10223	427
LSim_Allpass1OrdReal	2.0.1	9800	351	9898	422
LSim_Allpass2OrdReal	2.0.1	12731	584	12848	656
LSim_Valve	2.0.1	17100	496	17208	537
LSim_PT3HeatCool	2.0.1	15384	735	15507	800
LSim_Multizone	1.0.1	25884	1630	25990	1718

2 Blocks of the Library

What will you find in this section?

This chapter lists (chapter 2.1) and explains (chapter 2.2) all blocks of the “LSim” library.

2.1 List of the blocks

The following table lists all the blocks of the “LSim” library.

Chapter 2.2 provides a detailed description of the blocks.

Table 2-1

Block	Brief description	
LSim_PT1	Simulation of a PT1 system	Self-regulating systems
LSim_PT1asym	Simulation of an asymmetrical PT1 system	
LSim_PT2osc	Simulation of a PT2 system in the periodic case	
LSim_PT2aper	Simulation of a PT2 system in the aperiodic case	
LSim_PT3	Simulation of a PT3 system	
LSim_PDT1	Simulation of a PDT1 system	
LSim_TempProcess	Simulation of a temperature process	
LSim_DT1	Simulation of a DT1 system	
LSim_Allpass1OrdReal	Simulation of a first-order all-pass	
LSim_Allpass2OrdReal	Simulation of a second-order all-pass	
LSim_PT3HeatCool	Simulation of a PT3 system with separate inputs for heating and cooling	
LSim_Multizone	Simulation of thermally coupled zones	
LSim_I	Simulation of an I system	Systems without inherent regulation
LSim_IT1	Simulation of an IT1 system	
LSim_Valve	Simulation of a valve	
LSim_Lagging	Simulation of a lag element	

2.2 Explanation of the blocks

This chapter introduces the parameters of the blocks.
For some parameters, it shows the transfer function of the block and the step response of the respective block.

2.2.1 FB „LSim_PT1“

FB “LSim_PT1” simulates a PT1 element. The PT1 element is a proportional transfer element with first-order order delay.

Use case

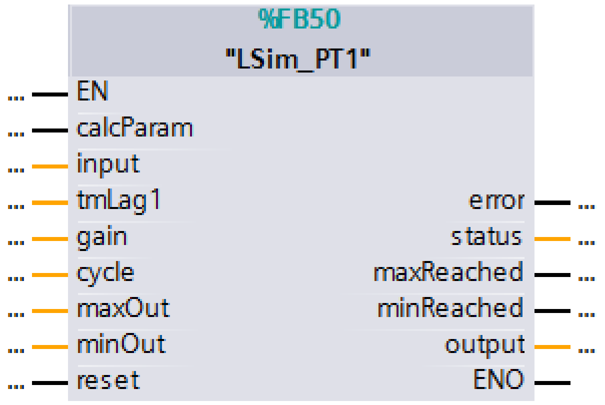
A PT1 element can be used, for example, to simulate a temperature system.

Transfer function

$$F(p) = \frac{gain}{tmLag1 * p + 1}$$

Figure

Figure 2-1: PT1 element



Input parameters

Table 2-2

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when “tmLag1”, “gain” or “cycle” changes)
input	Real	Input variable
tmLag1	Real	Time constant (in seconds)
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, “output” = “maxOut” will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, “output” = “minOut” will be set)
reset	Bool	Resets all relevant parameters to ‘0’, including “output”.

Output parameters

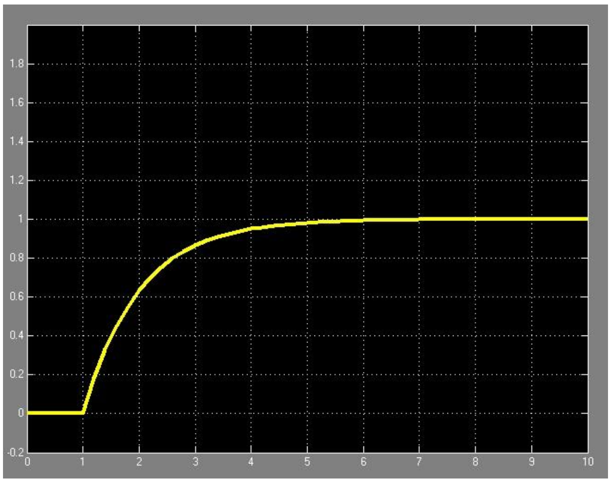
Table 2-3

Parameter	Data type	Description
error	Bool	If “tmLag1” <= 0, “error” = TRUE will be set.
status	Word	If “tmLag1” <= 0, “status” = 16#8001 will be set.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = gain * \left(1 - e^{-\frac{t}{tmLag1}}\right)$$

Figure 2-2: PT1 element step response



2.2.2 FB „LSim_PT1asym"

FB "LSim_PT1asym" simulates a PT1 system whose time response differs depending on whether the trigger is positive or negative. This makes it an asymmetrical PT1 system.

Use case

An asymmetrical PT1 element can be used, for example, to simulate a temperature system with different behavior for heating and cooling.

Transfer function

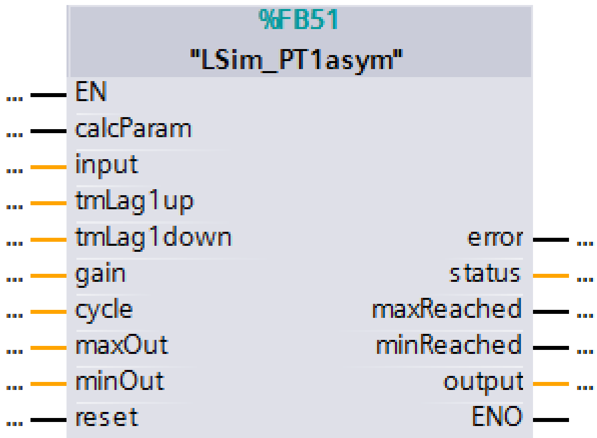
Unlike FB "LSim_PT1", the transfer function internally consists of two different PT1 elements for the up and down movement of the process value.

$$F_{up}(p) = \frac{gain}{tmLag1up * p + 1}$$

$$F_{down}(p) = \frac{gain}{tmLag1down * p + 1}$$

Figure

Figure 2-3: Asymmetrical PT1 element



Input parameters

Table 2-4

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when "tmLag1up", "tmLag1down", "gain" or "cycle" changes)
input	Real	Input variable
tmLag1up	Real	Time constant (in seconds) in the event of positive triggering ("input" > "output")
tmLag1down	Real	Time constant (in seconds) in the event of negative triggering ("input" < "output")
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the

Parameter	Data type	Description
		maximum limit, “output” = “maxOut” will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, “output” = “minOut” will be set)
reset	Bool	Resets all relevant parameters to ‘0’, including “output”.

Output parameters

Table 2-5

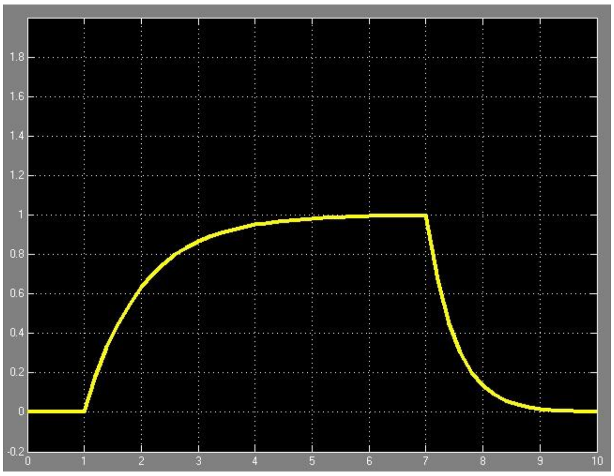
Parameter	Data type	Description
error	Bool	If “tmLag1up” <= 0 or “tmLag1down” <= 0, “error” = TRUE will be set.
status	Word	If “tmLag1up” <= 0 or “tmLag1down” <= 0, “status” = 16#8001 will be set.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = gain * \left(1 - e^{-\frac{t}{tmLag1up}}\right)$$ for a positive unit step

$$y(t) = gain * \left(1 - e^{-\frac{t}{tmLag1down}}\right)$$ for a negative unit step

Figure 2-4: Asymmetrical PT1 element step response



2.2.3 FB „LSim_PT2osc”

FB “Sim_PT2osc” simulates an oscillating PT2 system. Accordingly, the damping factor selected for this system must be less than 1. Accordingly, the system is in the periodic case.

Use case

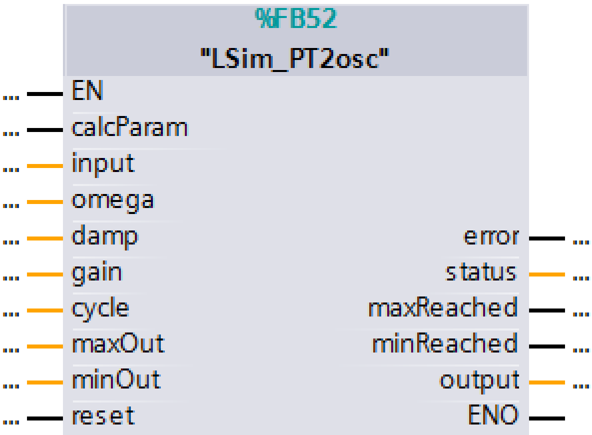
A periodic PT2 element can be used, for example, to simulate mechanical systems that execute a stroke/rotation and are oscillating.

Transfer function

$$F(p) = \frac{gain}{(\frac{p}{omega})^2 + 2 * damp * \frac{p}{omega} + 1}$$

Figure

Figure 2-5: PT2 element in the periodic case



Input parameters

Table 2-6

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when “omega”, “damp”, “gain” or “cycle” changes)
input	Real	Input variable
omega	Real	Angular frequency of the free oscillation
damp	Real	Damping ratio: 0 < “damp” < 1
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, “output” = “maxOut” will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, “output” = “minOut” will be set)
reset	Bool	Resets all relevant parameters to ‘0’, including “output”.

Output parameters

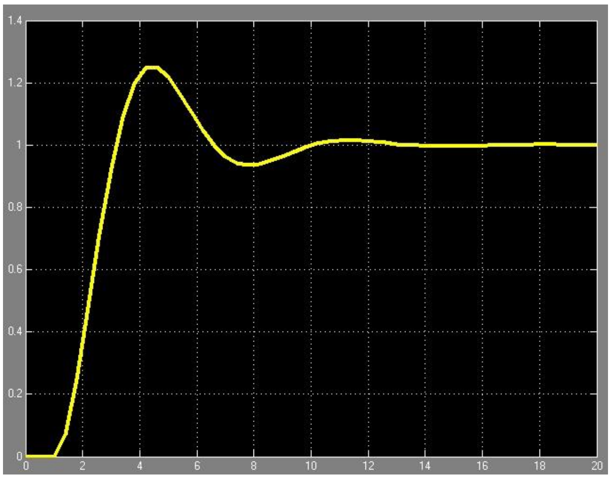
Table 2-7

Parameter	Data type	Description
error	Bool	If the conditions 0 < "damp" < 1 or "omega" > 0 are not met, "error" = TRUE will be set.
status	Word	If the conditions 0 < "damp" < 1 or "omega" > 0 are not met, "status" = 16#8001 will be output.
maxReached	Bool	When "maxReached" = TRUE, the "output" output variable was limited to the maximum value "maxOut".
minReached	Bool	When "minReached" = TRUE, the "output" output variable was limited to the minimum value "minOut".
output	Real	Output variable (valid only when "error" = FALSE)

Step response

$$y(t) = gain - \frac{gain}{\sqrt{1-damp^2}} e^{-damp*tmLag1*T} * \sin\left[\frac{\sqrt{1-damp^2}}{\frac{1}{tmLag1}} t + \arctan\left(\frac{\sqrt{1-damp^2}}{damp}\right)\right]$$

Figure 2-6: PT2 element with 0<"damp"<1



2.2.4 FB „LSim_PT2aper"

FB "LSim_PT2aper" simulates a PT2 system with a damping factor>1. The system is in the aperiodic case where no overshoot occurs.

Use case

An aperiodic PT2 element can be used, for example, to simulate a spring pendulum.

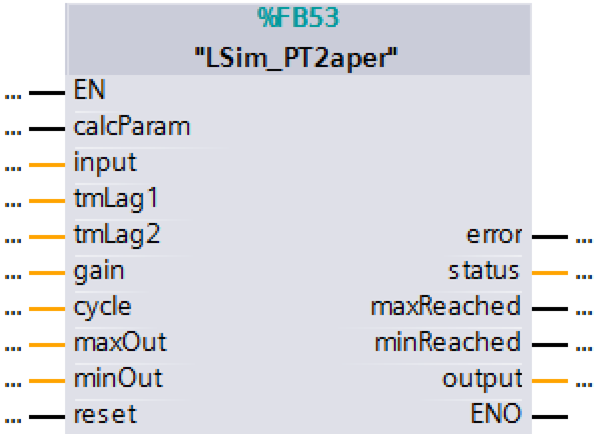
Transfer function

Unlike FB "LSim_PT2osc", the transfer function internally consists of two PT1 elements connected in series:

$$F(p) = \frac{gain}{(tmLag1 \ * p + 1) * (tmLag2 * p + 1)}$$

Figure

Figure 2-7



Input parameters

Table 2-8

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when "tmLag1up", "tmLag1down", "gain" or "cycle" changes)
input	Real	Input variable
tmLag1	Real	Time constant for the first PT1 element (in seconds)
tmLag2	Real	Time constant for the second PT1 element (in seconds)
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, "output" = "maxOut" will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, "output" = "minOut" will be set)
reset	Bool	Resets all relevant parameters to '0', including "output".

Output parameters

Table 2-9

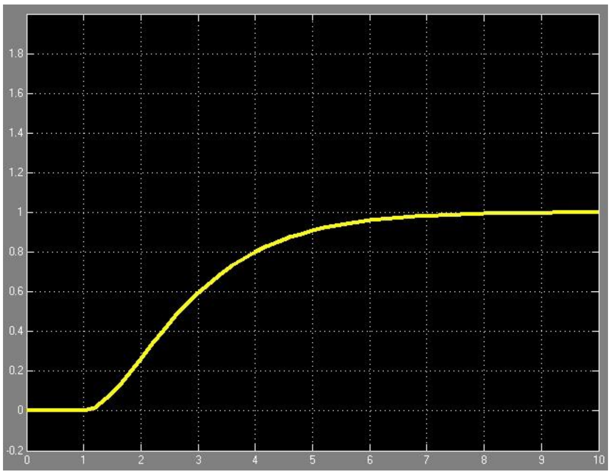
Parameter	Data type	Description
error	Bool	If the time constants are identical or <= 0, “error” = TRUE will be set.
status	Word	If the time constants are identical or <= 0, “status” = 16#8001 will be output.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = gain - \frac{gain}{tmLag1 - tmLag2}\left[tmLag1\ e^{\frac{-t}{tmLag1}} - tmLag2\ e^{\frac{-t}{tmLag2}}\right]$$

$$tmLag1 <> tmLag2$$

Figure 2-8: Step response of PT2 element with D>1



2.2.5 FB „LSim_PT3”

FB “LSim_PT3” simulates a third-order delay element.
Internally, it consists of the serial interconnection of three PT1 systems.

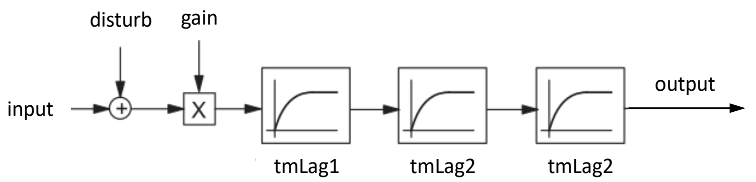
Use case

A PT3 element can be used, for example, to simulate a temperature system with multiple storage elements.

Transfer function

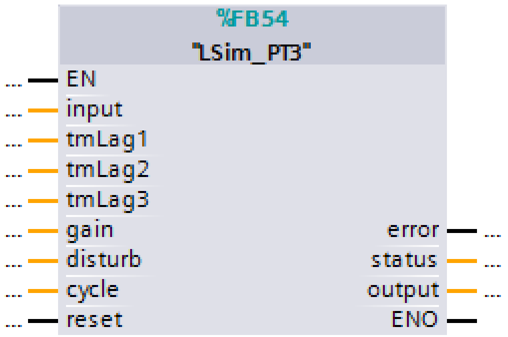
$$F(p) = \frac{gain}{(tmLag1 * p\ + 1) * (tmLag2\ * p + 1) * (tmLag3\ * p + 1)}$$

Figure 2-9: Structure and parameters of the “LSim_PT3” system block



Figure

Figure 2-10: PT3 element



Input parameters

Table 2-10

Parameter	Data type	Description
input	Real	Input variable
tmLag1	Real	Time constant for the first PT1 element (in seconds)
tmLag2	Real	Time constant for the second PT1 element (in seconds)
tmLag3	Real	Time constant for the third PT1 element (in seconds)
gain	Real	Gain factor
disturb	Real	Disturbance variable (applied at the “input” input)
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
reset	Bool	Resets all relevant parameters (including “output” = “input”).

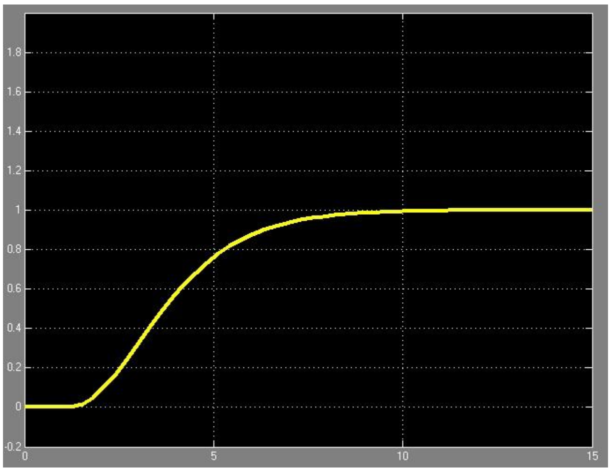
Output parameters

Table 2-11

Parameter	Data type	Description
error	Bool	If one of the time constants <= 0, “error” = TRUE will be set.
status	Word	If one of the time constants <= 0, “status” = 16#8001 will be output.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

Figure 2-11: PT3 element step response



2.2.6 FB „LSim_PDT1"

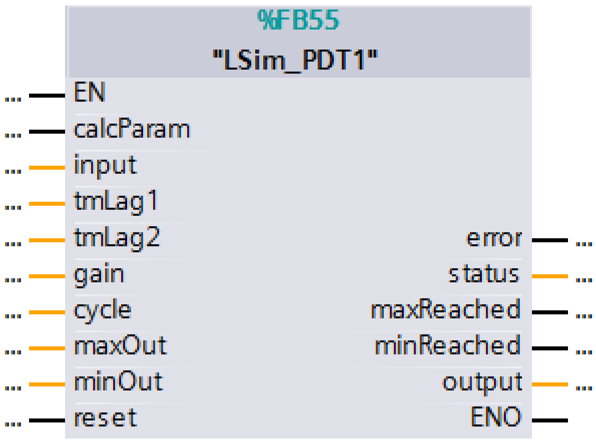
FB "LSim_PDT1" simulates the behavior of a PDT1 element (generic rational first-order element).

Transfer function

$$F(p) = \frac{gain * (1 + tmLag2 * p)}{(1 + tmLag1 * p)}$$

Figure

Figure 2-12



Input parameters

Table 2-12

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when "tmLag1", "tmLag2", "gain" or "cycle" changes)
input	Real	Input variable
tmLag1	Real	First time constant (in seconds)
tmLag2	Real	Second time constant (in seconds)
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, "output" = "maxOut" will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, "output" = "minOut" will be set)
reset	Bool	Resets all relevant parameters to '0', including "output".

Output parameters

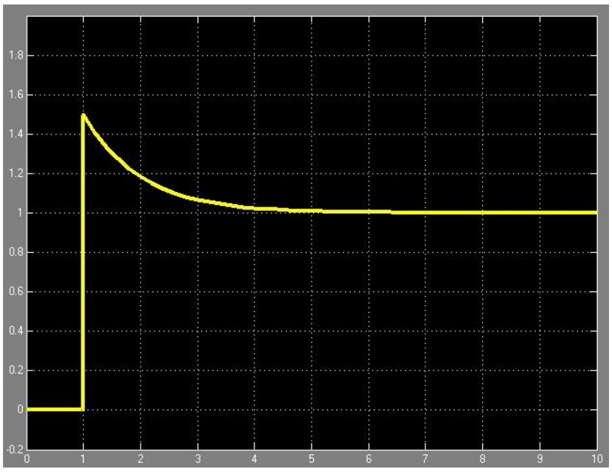
Table 2-13

Parameter	Data type	Description
error	Bool	If one of the time constants <= 0, “error” = TRUE will be set.
status	Word	If one of the time constants <= 0, “status” = 16#8001.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = GAIN\left(1 + \frac{(tmLag2 - tmLag1)}{tmLag1} * e^{-\frac{t}{tmLag1}}\right);\ tmLag1 > 0$$

Figure 2-13: PDT1 element step response (*tmLag1/tmLag2 < 1*)



2.2.7 FB „LSim_I”

FB “LSim_I” simulates a simple integrating system.

Use case

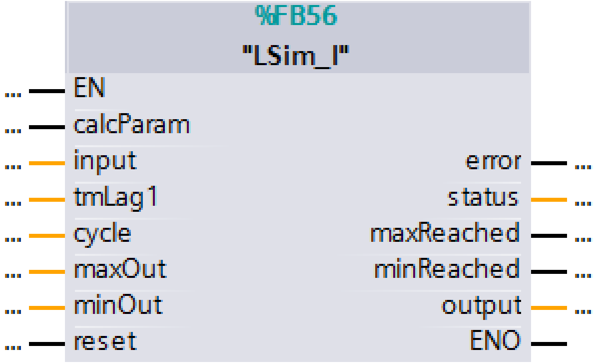
An I element can be used, for example, to simulate a fill level system (container).

Transfer function

$$F(p) = \frac{1}{(tmLag1 * p)}$$

Figure

Figure 2-14



Input parameters

Table 2-14

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when “tmLag1” or “cycle” changes)
input	Real	Input variable
tmLag1	Real	Integral action time (in seconds)
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, “output” = “maxOut” will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, “output” = “minOut” will be set)
reset	Bool	Resets all relevant parameters to ‘0’, including “output”.

Output parameters

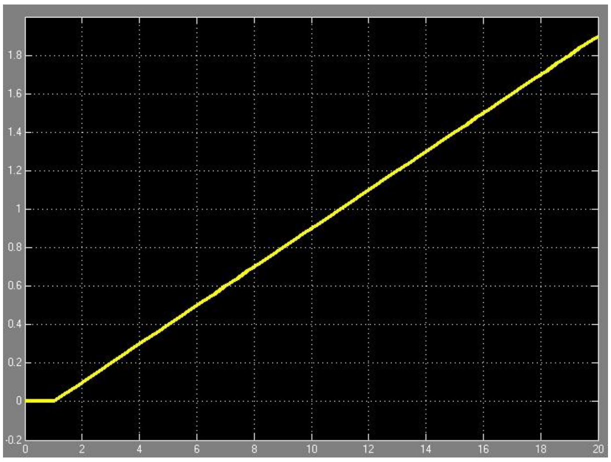
Table 2-15

Parameter	Data type	Description
error	Bool	If the integral action time <= 0, “error” = TRUE will be set.
status	Word	If the integral action time <= 0, “status” = 16#8001.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = \frac{t}{tmLag1}, \qquad t \geq 0$$

Figure 2-15: I element step response



2.2.8 FB „LSim_IT1"

FB "LSim_IT1" simulates a delayed integrator.

Use case

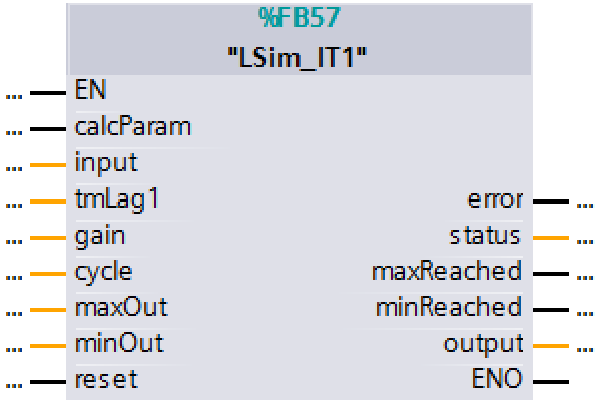
An IT1 element can be used, for example, to simulate a valve with servomotor.

Transfer function

$$F(p) = \frac{gain}{p * (tmLag1 * p + 1)}$$

Figure

Figure 2-16



Input parameters

Table 2-16

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when "tmLag1", "gain" or "cycle" changes)
input	Real	Input variable
tmLag1	Real	Integral action time (in seconds)
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, "output" = "maxOut" will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, "output" = "minOut" will be set)
reset	Bool	Resets all relevant parameters to '0', including "output".

Output parameters

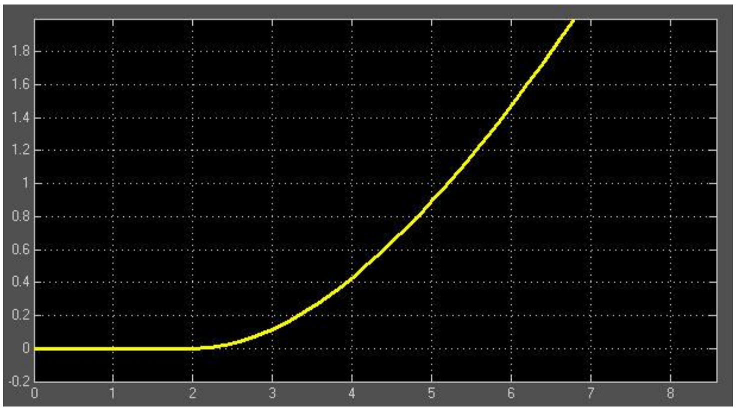
Table 2-17

Parameter	Data type	Description
error	Bool	If the integral action time <= 0, “error” = TRUE will be set.
status	Word	If the integral time <= 0, “status” = 16#8001.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = gain[t - tmLag1(1 - e^{\frac{-t}{tmLag1}})]$$

Figure 2-17: IT1 element step response



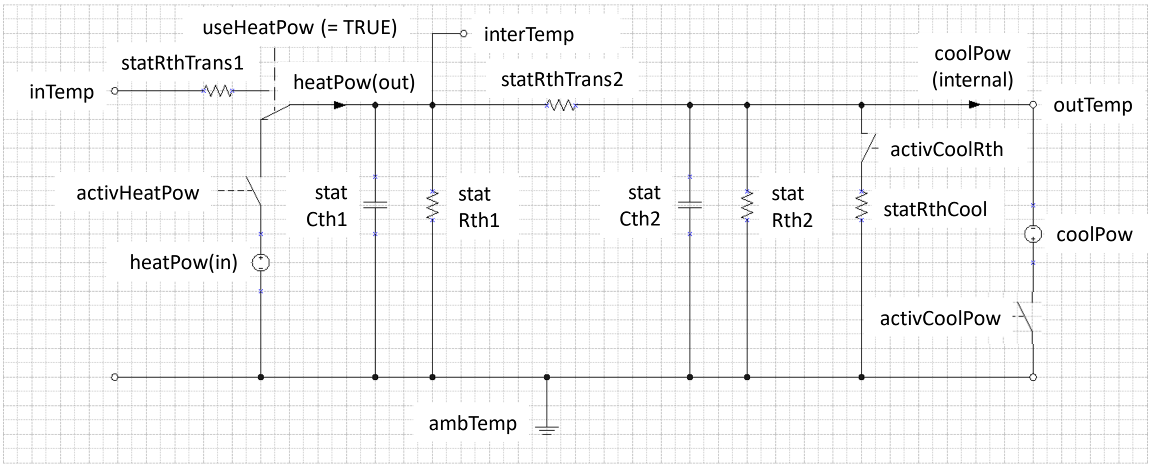
2.2.9 FB „LSim_TempProcess"

FB "LSim_TempProcess" simulates an asymmetrical temperature process that can be actively heated (using aids) and cooled (both actively and passively).

Connection diagram

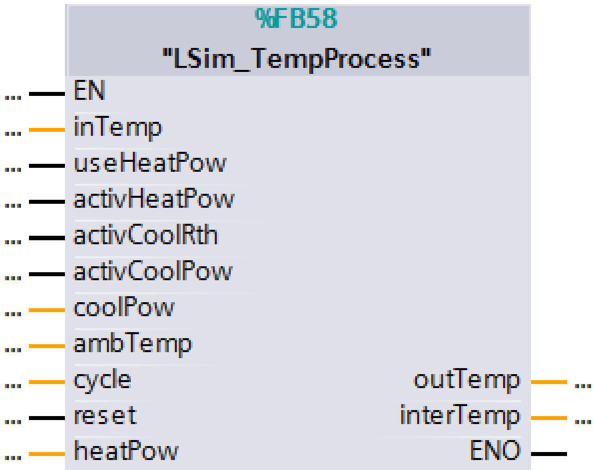
FB "LSim_TempProcess" uses the following equivalent connection diagram:

Figure 2-18



Figure

Figure 2-19



Parameters

Table 2-18

Parameter	Data type	Description
inTemp	IN: Real	Temperature input value (usable only when “useHeatPow” = FALSE)
useHeatPow	IN: Bool	FALSE: “inTemp” is used as the heat source TRUE: “heatPow” in conjunction with “activHeatPow” is used as the heat source.
activHeatPow	IN: Bool	Activate heating with “heatPow” (usable only when “useHeatPow” = TRUE)
activCoolRth	IN: Bool	Active cooling to ambient temperature via internal thermal resistance “statRthCool”
activCoolPow	IN: Bool	Active cooling via cooling power “coolPow” (even below ambient temperature)
coolPow	IN: Real	Cooling power as a positive value in W (active only when “activCoolPow” = TRUE)
ambTemp	IN: Real	Ambient temperature
cycle	IN: Real	Cycle time of the calling cyclic interrupt OB (in seconds)
reset	IN: Bool	Reset all relevant parameters (“outTemp” = “ambTemp”).
heatPow	INOUT: Real	Heating power in W (as an input when “useHeatPow” = TRUE and “activHeatPow” = TRUE / as an output when “useHeatPow” = FALSE)
outTemp	OUT: Real	Output temperature
interTemp	OUT: Real	Intermediate temperature (auxiliary quantity)
statRthTrans1	Static: Real	Input thermal resistance in K/W (active only when “useHeatPow” = FALSE)
statRth1	Static: Real	Thermal resistance parallel to thermal capacity 1 in K/W
statRthTrans2	Static: Real	Thermal resistance for heat flow between thermal capacity 1 and 2 in K/W
statRth2	Static: Real	Thermal resistance parallel to thermal capacity 2 in K/W
statRthCool	Static: Real	Thermal resistance for active cooling in K/W (active only when “activCoolRth” = TRUE)
statCth1	Static: Real	Thermal capacity 1 in Ws/K
statCth2	Static: Real	Thermal capacity 2 in Ws/K

Possible applications

The following applications can be implemented with FB “LSim_TempProcess” (heating and cooling can also be combined):

Table 2-19: Selection of heating options (1 out of 3)

Heating	**via power input “heatPow”**		**via temperature input “inTemp” (for an example, see Figure 2-20)**
Parameter	**analog**	**PWM signal**	
inTemp	irrelevant	irrelevant	variable input quantity
useHeatPow	TRUE	TRUE	FALSE
activHeatPow	TRUE	PWM signal	irrelevant
heatPow	variable input quantity	constant input quantity	output quantity

Table 2-20: Selection of cooling options with power setting (1 out of 2)

Cooling even below ambient temperature (with cooling power setting)		
Parameter	**analog**	**PWM signal**
activCoolPow	TRUE	PWM signal
coolPow	variable input quantity	constant input quantity

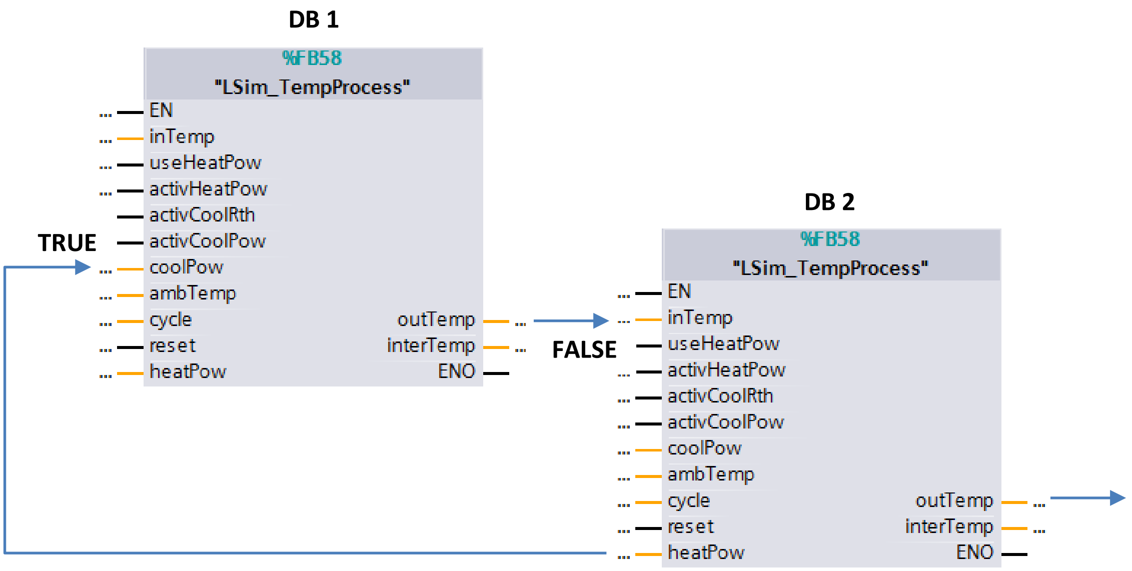
Table 2-21: Selection of cooling options via thermal resistance (coexistent)

Cooling to ambient temperature (via thermal resistance)		
Parameter	**Active**	**Passive**
Thermal resistance	statRthCool	statRth2
activCoolRth	TRUE	

Cascading

To map more complex processes, “LSim_TempProcess” can also be cascaded as follows:

Figure 2-20: Thermal influence of neighboring processes (cascade)



The first process is heated. The determined temperature “outTemp” of the first instance (DB1) is used as the input temperature “inTemp” of the second instance (DB2). This requires that input “useHeatPow” (DB2) be reset (FALSE). Therefore, the second process extracts energy from the first process. This means that the determined power “heatPow” of the second instance (DB2) must be returned to the first instance as cooling power “coolPow”. This requires that input “activCoolPow” (DB1) be set (TRUE).

Each process can be additionally cooled passively with “activCoolRth” via the thermal resistance. The second process, in turn, can be cooled actively via “coolPow” in conjunction with “activCoolPow” (for example, by the next instance).

2.2.10 FB „LSim_Lagging”

FB “LSim_Lagging” simulates a lag element.

Use case

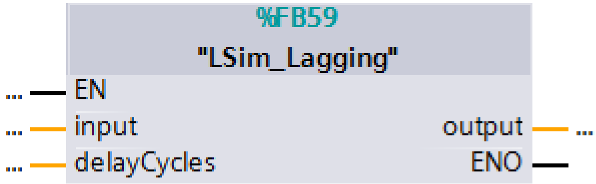
FB “LSim_Lagging” allows you to simulate, for example, conveyor systems.

Transfer function

$$F(p) = e^{-p*delayCycles*cycle}$$

Figure

Figure 2-21: lagging element



Parameters

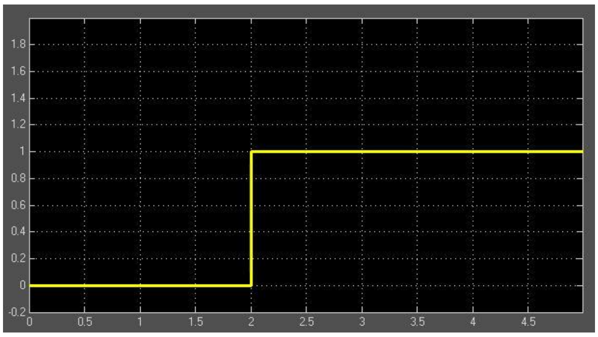
Table 2-22

Parameter	Data type	Description
input	IN: Real	Input variable
delayCycles	INOUT: UInt	Number of cycles by which the input signal is delayed (max. internal constant “MAX”; when exceeded, “delayCycles” = “MAX” will be set).
output	OUT: Real	Output variable
MAX	Constant: Int	Upper limit of the field for buffering the input signal (default: 100)

Step response

$$y(t) = u(t - delayCycles * cycle)$$

Figure 2-22: Lag element step response



2.2.11 FB „ LSim_DT1"

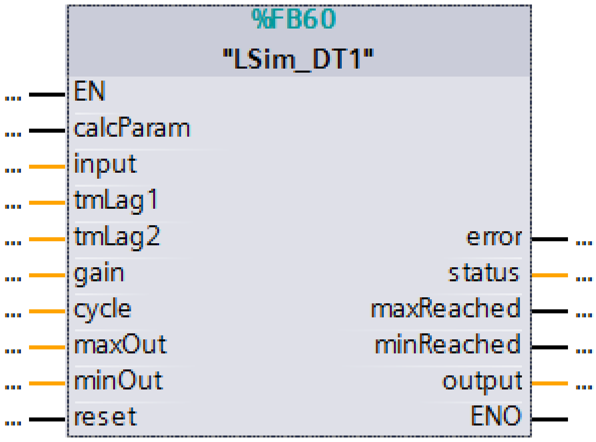
FB "LSim_DT1" simulates a delaying derivative action element.

Transfer function

$$F(p) = \frac{gain * tmLag2 * p}{(tmLag1 * p + 1)}$$

Figure

Figure 2-23



Input parameters

Table 2-23

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when "tmLag1", "tmLag2", "gain" or "cycle" changes)
input	Real	Input variable
tmLag1	Real	Derivative action time
tmLag2	Real	Delay time
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, "output" = "maxOut" will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, "output" = "minOut" will be set)
reset	Bool	Resets all relevant parameters to '0', including "output".

Output parameters

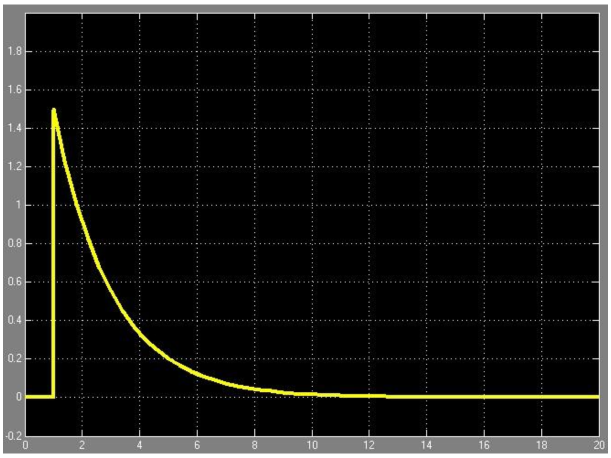
Table 2-24

Parameter	Data type	Description
error	Bool	If the derivative action time <= 0, “error” = TRUE will be set.
status	Word	If the derivative action time <= 0, “status” = 16#8001.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = \frac{tmLag2}{tmLag1} e^{-\frac{t}{tmLag1}}, \qquad t \geq 0$$

Figure 2-24: DT1 element step response



2.2.12 FB „LSim_AllPass1OrdReal"

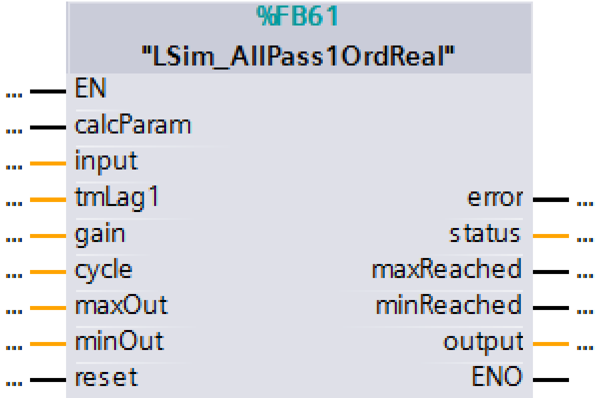
FB "LSim_AllPass1OrdReal" simulates a first-order all-pass with real zeros.

Transfer function

$$F(p) = \frac{gain * (1 - tmLag1 * p)}{(1 + tmLag1 * p)}$$

Figure

Figure 2-25



Input parameters

Table 2-25

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when "tmLag1", "gain" or "cycle" changes)
input	Real	Input variable
tmLag1	Real	Time constant
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, "output" = "maxOut" will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, "output" = "minOut" will be set)
reset	Bool	Resets all relevant parameters to '0', including "output".

Output parameters

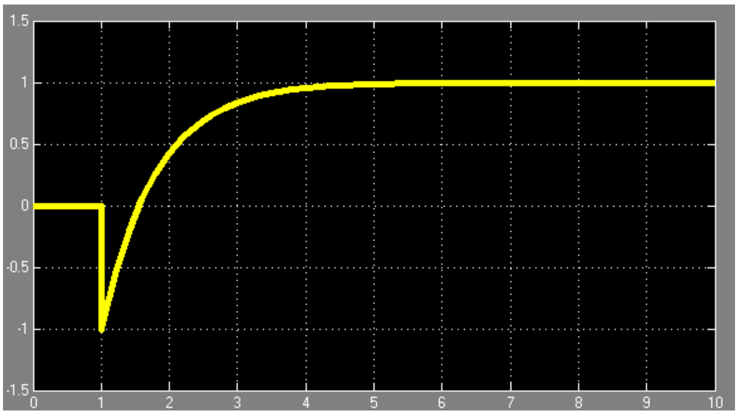
Table 2-26

Parameter	Data type	Description
error	Bool	If the time constant <= 0, “error” = TRUE will be set.
status	Word	If the time constant <= 0, “status” = 16#8001.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = gain * (1 - 2 * e^{\frac{t}{tmLag1}})$$

Figure 2-26



2.2.13 FB „LSim_AllPass2OrdReal"

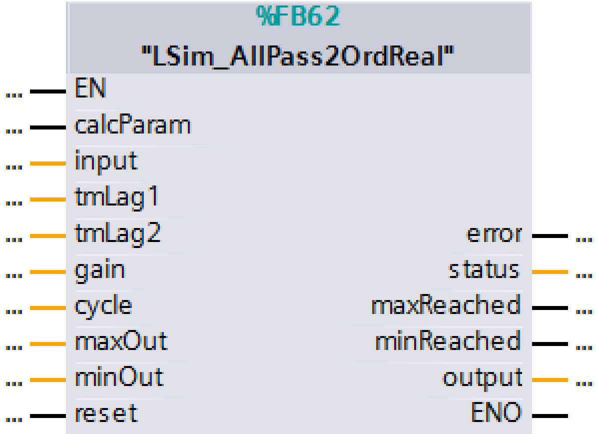
FB "LSim_AllPass2OrdReal" simulates a second-order all-pass with real zeros.

Transfer function

$$F(p) = \frac{gain * (1 - tmLag1 * p) * (1 - tmLag2 * p)}{(1 + tmLag1 * p) * (1 + tmLag2 * p)}$$

Figure

Figure 2-27



Input parameters

Table 2-27

Parameter	Data type	Description
calcParam	Bool	Recalculation of the internal parameters (activation required when "tmLag1", "tmLag2", "gain" or "cycle" changes)
input	Real	Input variable
tmLag1	Real	Time constant for the first PT1 element (in seconds)
tmLag2	Real	Time constant for the second PT1 element (in seconds)
gain	Real	Gain factor
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
maxOut	Real	Maximum limit of the output signal (when the signal is above the maximum limit, "output" = "maxOut" will be set)
minOut	Real	Minimum limit of the output signal (when the signal is below the minimum limit, "output" = "minOut" will be set)
reset	Bool	Resets all relevant parameters to '0', including "output".

Output parameters

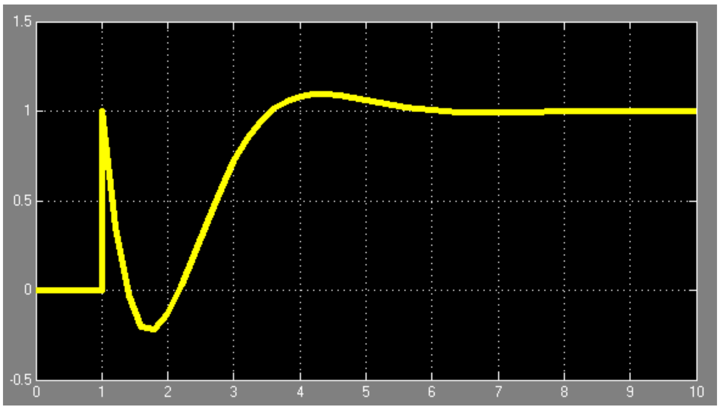
Table 2-28

Parameter	Data type	Description
error	Bool	If the time constants are identical or <= 0, “error” = TRUE will be set.
status	Word	If the time constants are identical or <= 0, “status” = 16#8001 will be output.
maxReached	Bool	When “maxReached” = TRUE, the “output” output variable was limited to the maximum value “maxOut”.
minReached	Bool	When “minReached” = TRUE, the “output” output variable was limited to the minimum value “minOut”.
output	Real	Output variable (valid only when “error” = FALSE)

Step response

$$y(t) = gain * (1 + K * e^{\frac{t}{tmLag1}} - K * e^{\frac{t}{tmLag1 \quad tmLag2}}), \text{ with } K = \frac{2*(tmLag1+tmLag2)}{tmLag2-tmLag1}$$

Figure 2-28

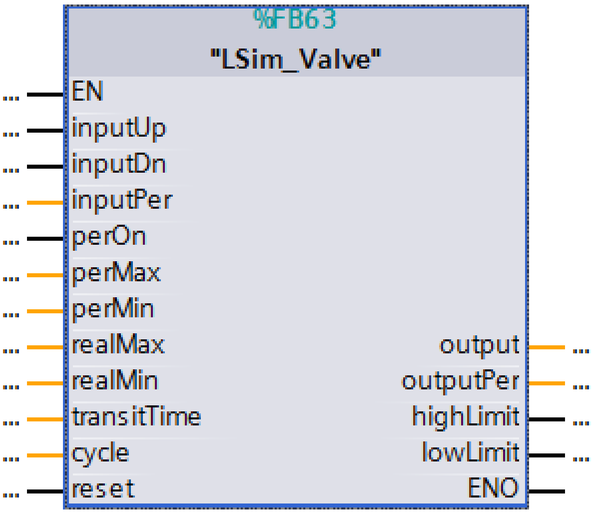


2.2.14 FB „LSim_Valve"

FB "LSim_Valve" simulates a valve with adjustable travel time.

Figure

Figure 2-29



Input parameters

Table 2-29

Parameter	Data type	Description
inputUp	Real	"Open valve" control signal
inputDn	Real	"Close valve" control signal
inputPer	Int	Analog manipulated variable
perOn	Bool	Selection switch to set whether "inputUp" / "inputDn" or "inputPer" will be used (when TRUE, "inputPer" will be used)
perMax	Real	Upper limit of the analog value of the valve position
perMin	Real	Lower limit of the analog value of the valve position
realMax	Real	Upper limit of the converted floating-point value for the valve position
realMin	Real	Lower limit of the converted floating-point value for the valve position
transitTime	Real	Travel time of the valve between the end positions
cycle	Real	Cycle time of the calling cyclic interrupt OB (in seconds)
reset	Bool	Resets "output" to "realMin"

Output parameters

Table 2-30

Parameter	Data type	Description
output	Real	Calculated valve position (within the "realMin" and "realMax" limits)
outputPer	Int	Calculated analog valve position (within the "perMin" and "perMax" limits)
highLimit	Bool	Simulated upper valve end position reached when "highLimit" = TRUE.
lowLimit	Bool	Simulated lower valve end position reached when "lowLimit" = TRUE.

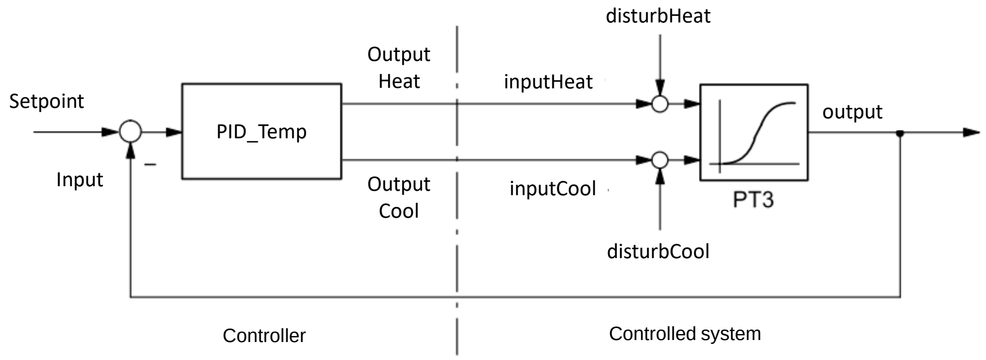
2.2.15 FB „LSim_PT3HeatCool“

FB “LSim_PT3HeatCool” simulates a PT3 temperature system.
The block has separate inputs for heating and cooling.

Use case

The block can be used to simulate a temperature system (for example, with the “PID_Temp” technology object).

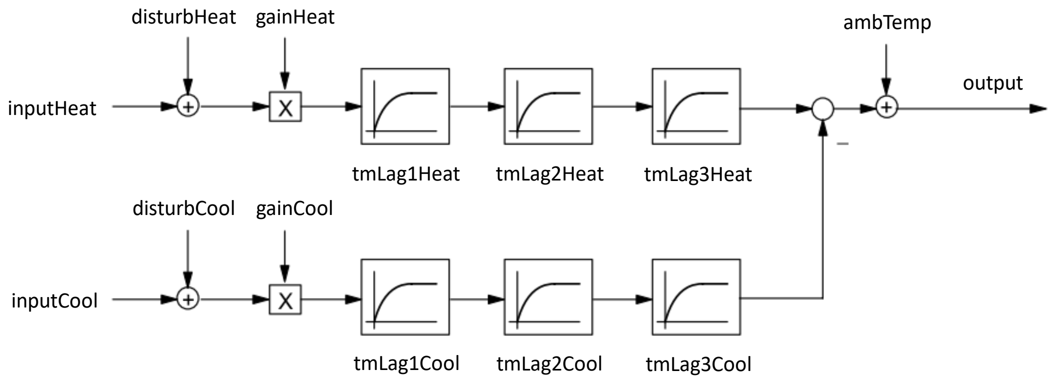
Figure 2-30: Sample control loop “PID_Temp” with “LSim_PT3HeatCool”



Connection diagram

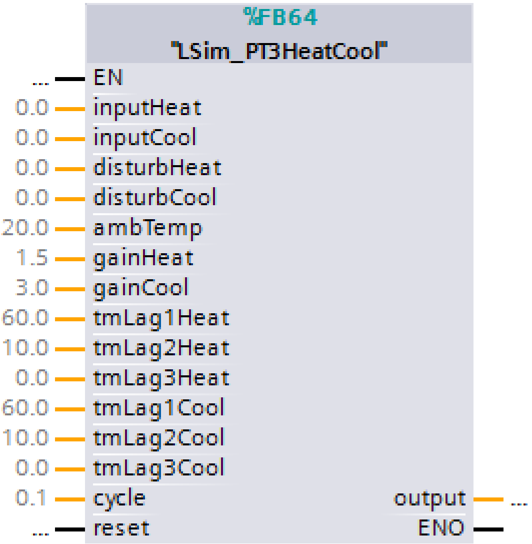
FB “LSim_PT3HeatCool” uses the following equivalent connection diagram:

Figure 2-31: Structure and parameters of the “LSim_PT3HeatCool” system block



Figure

Figure 2-32



Parameters

Table 2-31

Parameter	Data type	Description
inputHeat	IN: Real	Input variable (heat)
inputCool	IN: Real	Input variable (cool)
disturbHeat	IN: Real	Disturbance variable (heat); applied at the “inputHeat” input
disturbCool	IN: Real	Disturbance variable (cool); applied at the “inputCool” input
ambTemp	IN: Real	Ambient temperature
gainHeat	IN: Real	Gain factor (heat)
gainCool	IN: Real	Gain factor (cool)
tmLag(x)Heat	IN: Real	x-order lag factor (heat)
tmLag(x)Cool	IN: Real	x-order lag factor (cool)
cycle	IN: Real	Cycle time of the calling cyclic interrupt OB (in seconds)
reset	INOUT: Bool	Reset all relevant parameters (“output” = “ambTemp”); “reset” will be reset autonomously
output	OUT: Real	Output variable (valid only when “error” = FALSE)

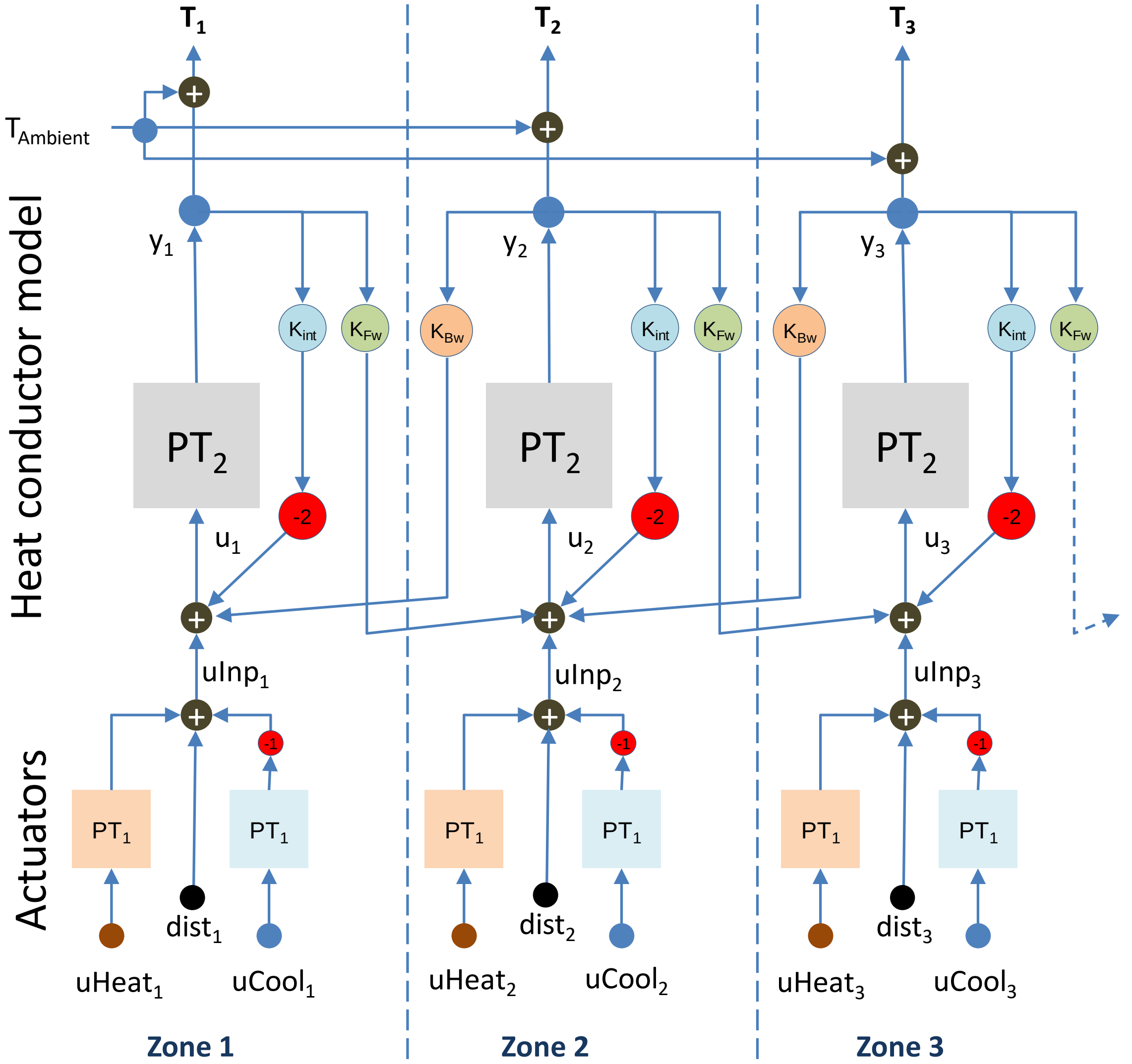
2.2.16 FB “LSim_Multizone”

The FB “LSim_Multizone” simulates the temperature curves of the coupled zones. It can be used for simulating the controlled system in multi-zone controlling with technology object “PID_Temp”. In this case each temperature zone is controlled by its own “PID_Temp” instance.

Simulation of a coupled thermal controlled system

As mathematic model, the discrete solution of a partial differential equation of a simple heat conductor with n grid points is used.

Figure 2-33: Flow chart of a coupled thermal controlled system



The dynamic of a gird point is exemplary as second-order aperiodic delay element. However, depending on the desired controlled system behavior, you can also use other dynamic elements with self-regulating here (in the simplest situation use a constant).

The simulated zone temperature T_z is the result of equation 2.2.16.1:

Equation 2.2.16.1: Calculating the zone temperature T_z:

$$T_z = y_z + T_{Ambient}$$

$T_{Ambient}$ = ambient temperature

z = Zone

Equation 2.2.16.2: Transfer function of the aperiodic PT2 delay element

$$y_z = \frac{gain_{PT2}}{(tmLag1_{PT2} * p + 1) * (tmLag2_{PT2} * p + 1)} * u_z$$

Equation 2.2.16.3: Equation for calculating the PT2 input

$$u_z = kFW_{z-1} * y_{z-1} - 2 * kINT_z * y_z + kBW_{z+1} * y_{z+1} + uInp_z$$

kFW = coupling factor forward

$kINT$ = coupling factor internal

kBW = coupling factor backward

Equation 2.2.16.4: Summation of the manipulated variable delay by PT1

$$uInp_z = \frac{gain_{Heat}}{tmLag1_{Heat} * p + 1} * uHeat_z - \frac{gain_{Cool}}{tmLag1_{Cool} * p + 1} * uCool_z + dist_z$$

$uHeat$ = manipulated variable heating

$uCool$ = manipulated variable cooling

$dist$ = disturbance variable

The model can be expanded to any number of zones by copying the pattern (Figure 2-33).

In equation 2.2.16.3 it has to be observed that for the peripheral zones, the previous zone or the following zone is omitted and the terms are therefore to be set to z-1 or z+1 here = 0.

NOTE To change the number of zones of the FB „LSim_Multizone“ you just have to adjust the local constant „MAX“ (at least 3).

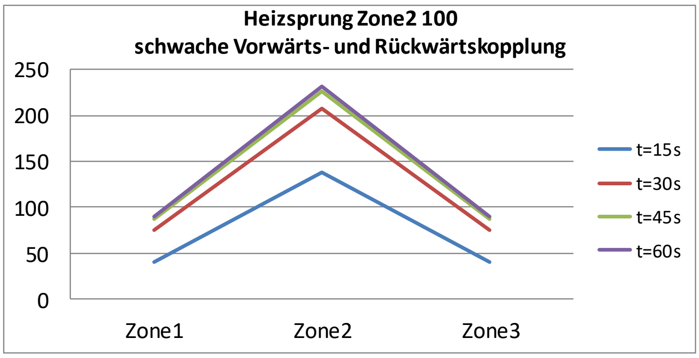
Through the coupling factors K_{FW} and K_{BW} (0 to 1), the thermal influence to neighboring zones can be varied.

You can vary the static end value of the zone temperatures through the internal coupling factors K_{int} (0 to 1) and the gain factors of the PTx delay elements.

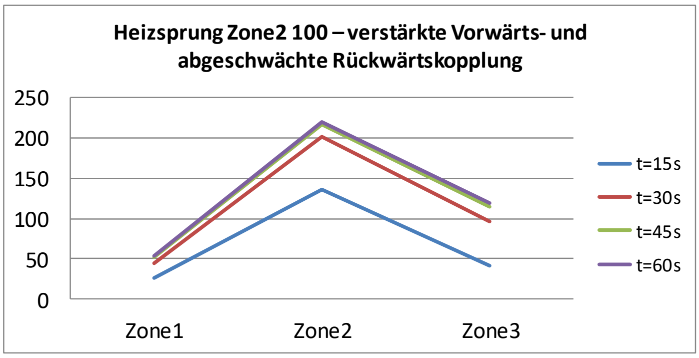
Chronological sequence

In terms of time, a test jump of 100% in zone 2, for example, is divided according to t seconds to the neighboring zones as follows:

Figure 2-34



$Gain_{PT2}$ = 5,0
K_{int} = 0,2
K_{FW} = 0,2
K_{BW} = 0,2



$Gain_{PT2}$ = 5,0
K_{int} = 0,2
K_{FW} = 0,3
K_{BW} = 0,1

Configuration explanations

The FB "LSim_Multizone" is called in the interrupt OB in which the controllers "PID_Temp" are also called.

Table 2-32: Parameters of "LSim_Multizone"

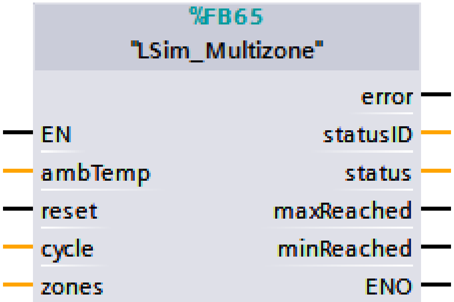
Name	P type	Data type	Comment
ambTemp	IN	Real	Ambient temperature $T_{Ambient}$
reset	IN	Bool	Resets all relevant parameters including "outputX" to '0'.
cycle	IN	Real	Cycle time of the calling cyclic interrupt OB in seconds
error	OUT	Bool	FALSE: No errors TRUE: Block error, "statusID" specifies the error source, "status" specifies the error code.
statusID	OUT	UInt	Error source: Ten digits specify the zone, One digit specifies the subfunction: 1 = instZxHeat (LSim_PT1) 2 = instZxCool (LSim_PT1) 3 = instZxPT2 (LSim_PT2aper)
status	OUT	Word	Error code of the respective subfunction FB „LSim_PT1" or FB „LSim_PT2aper"

Name	P type	Data type	Comment
maxReached	OUT	Bool	For “maxReached” = TRUE at least one output tag “output” of the subfunctions “LSim_PT1” or “LSim_PT2aper” has been limited by the respective static parameter “maxOut”.
minReached	OUT	Bool	For “minReached” = TRUE at least one output tag “output” of the subfunctions “LSim_PT1” or “LSim_PT2aper” has been limited by the appropriate static parameter “minOut”.
zones	IN_OUT	Array[1..#MAX] [1] of "LSim_typeZone"	PLC data type with the required interface parameters for each temperature zone (The size of the field is specified via the local constant “MAX”.)

Table 2-33: PLC data type „LSim_typeZone“

Name	Data type	Comment
inputHeat	Real	Manipulated variable heating (output value of „PID_Temp“)
inputCool	Real	Manipulated variable cooling (output value of „PID_Temp“)
disturbance	Real	Disturbance (applied at the input)
output	Real	Zone temperature (output value)

Figure 2-35



Configuring the FB “LSim_Multizone”

You can vary the thermal influence to neighboring zones via the default values of the coupling factors:

Table 2-34: PLC data type „LSim_typeZone“

Name	Data type	Default value	Comment
kInt	Real	0.2	Internal coupling factor
kFor	Real	0.3	Coupling factor to the next zone
kBack	Real	0.1	Coupling factor to the previous zone
uInp	Real	0.0	Sum $uInp_z$ (equation 2.2.16.4)
u	Real	0.0	Input u_z of aperiodic element PT2 (equation 2.2.16.3)
y	Real	0.0	Output y_z of aperiodic element PT2 (equation 2.2.16.2)

[1] The local constant “MAX” has to match the number of zones of the multi-zone controller.

You can set the coupling factors central via the default values of the PLC data type or just for each zone z via the default values of each element of the static array „statZ[z]“.

The controlled system properties of the simulated coupled temperature curves can be adjusted via the local constants of the FB “LSim_Multizone”.

Table 2-35

Name	Data type	Default value	Comment
MAX	DInt	3	Number of the zones
TMLAG_HEATER	Real	5.0	Time constant of the actuator delay heating (PT1) in seconds
TMLAG_COOLER	Real	10.0	Time constant of the actuator delay cooling (PT1) in seconds
GAIN_ZONE	Real	5.0	Gain factor temperature curve (PT2)
TMLAG1_ZONE	Real	20.0	Time constant 1 temperature curve (PT2)
TMLAG2_ZONE	Real	3.0	Time constant 2 temperature curve (PT2)
PT2_MAX_OUT	Real	1000.0	Maximum output limit temperature curve (PT2)
PT2_MIN_OUT	Real	-1000.0	Minimum output limit temperature curve (PT2)

For using the FB „LSim_Multizone“ you must copy the subroutines „LSim_PT1“ and „LSim_PT2aper“ and the PLC data types „LSim_typeDynState“ and „LSim_typeZone“ into your project.

NOTE

The FB „LSim_Multizone“ is used in the application example „Multi-Zone Control with "PID_Temp” for SIMATIC S7-1200/S7-1500“ (\8\).

3 Working with the Library

What will you find in this section?

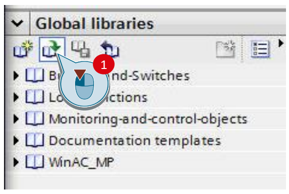
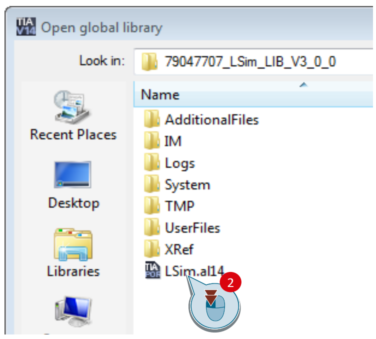
This chapter consists of instructions for integrating the „LSim“ library into your STEP 7 project and instructions for using the library blocks.

3.1 Integrating the library into STEP 7

The table below lists the steps for integrating the „LSim“ library into your STEP 7 project. Subsequently, you can use the blocks of the „LSim“ library.

Note The following section assumes that a STEP 7 (≥V14) project exists.

Table 3-1

No.	Action	Note
1.	Download the “79047707_LSim_LIB_V3_0_0.zip” file and extract the file on your engineering station.	
2.	In the right-hand pane, open the “Libraries” tab and click “Open global library”. Navigate to the storage location of the extracted folder and double-click to open the “LSim” file.	 

3.2 Integrating the library blocks into STEP 7

The table below lists the steps for integrating the blocks of the „LSim“ library into your STEP 7 program.

Table 3-2

No.	Action	Note
1.	Use drag and drop to drag the simulation blocks you want to use from the “Master copies” of the library to the program blocks folder of your user program. For using the FB „LSim_Multizone“ you must copy also the FBs „LSim_PT1“ and „LSim_PT2aper“ to your program blocks and the PLC data types „LSim_typeDynState“ and „LSim_typeZone“ into your project.	
2.	Now you can use the inserted blocks in your user program to simulate systems.	

3.3 Interconnecting controlling elements

Controlled system

As an example, we will implement a controlled system with the following controlling elements:

Figure 3-1 Controlled system

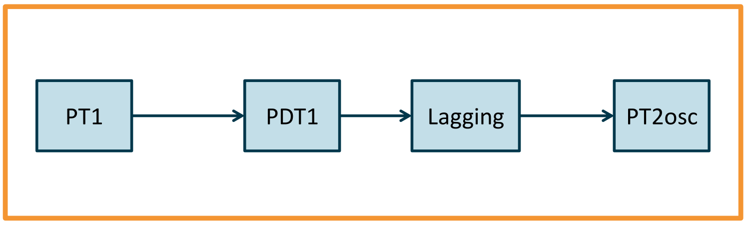
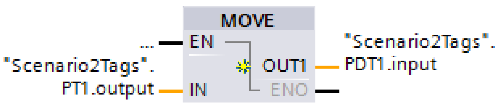


Table 3-3

No.	Action	Note
1.	Add the controlling elements you want to use as described in Table 3-2.In this example, these are the following controlling elements: • PT1 (FB “LSim_PT1”) • PDT1 (FB “LSim_PDT1”) • Lag element (FB “LSim_Lagging”) • PT2 in the periodic case (FB “LSim_PT2osc”)	
2.	Insert the function blocks into your cyclic interrupt OB and between each call, leave one network free for the MOVE commands.	
3.	Use MOVE commands to interconnect the outputs of the controlling elements with the input of the respective following controlling element.	
4.	Now you have set up a complete controlled system. **Note** If you want to monitor the controlled system, you can use the “Traces” function.	

4 Notes and Support

Overview

The “LSim” library is used to simulate controlled systems in STEP 7 (TIA Portal).

Examples

The documentation “Closed-Loop Control with PID_Compact V2.3” contains examples of how the blocks of the simulation library can be called and used.

The complete documentation, including the library and a sample project, can be downloaded at the following link:

https://support.industry.siemens.com/cs/ww/en/view/79047707

5 Related literature

Table 5-1

	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Download page of this entry https://support.industry.siemens.com/cs/ww/en/view/79047707
\3\	Function Manual: SIMATIC S7-1200, S7-1500 PID Control https://support.industry.siemens.com/cs/ww/en/view/108210036
\4\	Application example: „Single and Multi Loop Controller Structures (Cascade Control) with PID_Temp” https://support.industry.siemens.com/cs/ww/en/view/103526819
\5\	Application example: “3-Point Stepper Control with SIMATIC S7-1500” https://support.industry.siemens.com/cs/ww/en/view/68011827
\6\	Application example: PID Control with PID_Compact (S7-1200) https://support.industry.siemens.com/cs/ww/en/view/100746401
\7\	Application example "Library of general functions (LGF) for STEP 7 (TIA Portal) and S7-1200 / S7-1500" https://support.industry.siemens.com/cs/ww/en/view/109479728
\8\	Application example "Multi-Zone Control with "PID_Temp” for SIMATIC S7-1200/S7-1500" https://support.industry.siemens.com/cs/ww/en/view/109740463

6 History

Table 6-1

Version	Date	Modifications
V1.0	08/2013	First version
V1.1	12/2014	“LSim_PT3HeatCool” block added
V2.0	03/2016	Review and publication for STEP 7 V13 SP1
V2.0.1	03/2017	Textual correction in FB „LSim_TempProcess”
V3.0.0	03/2017	Release for STEP 7 V14