S7-200 – Three-Step Control

service & SUPPORT

Micro Automation - Three-Step Control

SIEMENS

FAQ

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 correctly used. These application examples do not relieve you of the responsibility to use sound practices in application, installation, operation and maintenance. By using these application examples you accept that Siemens is not liable for any damages except for those specified in the above liability clause. 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.

Warranty, Liability and Support

We accept no liability for 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"). However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health. The above provisions do not imply a change in the burden of proof to your detriment.

For questions about this document please use the following e-mail address:

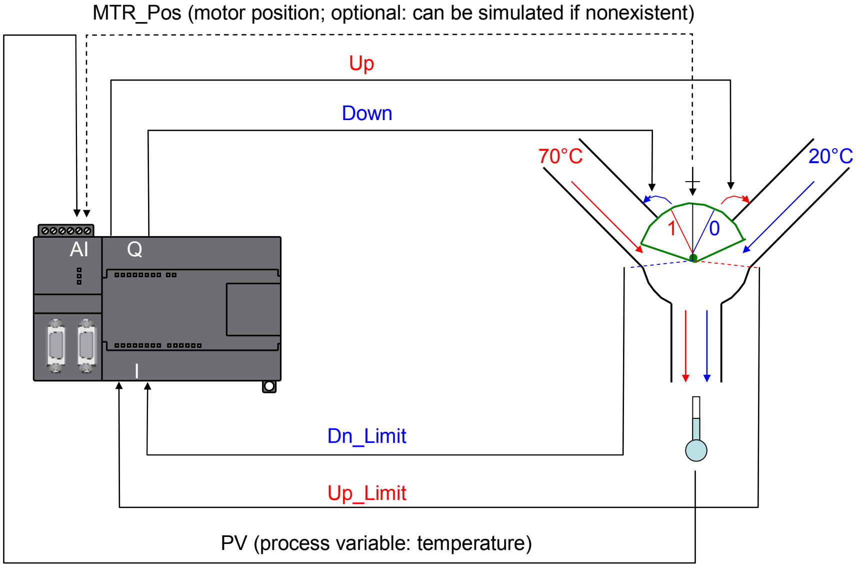
mailto:csweb@ad.siemens.de

Table of Contents

1 Objective

The valve of a mixing tap is to be controlled according to a desired temperature set point.

Figure 1-1



To do so, a three-way mixing valve with valve-actuating is used. Hot water (e.g. 70°C) runs through one inlet and cold water (20°C) runs through the other one. By varying the water feed between the two preset temperatures via the valve position, the effluent water can obtain any temperature ranging from 20°C to 70°C.

This is a **three-step control**:

1. The valve is set towards “more hot water” (digital input “Up”)

   or

2. the valve is set towards “more cold water” (digital input “Down”)

   or

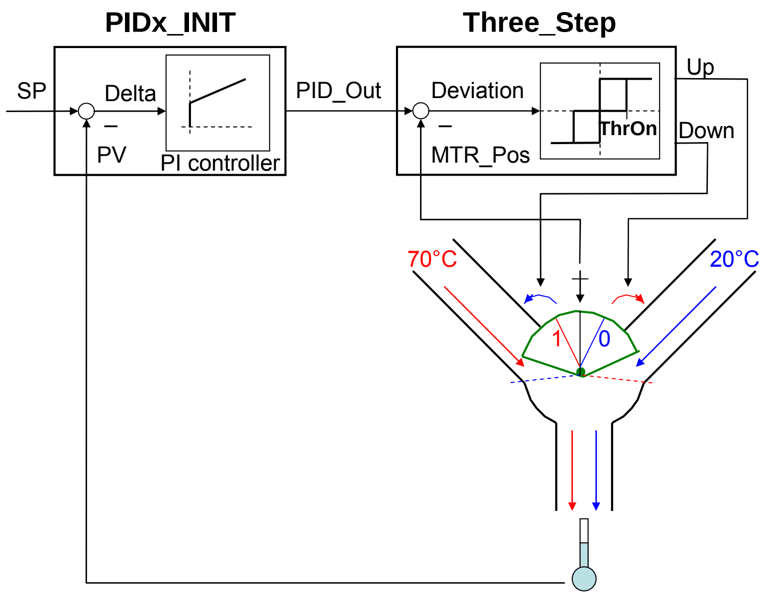
3. the valve is not selected at all.

For this close-loop control the S7-200 CPU (as of release V2.0) requires the temperature-process variable (analog input “PV”). Additionally the valve motor position is required. The position can either be obtained using the analog feedback signal (analog input “MTR_Pos”) or by a simulation. The valve limit switches “Up_Limit” and “Dn_Limit” signalize the end positions of the valve and are used as control limits and as reference points for the valve position simulation.

2 Functionality

The three-step control is implemented using the PID-controller integrated in STEP 7-Micro/WIN (as of version 4.0) and the library „Three_Step". The functionality is explained using the following figure:

Figure 2-1



Method of Function

The temperature set point ("SP") is compared to the process variable ("PV") and the control deviation („Delta" = „SP" – „PV") is generated in the PID-control block "PIDx_INIT". The control with PI-action ("PI controller" necessary for this closed loop control) puts the control variable "PID_Out" out. It provides the set point for the three-step control element "Three_Step". "PID_Out" is compared with the current valve motor position "MTR_Pos" and the "Deviation" is calculated.

Hysteresis

If the "Deviation" exceeds the preset threshold value "ThrOn" the valve is opened towards "more hot water" until the deviation equals zero or is less than zero. If the "Deviation" is smaller than the preset negative threshold value the valve is opened towards "more cold water" until the deviation equals zero or is greater than zero.

2.1 Correction control in stationary mode

As the set point temperature is never exactly obtained due to process behaviour between control element (valve) and temperature monitoring point and the integral part of the PI-controller put out every slight control deviation in form of an summed actuating variable, the control element deviation will exceed the tolerance range +/- “ThrOn” and therefore counteract even in stationary mode.

Figure 2-2

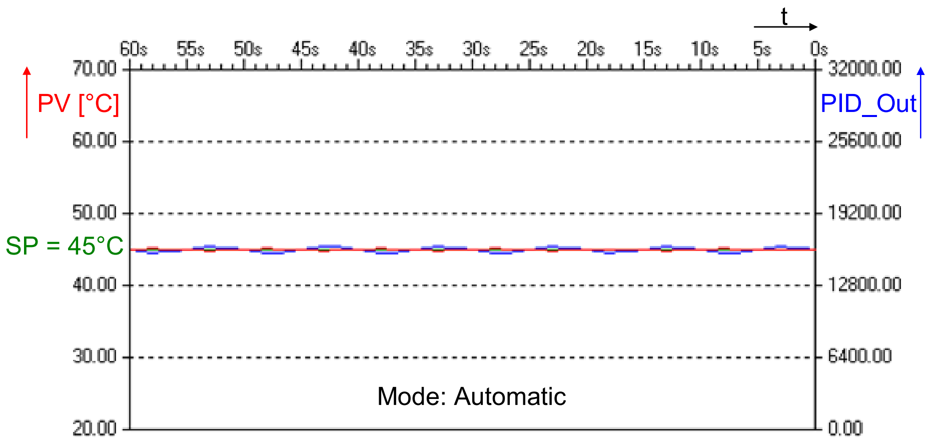
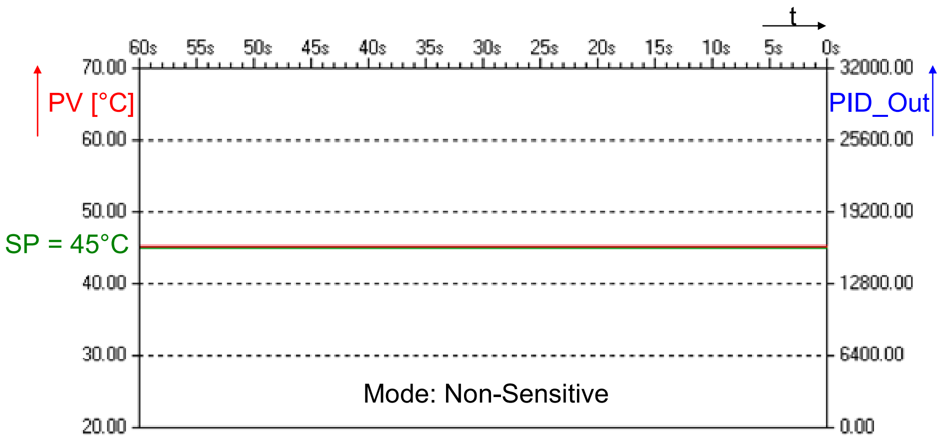


Figure 2-2 shows in what way the process variable “PV“ oscillates around the set point “SP” in stationary state without being influenced in automatic mode (particularly distinguishable at the control variable “PID_Out”).

Switch off integral part

In order to avoid this unwanted oscillation around the temperature set point, the block “Non_Sensitive”is used. It “freezes” the controller in a specified range and, in doing so, switches off the I-part of the controller.

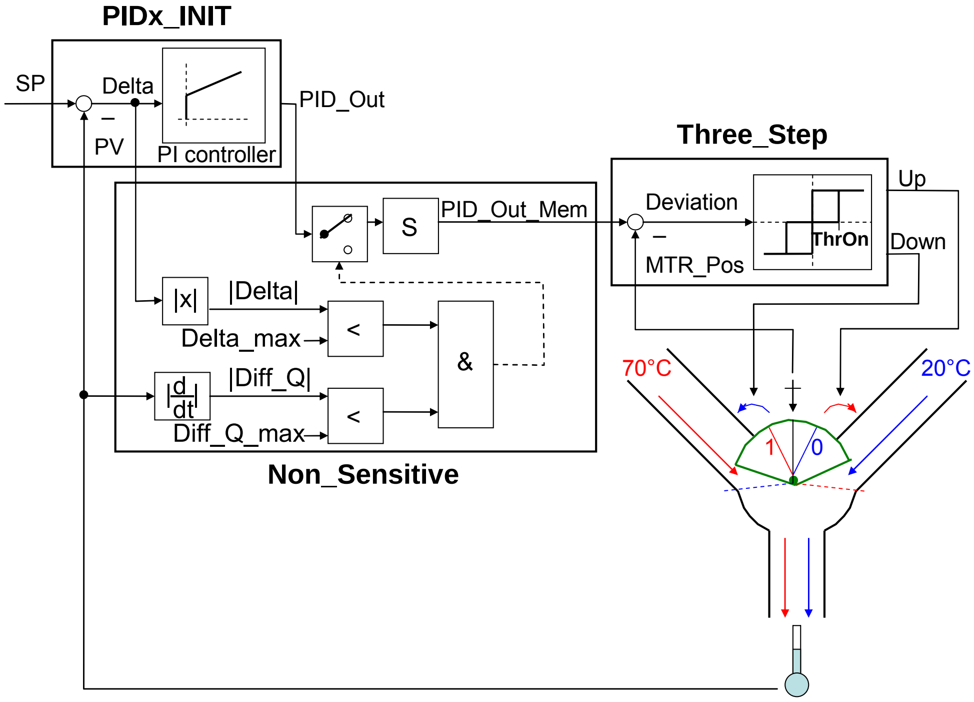
Figure 2-3



Unlike figure 2-3, figure 2-5 shows no activity of the valve motor in stationary state in the “Non_Sensitive” mode.

2.2 Operation principle "Non_Sensitive"

Figure 2-4



The control variable “PID_Out“ is temporarily stored by the block “Non_Sensitive“ (“PID_Out_Mem“). If the absolute control deviation “|Delta|“ stays within the specified maximum deviation “Delta_max“ and if the absolute difference quotient of the process variable “|Diff_Q|“ is smaller than a specified maximum increase “Diff_Q_max“, the last stored control variable value “PID_Out_Mem“ is adopted as the set point for the three-step control element until the conditions for the ”Non-Sensitive“-mode are not met anylonger and the controller adopts the control value settings.

Figure 2-5

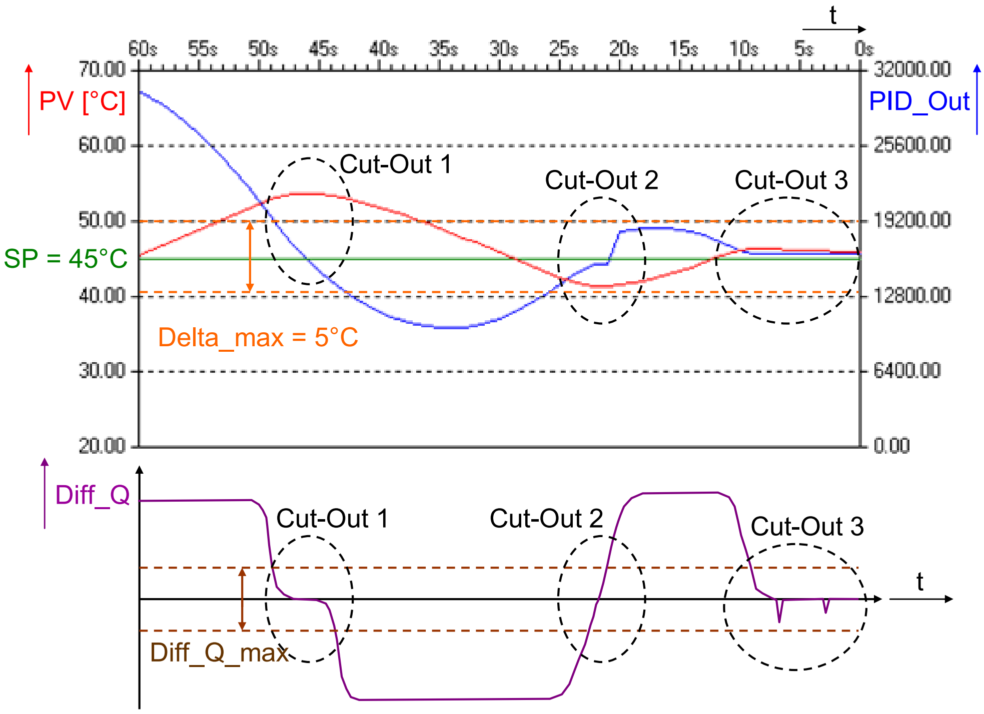


Figure 2-5 shows the transient for a specified set point of 45°C (green) and for a maximum allowable control deviation of +/- 5°C (orange) for the dead band. In addition to the control deviation verification, the process variable temperature increase must stay within the tolerance band +/- “Diff_Q_max” (brown).

In cut-out 1 the temperature increase meets this condition, but the temperature is not within the deviation tolerance band “SP” +/- “Delta_max”. The control variable is still determined by the PI-controller.

In cut-out 2 the temperature is within the tolerance band and the temperature increase is within +/- “Diff_Q_max”. The control variable “PID_Out“ is stored until the process variable temperature increase gets too high and the control variable is calculated by the PI-controller again.

In cut-out 3 the conditions for the Non-Sensitive mode are met for “Delta” and for “Diff_Q”. The current setting is “frozen” and this time the temperature stays within the preset tolerance.

2.2.1 Parameter preset

This means that for an optimum loop control without activity in the stationary state the control parameters specifications and the tolerance band specifications are of crucial importance. The gain and the integral time have to be adjusted to the controlled system for the purpose of a quick transient, if possible without an overshooting.

The tolerance band values have to be determined empirically depending on the controlled system and the control parameters assignment.

The block “Non_Sensitive“ offers the opportunity to monitor the current absolute deviations “Delta” and “Diff_Q” during operation and, in doing so, to adjust the limit values “Delta_max” and “Diff_Q_max” accordingly. The same can be done for the maximum control element deviation “ThrOn” using the signed parameter “Deviation” of the block “Three_Step”.

2.3 Three-Step control without valve position feedback

If the valve used does not possess a valve position feedback, the block “Valve_Sim” is used to simulate the valve position.

Figure 2-6

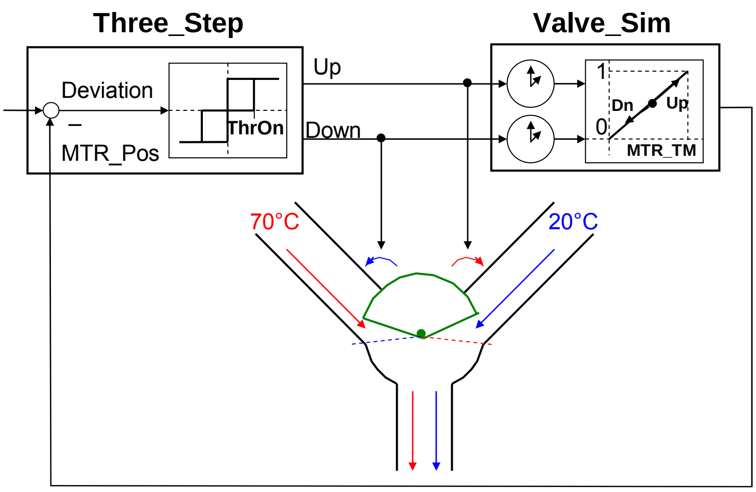


Figure 2-6 shows the altered part of the block diagram from figure 2-4 in order to simulate the valve position. The time needed for the activation the valve control signals “Up” and “Down” is measured and, in relation to the entire traversing time “MTR_TM” (time the motor needs to travel from one end position to the other one) the standardized position angle covered is summed up to the current position (for “Up” added, for “Down” subtracted). The new position evaluated this way is used as initial position for the next calculation cycle and is assigned to the three-step control element “Three_Step” in form of a position feedback “MTR_Pos”.

2.4 Simulation of the controlled system

In order to easily check the three-step control the attached download additionally possesses the simulation library “SYS_Sim”. Here a dead time element (“Dead_Time“) and a low pass filter (“Low_Pass”) can be found. Using these blocks you can emulate the controlled system between the valve control element and the process variable temperature sensor. This way you can check the functioning of the three-step control even without any hardware.

Figure 2-7

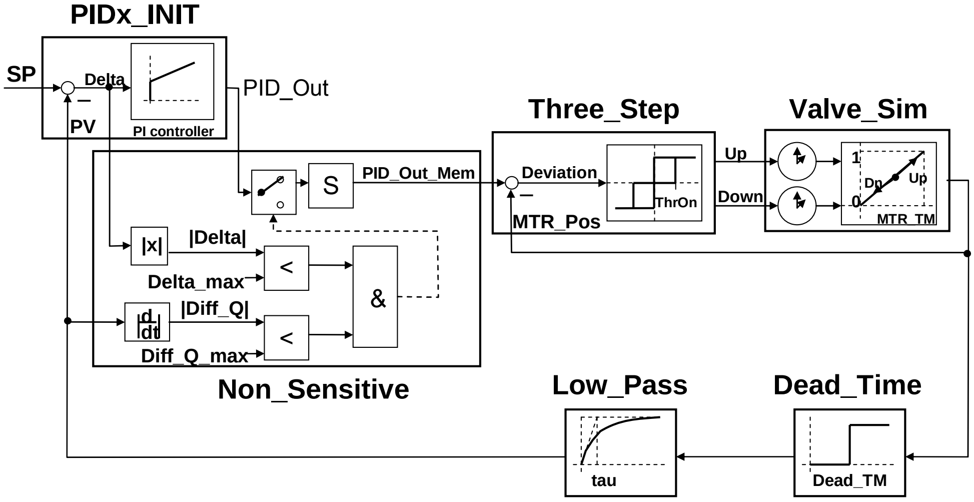
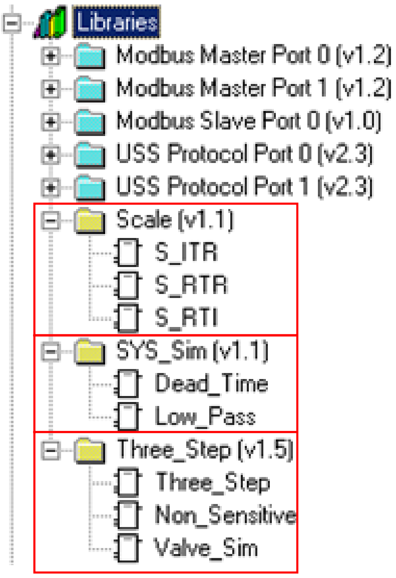


Figure 2-7 shows the block diagram of the three-step control with simulation of the valve position and of the controlled system. Besides the feedback of the simulated valve motor position to the three-step control element (“MTR_Pos”) the process variable is generated using the dead time element “Dead_Time” and the low pass filter “Low_Pass”. This means that the control system behaviour is determined by the dead time “Dead_TM” and by the time constant “tau”.

3 The Libraries

The libraries "Three_Step" and "SYS_Sim" are provided in the attached download. The link to download the conversion library "Scale" can be obtained on the same HTM-page the present document is obtained from.

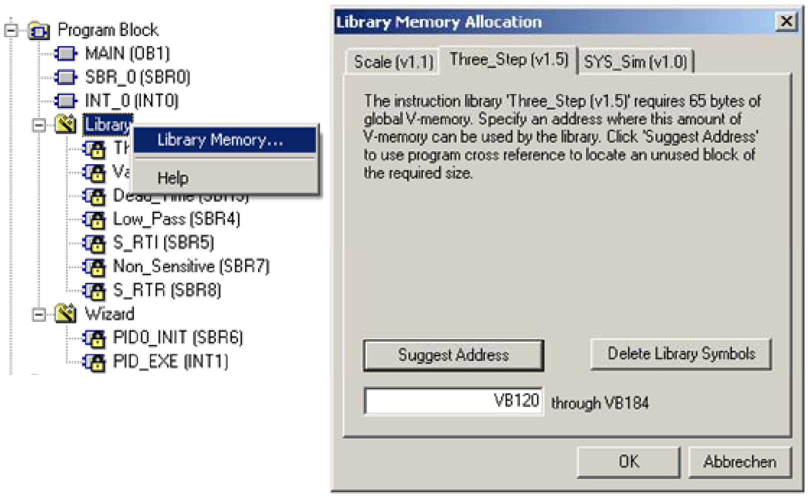
Figure 3-1



3.1 Library "Three_Step"

In order to use the "Three_Step" library blocks, 65 bytes of memory must be reserved for the global variables.

Figure 3-2



3.1.1 Block "Three_Step"

The block "Three_Step" converts the control variable "PIDx_Output" provided by the PID-controller “PIDx_INIT” into a three-step signal.

Figure 3-3

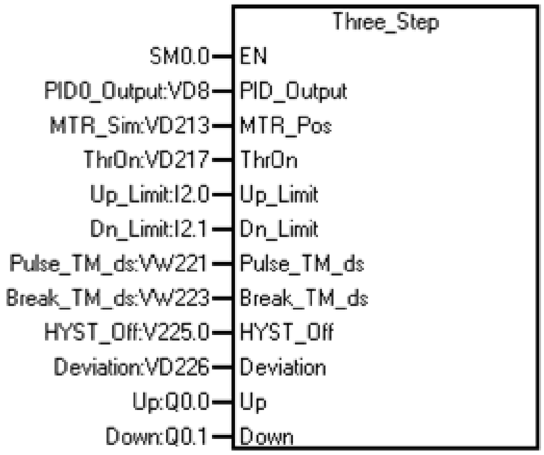


Table 3-1

Symbol	Type of variable	Data Type	Comment	Range of values
EN	IN	BOOL	Activation	
PID_Output	IN	REAL	PID-control variable (standardized)	0.0 – 1.0
MTR_Pos	IN	REAL	Valve position (standardized)	0.0 – 1.0
ThrOn	IN	REAL	Threshold parameter	0.0 – 1.0
Up_Limit	IN	BOOL	Upper valve limit switch	
Dn_Limit	IN	BOOL	Lower valve limit switch	
Pulse_TM_ds	IN	WORD	Minimum pulse time in 0.1 s	
Break_TM_ds	IN	WORD	Minimum break time in 0.1 s	
HYST_0ff	IN	BOOL	Hysteresis switch off	
Deviation	IN_OUT	REAL	= PID_Output – MTR_Pos	-1.0 – 1.0
Up	IN_OUT	BOOL	Valve actuation “UP”	
Down	IN_OUT	BOOL	Valve actuation “DOWN”	

PID_Output

- Control variable of the PID-controller. This signal is fed with global variables of the library of the PID-controller.

Pulse_TM_ds

- determines the minimum activation time of a valve actuation in deciseconds (0.1 secs). The valve is actuated for this period eventhough the conditions for its activation are not met anylonger during operation.

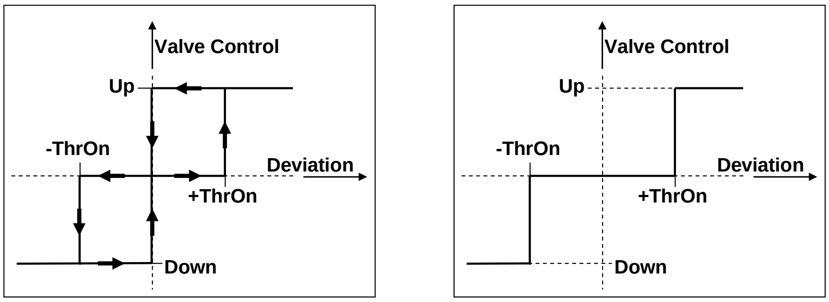
Break_TM_ds

- determines the minimum break time between two valve actuations in deciseconds (0.1 secs). This is the time that elapses before the valve is controlled again, eventhough if the conditions for its activation are met prior to the operation.

HYST_Off

- used to switch off the hysteresis. This bit determines the stop-condition of the valve control.

Figure 3-4



0 1

Figure 3-4 shows the valve control depending on the “Deviation” for “HYST_Off”-states “0” and ”1”:

- In state 0 the control is stopped when the “Deviation”-zero crossing (equals “MTR_Pos” = “PID_Output”) is attained.
- In state 1 the control is stopped when the “ThrOn”-tolerance band is entered (equals “MTR_Pos” < “PID_Output” +/- “ThrOn”).

This means that the control activity of the valve is minimized when the hysteresis is switched off as the temperature change is carried out time-delayed due to the controlled system. Therefore it is more likely that the temperature stays within the neutral range.

3.1.2 Block "Non_Sensitive"

The block “Non_Sensitive”allows to “freeze” the PID-controller in a specified area.

Figure 3-5

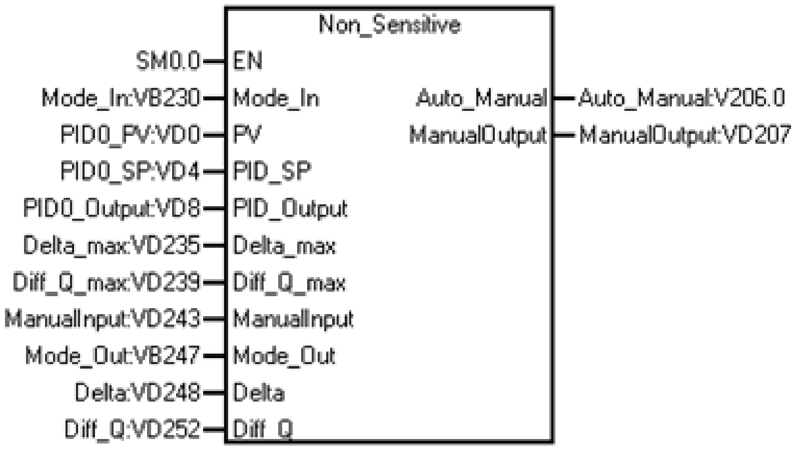


Table 3-2

Symbol	Type of variable	Data Type	Comment	Range of values
EN	IN	BOOL	Activation	
Mode_In	IN	BYTE	Selector switch for operation mode	0=M; 1=A 2=NS
PV	IN	REAL	Process variable (standardized)	0.0 – 1.0
PID_SP	IN	REAL	PID-set point (standardized)	0.0 – 1.0
PID_Output	IN	REAL	PID-control variable (standardized)	0.0 – 1.0
Delta_max	IN	REAL	Maximum absolute control deviation for NS	0.0 – 1.0
Diff_Q_max	IN	REAL	Maximum absolute process variable increase for NS	
ManualInput	IN	REAL	Control variable parameter for M	0.0 – 1.0
Mode_0ut	IN_OUT	BYTE	Current operation mode	0=M; 1=A 2=NS
Delta	IN_OUT	REAL	Absolute control deviation	0.0 – 1.0
Diff_Q	IN_OUT	REAL	Absolute process variable increase	
Auto_Manual	OUT	BOOL	Mode output for “PIDx_INIT“	0=M; 1=A
ManualOutput	OUT	REAL	Control variable parameter (M) for “PIDx_INIT” (standardized)	0.0 – 1.0

Mode_In / Mode_Out

"Mode_In" specifies the desired mode (0 = manual; 1 = automatic; 2 = Non-Sensitive). Values that are not within the domain of definition are not adopted. This can be checked using the byte "Mode_Out": For an undefined parameter in "Mode_In" the present mode is maintained in "Mode_Out".

PID_SP / PID_Output

"PID_SP" and "PID_Output" are fed with global variables of the library of the PID-controller.

PV

- is the standardized process variable and it is generated using the "Scale"-block "S_ITR" of the analog temperature sensor signal (see chapter 3.3).

WARNING	**The standardized process variable "PV" must not be fed with the global variables of the library of the PID-controller as it is only updated in the specified interval "PIDx_SampleTime". Otherwise the condition for the increase check between these samples would be fulfilled by mistake.**

Delta_max / Diff_Q_max / Delta / Diff_Q

"Delta_max" and "Diff_Q_max" determine the switch-over conditions for the "Non-Sensitive" mode. They can empirically be determined using the smallest occurring values of "Delta" and of "Diff_Q" (see chapter 2.3.1).

ManualInput

"ManualInput" is the standardized control variable which is available at the output "ManualOutput" if the "Manual" mode is selected.

ManualOutput / Auto_Manual

- determine the same-named input parameters for the PID-control block "PIDx_INIT" (see figure 4-1).

3.1.3 Block "Valve_Sim"

The block "Valve_Sim" is used to simulate the valve position.

Figure 3-6

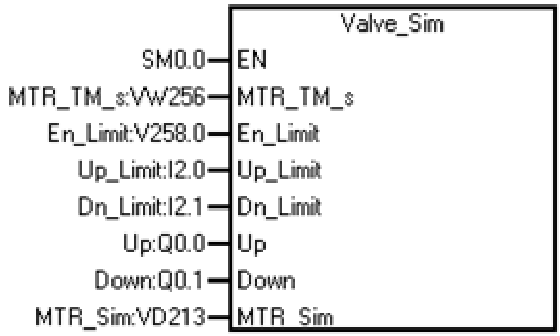


Table 3-3

Symbol	Type of variable	Data Type	Comment	Range of values
EN	IN	BOOL	Activation	
MTR_TM_s	IN	WORD	Entire motor traversing time [s]	
En_Limit	IN	BOOL	Enable correction	
Up_Limit	IN	BOOL	Upper valve limit switch	
Dn_Limit	IN	BOOL	Lower valve limit switch	
Up	IN	BOOL	Valve actuation “UP”	
Down	IN	BOOL	Valve actuation “DOWN”	
MTR_Sim	IN_OUT	REAL	Simulated standardized valve position	0.0 – 1.0

MTR_TM_s

- determines the traversing time between the two end positions of the valve in integer seconds.

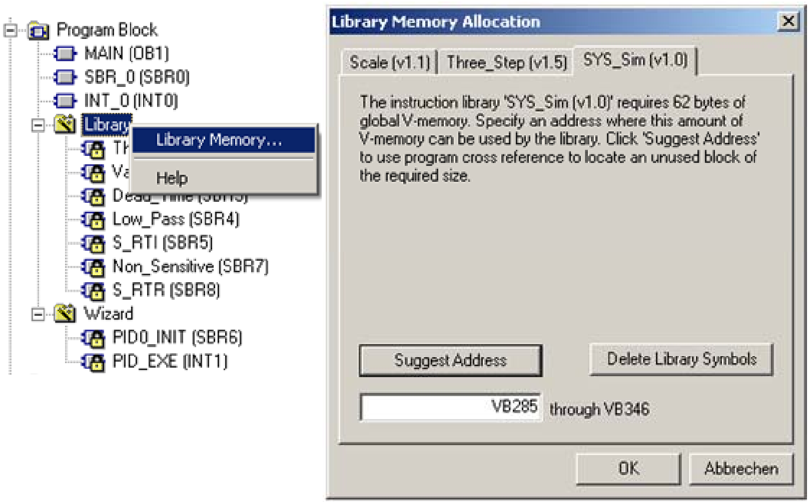
En_Limit

- allows to correct the estimated valve position, if the time “MTR_TM_s” was chosen too short. On the activation the simulated valve position "MTR_Sim" jumps back to 0.98 when value 1 is attained as long as “Up_Limit” is not reached, and to 0.02 when value 0 is reached as long as “Dn_Limit” is not reached.

3.2 Library "SYS_Sim"

In order to use the "SYS_Sim" library blocks, 62 bytes of memory must be reserved for the global variables.

Figure 3-7



3.2.1 Block "Dead_Time"

The block “Dead_Time” allows the time-delayed output “OUT” of the input signal “IN”.

Figure 3-8

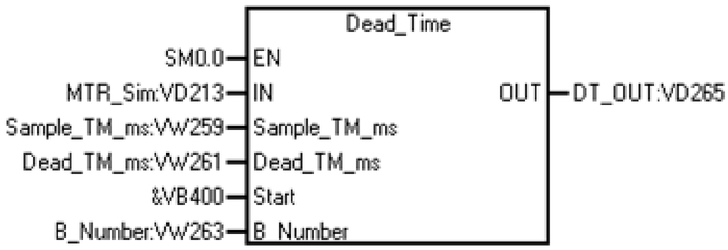


Table 3-4

Symbol	Type of variable	Data Type	Comment	Range of values
EN	IN	BOOL	Activation	
IN	IN	REAL	Standardized input signal	0.0 – 1.0
Sample_TM_ms	IN	WORD	Sampling time in milliseconds	
Dead_TM_ms	IN	WORD	Dead time in milliseconds	
Start	IN	DWORD	Memory start address	
B_Number	IN_OUT	INT	Memory size (byte)	
OUT	OUT	REAL	Standardized output signal	0.0 – 1.0

Sample_TM_ms

- must not be smaller than the cycle time (SMW 22). Times that are smaller than the cycle time are not adopted (can be checked using the global variable “Sample_Time”). The “Dead_Time“-sampling time should not be greater than PID_controller sampling time “PIDx_SampleTime” (better: Sample_TM_ms = 0,5 * PIDx_SampleTime/1000; PIDx_SampleTime is indicated in seconds).

Dead_TM_ms

- must only be an integer multiple of the sampling time “Sample_TM_ms”. A maximum dead time of 65.535 seconds can be selected.

Start / B_Number

“Start“ determines the pointer to the start address of the selected buffer (e.g. “&VB0”).

The buffer size in bytes is calculated as follows:

B_Number = 4 * Dead_TM_ms / Sample_TM_ms

WARNING **Make sure the selected storage area exists and is not reserved.**

3.2.2 Block "Low_Pass"

The block “Low_Pass” simulates the function of a low pass filter.

Figure 3-9

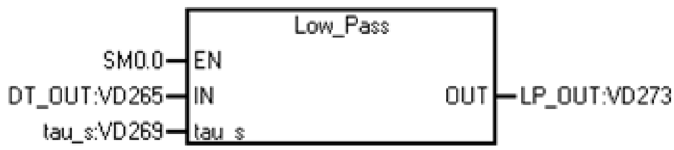


Table 3-5

Symbol	Type of variable	Data Type	Comment	Range of values
EN	IN	BOOL	Activation	
IN	IN	REAL	Standardized input signal	0.0 – 1.0
tau_s	IN	REAL	Time constant in seconds	
OUT	OUT	REAL	Standardized output signal	0.0 – 1.0

tau_s

- must be greater than zero. The smallest possible time constant can be 1*e^(-37) seconds (can be checked using the global variable “tau”).

3.3 Conversion library “Scale”

The controller “PIDx_INIT” works with three different value ranges:

1. the actual physical value range (in this case: the temperature range from 20 to 70 °C)
2. the standardized parameters in the value ranges from 0.0 to 0.1 and
3. the analog input parameters of the temperature sensor (adjustable with the PID-wizard in Micro/WIN) and of the value position feedback (both in INTEGER format)

For the conversion between these value ranges you also need the library “Scale“.

The link to download the conversion library "Scale" can be obtained on the same HTM-page the present document is obtained from.

Further information on the application of these library blocks can also be found in this link.

4 Exemplary projects

In addition to the libraries “three_step.mwl” and “sys_sim.mwl” the attached download contains the three following exemplary projects concerning the three-step control:

Table 4-1

Name	Function	Chapter
Three_Step_Sim.mwp	Only simulation (valve position simulation and path simulation)	4.1
Three_Step.mwp	For valves with position feedback	4.2
Three_Step_NFB.mwp	For valves without position feedback	4.3

These are used either as templates or for the direct use.

4.1 „Three_Step_Sim“

The exemplary project “Three_Step_Sim.mwp” is only used as a simulation (including the controlled system) of a three-step control with the ready-made blocks.

Figure 4-1

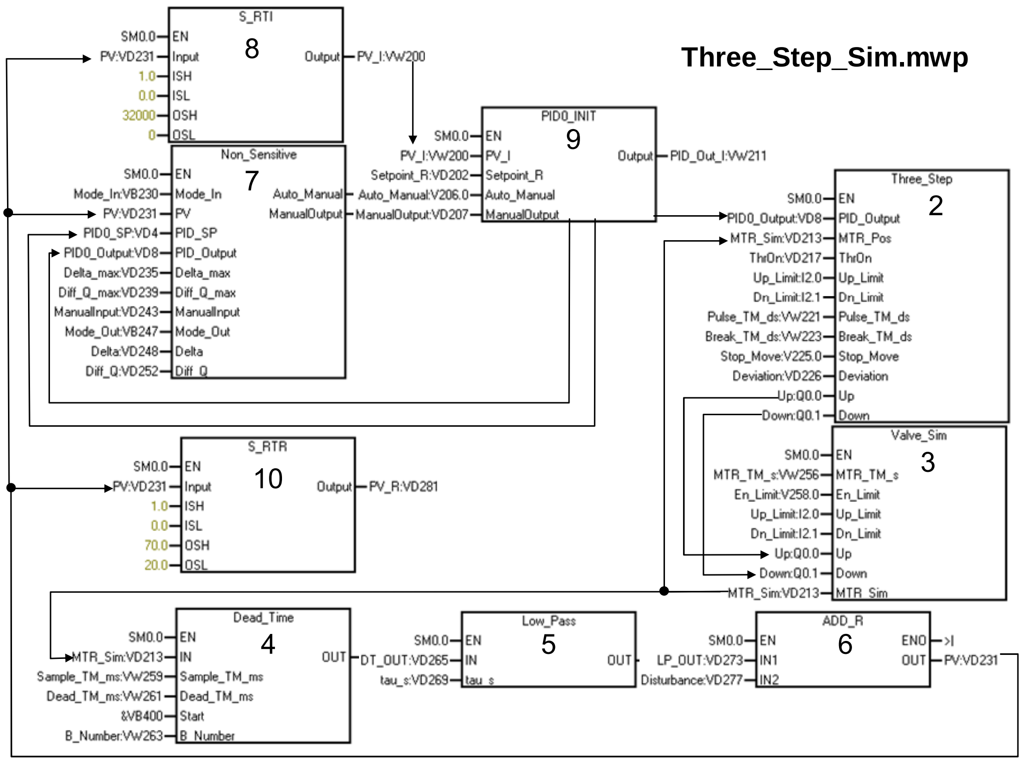


Figure 4-1 shows the block structure of the project “Three_Step_Sim”.

The numbers in the blocks represent the network number. In network 1, which is not depicted here, the parameter initialization is made.

Figure 4-1 is equivalent to the block diagram in figure 2-11. Additionally the scaling block “S_RTI” (network 8) is required. It converts the simulated standardized temperature “PV” into the analog signal of the virtual temperature sensor for the PID-controller “PID0_INIT” (network 9). Furthermore the block “S_RTR” (network 10) is required in order to check the temperature value “PV_R”.

“Disturbance” (network 6) is in this case used to check the disturbance correction.

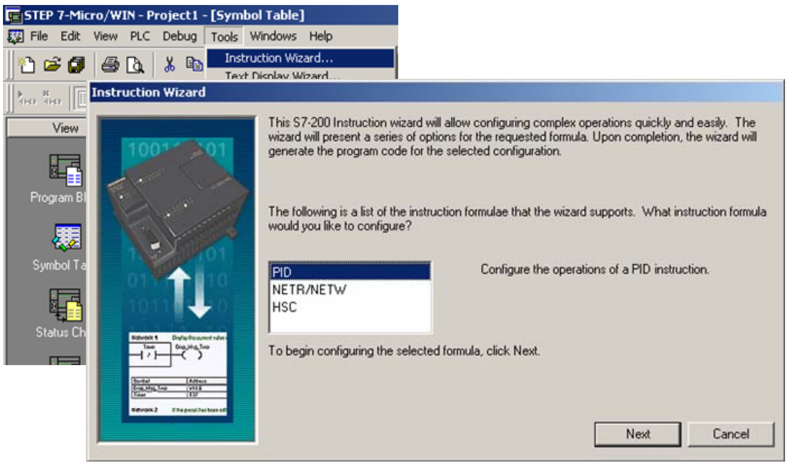
4.1.1 Configuration of the operating mode of the PID-controller

The three-step control is based on the conversion of the control variable calculated by the PID-control loop integrated in Micro/WIN. This PID-controller has to be configured in advance.

The following describes the configuration of the PID-controller in Micro/WIN (as of version 4.0) using the instruction wizard.

1. Open the instruction wizard for the PID-controller

Figure 4-2



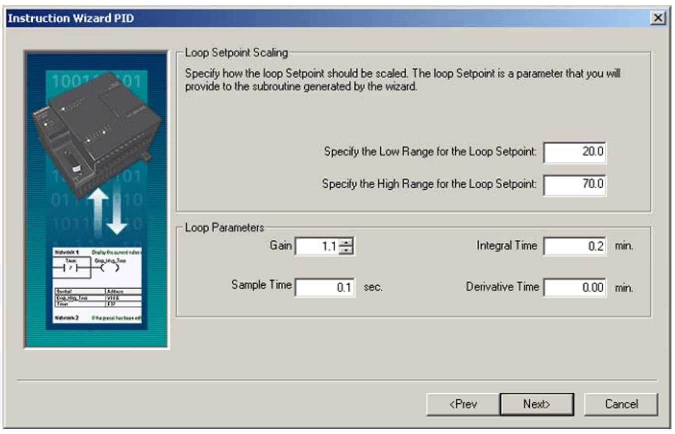
STEP 7-Micro/WIN -> Tools -> Instruction Wizard… -> PID -> Next

2. Select the PID-control loop to be configured (0 - 7)

Next >

3. Loop Setpoint Scaling and Loop Parameters

Figure 4-3

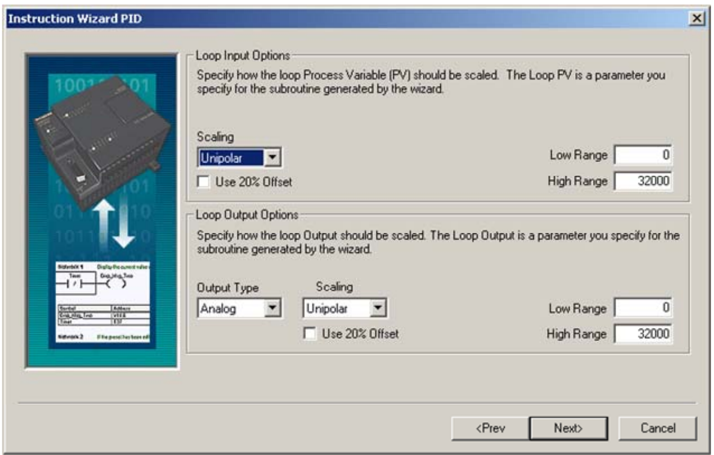


Specify the low range temperature and the high range temperature and select the loop parameters. A PI-control behaviour has to be selected here. Specify the gain, the integral time and the sample time. Later you can adjust the loop parameters to the derivative time.

Next >

4. Process Variable Scaling

Figure 4-4



Set the scaling according to the signal of your temperature sensor. The scaling of the control variable can be adopted unaltered as the standardized control variable signal from the global library memory of the PID-controller is converted into a three-step signal by the block “Three_Step”.

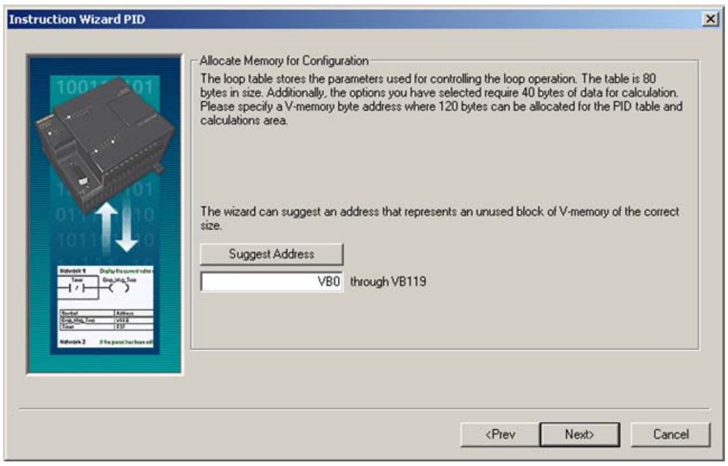
Next >

5. If necessary specify peril points (not required)

Next >

6. Allocation of Memory

Figure 4-5

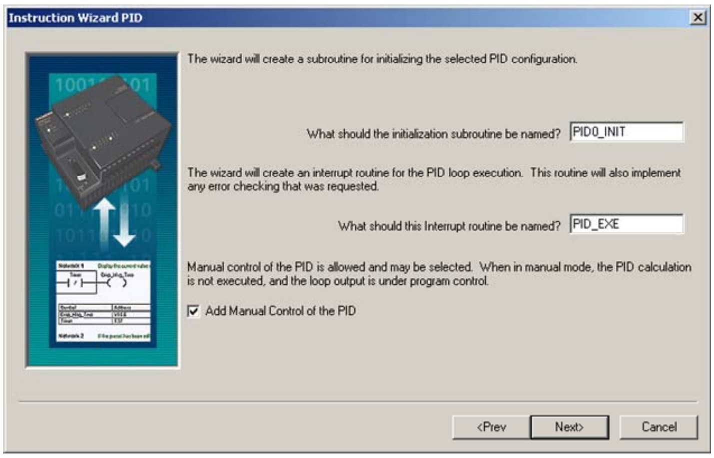


Specify the variable memory for the PID-controller.

Next >

7. Allow manual control of the PID-controller

Figure 4-6



Activate the checkbox for the manual mode.

Next >

8. Finish the Instruction Wizard PID

Finish

9. Do you want to stop the configuration in the wizard?

Yes

4.1.2 Variable table

The handling and monitoring of the three-step control is explained using the variable table of the project "Three_Step_Sim".

For the projects “Three_Step” and “Three_Step_NFB” equivalent variable tables do exist. These are equally designed.

Figure 4-7

	Address	Format	Current Value	New Value
1	Setpoint_R:VD202	Gleitpunkt	0.0	45.0
2	PV_R:VD281	Gleitpunkt	20.0	
3		Mit Vorzeichen		
4	PID0_SP:VD4	Gleitpunkt	0.0	
5	PV:VD231	Gleitpunkt	0.0	
6	PID0_Output:VD8	Gleitpunkt	0.0	
7	MTR_Sim:VD213	Gleitpunkt	0.0	
8	DT_OUT:VD265	Gleitpunkt	0.0	
9	LP_OUT:VD273	Gleitpunkt	0.0	
10	Disturbance:VD277	Gleitpunkt	0.0	
11		Mit Vorzeichen		
12	ThrOn:VD217	Gleitpunkt	0.005	
13	Deviation:VD226	Gleitpunkt	0.0	
14		Mit Vorzeichen		
15	Delta_max:VD235	Gleitpunkt	0.01	
16	Delta:VD248	Gleitpunkt	0.0	
17		Mit Vorzeichen		
18	Diff_Q_max:VD239	Gleitpunkt	0.001	
19	Diff_Q:VD252	Gleitpunkt	0.0	
20		Mit Vorzeichen		
21	Mode_In:VB230	Vorzeichenlos	2	
22	Mode_Out:VB247	Vorzeichenlos	2	
23	Auto_Manual:V206.0	Bit	2#0	
24		Mit Vorzeichen		
25	ManualInput:VD243	Gleitpunkt	0.0	
26	ManualOutput:VD207	Gleitpunkt	0.0	
27		Mit Vorzeichen		
28	PID0_SampleTime:VD16	Gleitpunkt	0.1	
29	PID0_Gain:VD12	Gleitpunkt	1.1	
30	PID0_I_Time:VD20	Gleitpunkt	0.2	
31	PID0_D_Time:VD24	Gleitpunkt	0.0	
32		Mit Vorzeichen		
33	Pulse_TM_ds:VW221	Vorzeichenlos	5	
34	Break_TM_ds:VW223	Vorzeichenlos	5	
35	HYST_Off:V225.0	Bit	2#0	
36		Mit Vorzeichen		
37	MTR_TM_s:VW256	Vorzeichenlos	73	
38	En_Limit:V258.0	Bit	2#0	
39		Mit Vorzeichen		
40	Sample_TM_ms:VW259	Vorzeichenlos	50	
41	VW291	Vorzeichenlos	50	
42	Dead_TM_ms:VW261	Vorzeichenlos	2000	
43	B_Number:VW263	Vorzeichenlos	160	
44		Mit Vorzeichen		
45	tau_s:VD269	Gleitpunkt	5.0	
46	VD311	Gleitpunkt	5.0	

All parameters with a blue rectangle in the “New Value” column can be modified. All the other values display elements for checking purposes. The parameters are divided into blocks and separated by a blank line:

Table 4-2

Line	Block	Function
1 - 2	Temperature	Specified setpoint / process variable checking
4 - 10	Standardized intermediate parameters	Intermediate parameter output / disturbance parameter
12 - 13	Control element deviation	Max. deviation parameter / deviation checking
15 - 16	Control deviation	Max. deviation parameter / deviation checking
18 - 19	Process variable increase	Max. deviation parameter / deviation checking
21 - 23	Operating mode	Parameter / adoption / PID implementation
25 - 26	Manual IN/OUT	Parameter / PID-input
28 - 31	Control parameters	Δt / P / I / D
33 - 35	Three_Step-Parameter	Pulse parameter / break time parameter / hysteresis
37 - 38	Valve_Sim-Parameter	Motor running time / correction option
40 - 43	Dead_Time-Parameter	Δt-parameter/ -checking / t_{dead} / memory size
45 - 46	Low_Pass-Parameter	Tau-parameter / checking

4.1.3 Graphical check

For S7-200 CPUs as of release-version 2.0 STEP 7-Micro/WIN (as of version 4.0) offers the function “PID Tune Control Panel…” (menu item: "Tools").

Figure 4-8

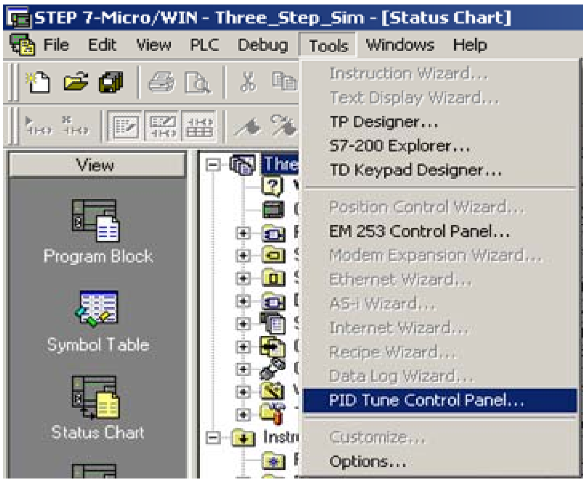
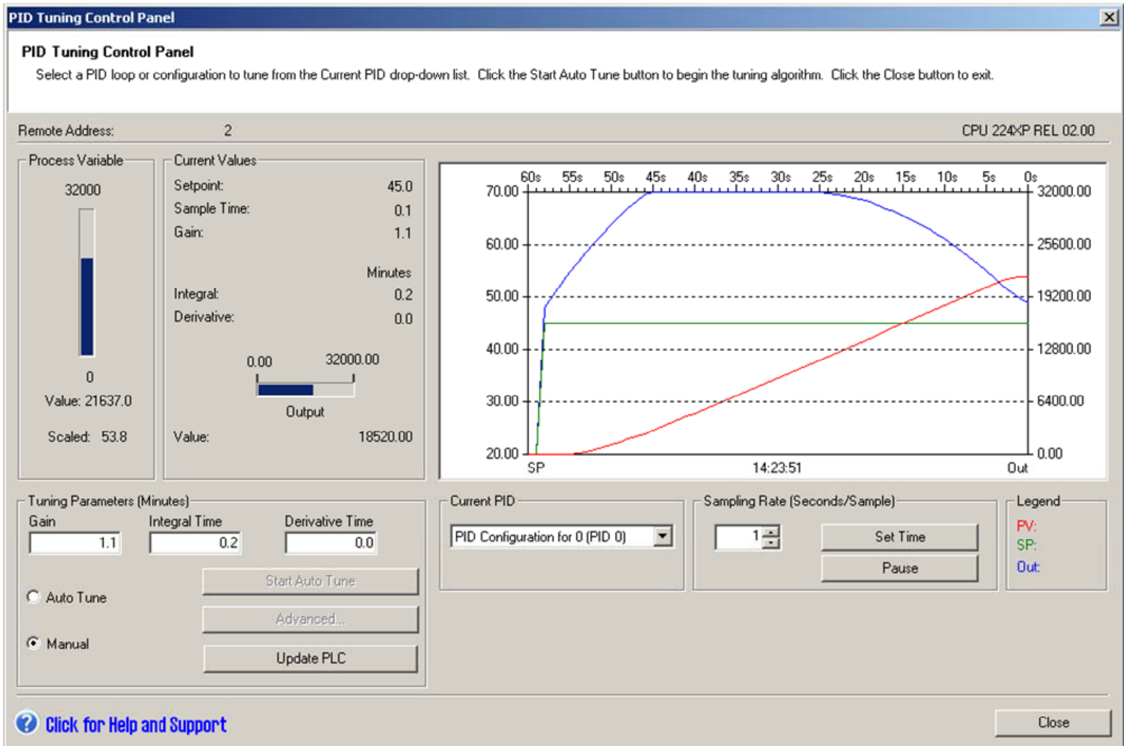


Figure 4-9



Using this control panel, the changes in terms of time of the regulator variables can be monitored: Process variable (red), setpoint (green) and control variable (blue).

Additionally the parameters gain, integral time and derivative time (not required for this control) can be tuned, either manually or automatically.

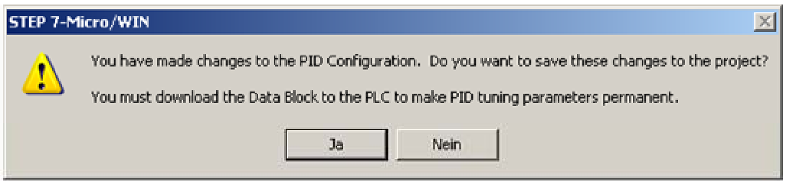
In order to tune the parameters automatically you have to switch the “Non_Sensitive” block to automatic mode (“Mode_In” = “1”).

In order to influence the automatic parameter tuning in terms of disabling single parts:

- Set gain to “0.0” (disable P-part)
- Set integral time to “+INF” (disable I-part)
- Set derivative time to “0.0” (disable D-part)

The adaption (“Update PLC”) and the confirmation (see figure 4-11) when the control panel is closed saves the parameters permanently. Changes are not saved permanently when carried out in the variable table.

Figure 4-10



Further information on the Micro/WIN – PID-controller and on the control panel can be found using the Micro/WIN-F1-Help and in the S7-200 System Manual.

4.2 "Three_Step"

The exemplary project "Three_Step.mwp" is used as a model for three-way mixing valves with valve position feedback.

Figure 4-11

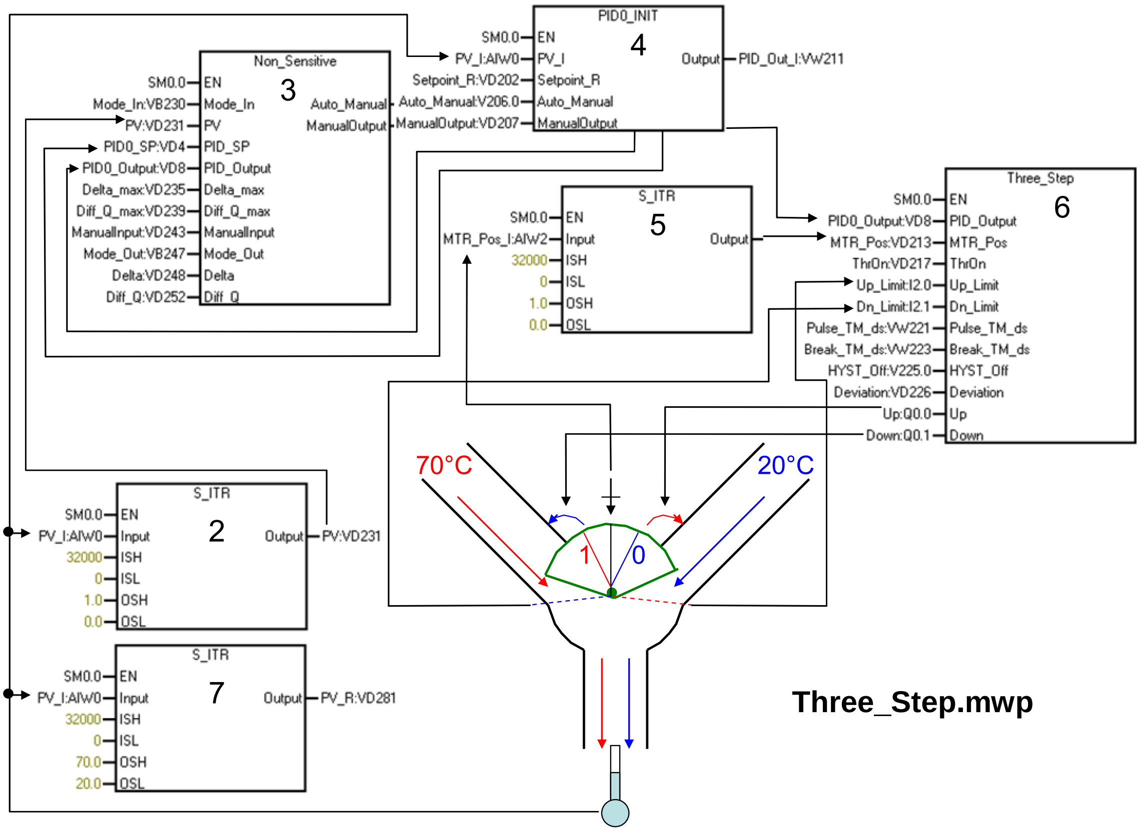


Figure 4-11 shows the block structure of the project "Three_Step". The numbers in the blocks represent the network number. In network 1, which is not depicted here, the parameter initialization is made.

Note The initialization is used as a default. The parameters can be changed to the runtime via the variable table. Nevertheless you must consider that, on the event of a voltage recovery after a grid failure, the initialization is recalled and modifications of parameters will be reset.

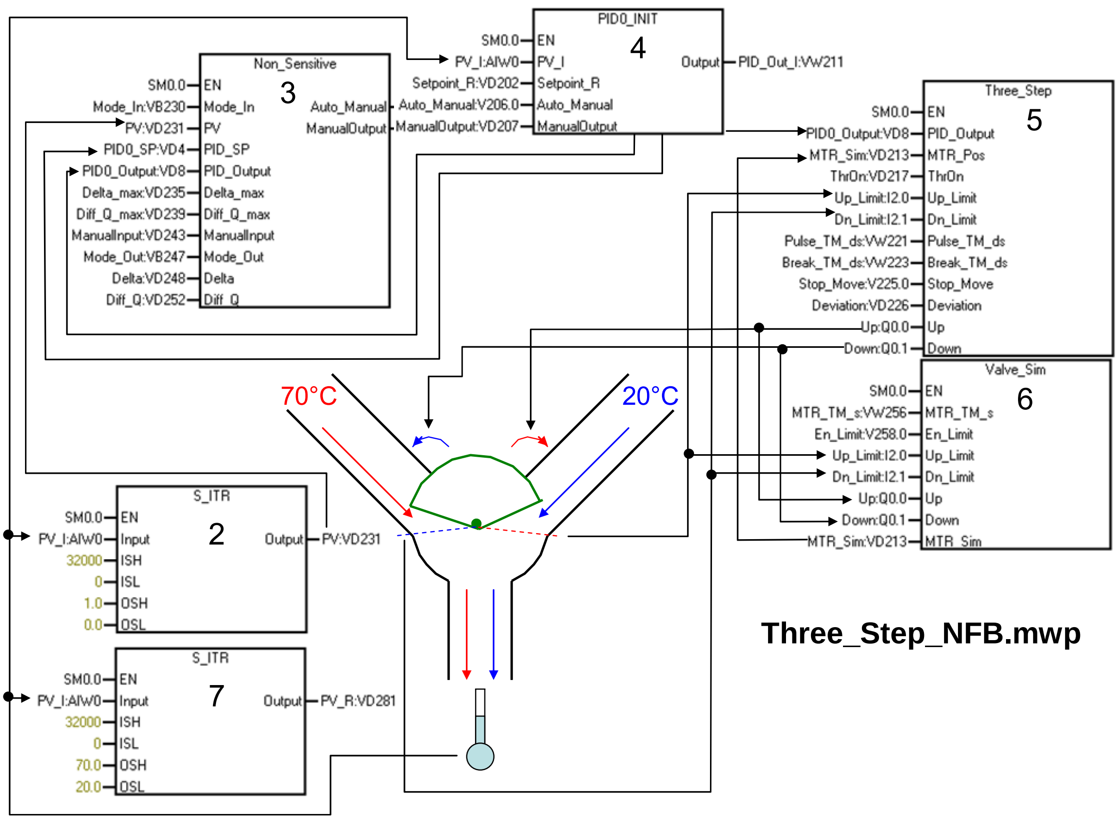
Figure 4-11 is equal to the block diagram in figure 2-4. Additionally the scaling block “S_ITR” is required. It converts the analog input signal of the temperature sensor into the standardized parameter “PV” (network 2) and into the temperature value for checking purposes (network 7). Furthermore it converts the analog input signal of the valve position into the standardized parameter “MTR_Pos” (network 5).

In contrast to figure 4-1 the path simulation (network 4 – 6) is replaced by the three-way mixing valve and by the temperature sensor. The block “Valve_Sim” was eliminated as the valve possesses a position feedback.

4.3 "Three_Step_NFB"

The exemplary project “Three_Step_NFB.mwp” is used as a model for three-way mixing valves without valve position feedback (non feedback).

Figure 4-12



Three_Step_NFB.mwp

Figure 4-12 shows the block structure of the project “Three_Step_NFB”. The numbers in the blocks represent the network number. In network 1, which is not depicted here, the parameter initialization is made.

Note The initialization is used as a default. The parameters can be changed to the runtime via the variable table. Nevertheless you must consider that, on the event of a voltage recovery after a grid failure, the initialization is recalled and modifications of parameters will be reset.

In contrast to figure 4-11 the scaling block “S_ITR” used for the conversion of the analog input signal of the valve position “MTR_Pos” is not required in figure 4-12, as the valve position is calculated via the block “Valve_Sim” (network 6).

4.4 Three-step control quality

The quality of the three-step control depends on the following parameters:

- Behaviour of the controlled system (can hardly be influenced)
- Control parameters (P downsized / enlarge I for a slower control behaviour)
- Tolerance range parameter (can empirically be specified by determination of the smallest occurring deviations in automatic mode “Mode_In” = “1”)
- Three-way control element settings („Pulse_TM_ds“ / „Break_TM_ds“ / "HYST_Off")