Siemens
Industry
Online
Support

APPLICATION EXAMPLE

Configuration of Cascade Control

SIMATIC PCS 7 V9.1 SP2

SIEMENS

Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (https://support.industry.siemens.com) shall also apply.

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 constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial security measures that may be implemented, please visit https://www.siemens.com/industrialsecurity.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the 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 https://www.siemens.com/cert.

Table of contents

1. Preface

1.1. Objective of the Application

A cascade control structure can be established if, beside the main process variable, more system variables are measured and fed back as auxiliary process variables, creating subordinate control loops. Thus there is a master (or main) control loop and a slave (or auxiliary) control loop.

Cascade control is one of the most common control system structures in process plants.

The project example shows the application of cascade control to a simulated process with one main and one auxiliary process variable. The objective of this application note is to show the issues in configuration of cascade control loops that are particularly relevant to achieve fast dynamics and correct behaviour in all operating modes of real world applications.

1.2. Main Contents of this Application Note

The following issues are discussed in this application note:

- Interaction between master and slave controller with respect to controller tuning
- Interaction between master and slave controller with respect to operating mode changeovers
- Simulation example

1.3. Components used

This application example has been created with the following hard- and software components:

Component	Number	Article Number	Note
PCS 7 V9.1 SP2	1	6ES7658-...68-....	

Table 1-1

You can purchase these components from the Siemens Industry Mall.

2. Introduction

2.1. Basic Principles of Cascade Control

A cascade control structure involves two or more PID controllers in serial connection. The manipulated variable of the master controller is connected to the external setpoint of the slave controller so that both control loops are nested.

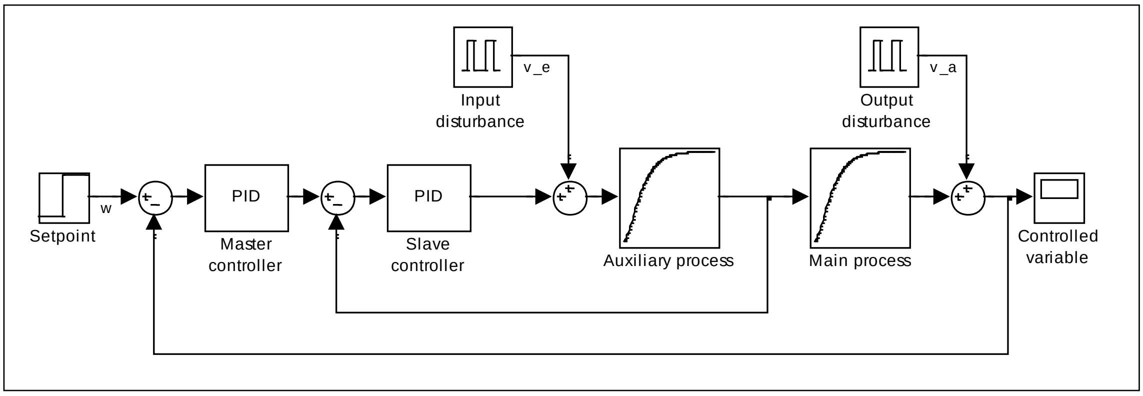


Figure 2-1: Cascade control structure

The advantage of cascade control is that disturbances affecting the inner loop can be compensated much more quickly in the slave loop than in the slower master loop.

In some situations, non-linear effects of the actuator (e.g.: valve) can be compensated in the slave loop so that a linear process response can be achieved for the main control loop.

Cascade control is only feasible when there are further measurable variables in the process in addition to the main control variable and when the inner control loop is significantly faster than the outer loop.

The inner controller will be chosen such that the dynamics of the inner control loop is beneficial, e.g., time constants of the inner loop are compensated. If there is a disturbance affecting the inner loop of the system, this can be controlled better than with single-loop control, where disturbances possibly must pass through the slow time constants of the outer loop.

In Chapter 4 you can see which advantages a cascade control structure has compared to single-loop control of the same system, particularly with respect to disturbance compensation.

2.2. What to Consider for any Cascade Control Loop?

The following notes from \4\, page 7-9 are relevant for all kinds of cascade control.

The manipulated variable range of the master controller must match the setpoint range of the slave controller to ensure proper operation of the anti-windup functions of the master controller. By design of actuators (valves, pumps) and specification of set point-range of slave controller it must be ensured that the slave controller does not work near the edge of its setpoint range at the typical operating point of the master controller, because otherwise the slave controller will have less power of intervention into the process.

If the slave controller is not operated in "cascade" mode (automatic mode with external setpoint) but in any other mode (for example manual or automatic mode with local setpoint) and therefore does not respond to commands of the master controller, the master controller must be switched into "tracking" mode to prevent wind-up of the integral action in the master controller. The manipulated variable of the master controller tracks the process value or setpoint of the slave controller to allow a bumpless return to cascade mode. The difference between tracking the setpoint and tracking the process value becomes apparent when the slave controller is put into manual mode. If the process value is tracked, the response is similar to "Track setpoint to process value in manual mode" of a single loop controller.

If the slave controller reaches a (high/low) manipulated variable limit, the integrator of the master controller must be blocked to prevent it from going on any further in this direction (up / down). The slave controller cannot go any further in this direction anyway. This prevents windup of the master controller when the real actuator has already reached its physical limits while the master controller has not yet reached its manipulated variable limits.

CAUTION

The direction of blocking the integrator of the master controller must be reversed if the slave controller has a negative gain. I.e., if the slave controller reaches a high limit, the integrator of the master controller must be prevented from going further down and vice versa.

2.3. Priorities of Different Operating Modes

The operating mode "manual" of the master controller has higher priority than tracking caused by the slave controller. The operating mode "manual" again has lower priority than force-mode caused by an external logic, e.g. emergency shutdown of the plant (controller input MV_Forced, not limited, activated by MV_ForOn with highest priority). Hence there is an additional tracking input MV_Trk at the PIDConL block of the Advanced Process Library, activated by input MV_TrkOn. This value is limited by the manipulated variable limits and has lower priority than the operating mode "manual".

NOTE

If you work with a PID-controller from the "classic" PCS 7 Library V7.1 or former PCS 7-versions e.g. with the CTRL_PID-Block, the second tracking input is not existing. In these controllers the tracking mode with the input LMN_TRK, activated by LMN_SEL has higher priority than the operating mode "manual". As long as you are running the master controller of a cascade structure in that tracking mode, the operator at the OS has no chance to operate this controller manually.

2.4. Application Examples

- Temperature control of a reactor or tank via flow of a heating medium (or cooling medium):
  $TIC_{reactor}$ → $FIC_{heat\ medium}$ → $Valve_{heat\ medium}$
  See Figure 2-2
- Temperature control of the inner reactor temperature via slave controller for the jacket temperature, sometimes with subordinated controller for heating- and cooling medium flow (multi cascade!):
  TICreactor → TICjacket → FICheat medium → Valve heat medium
  Often in combination with split-range at the slave controller because of heating and cooling.
  See Figure 2-2
- Temperature control of a furnace via slave controller for fuel flow (or combustion gas pressure):
  TIC_{oven} → FIC_{fuel} → $Valve_{fuel}$
- Level control of a tank via slave controller for feed- and/or drain flow:
  LIC_{tank} → FIC_{drain} → $Valve_{drain}$
- Temperature control of a distillation column (master controller) via reflux ratio (slave controller at column head) or vapor flow (slave controller column bottom).
- Position control (drive technology) as a multi cascade structure with slave controllers for revolution speed and torque.

NOTE Generally, slave controllers for flow are applied to avoid that control objectives of the master controller are disturbed by flow fluctuations. Furthermore, typical nonlinearities of the flow actuator (e.g. the valve) are "hidden" in the slave control loop, because the closed slave control loop shows a linear behaviour. This way, nonlinearities of the valve have no effects on the master control loop and its controller design.

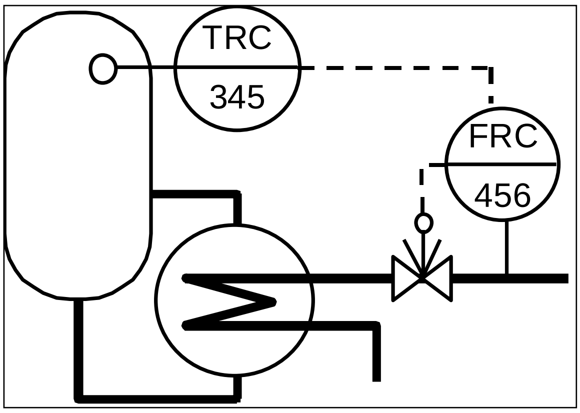


Figure 2-2: Temperature control of a reactor via flow of heating medium by heat exchanger, Source: \5\

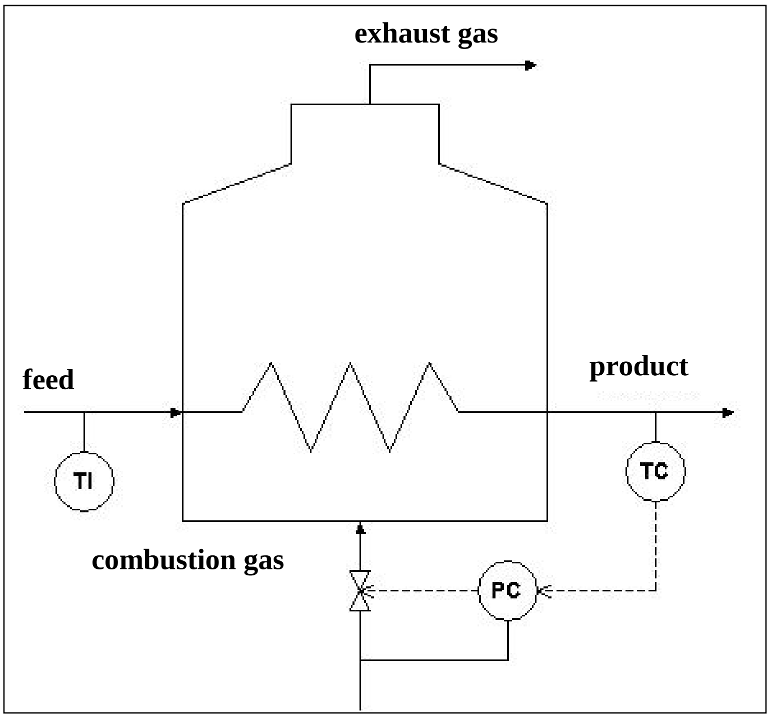


Figure 2-3: Temperature control of a furnace via slave controller for combustion gas pressure, Source: \4\

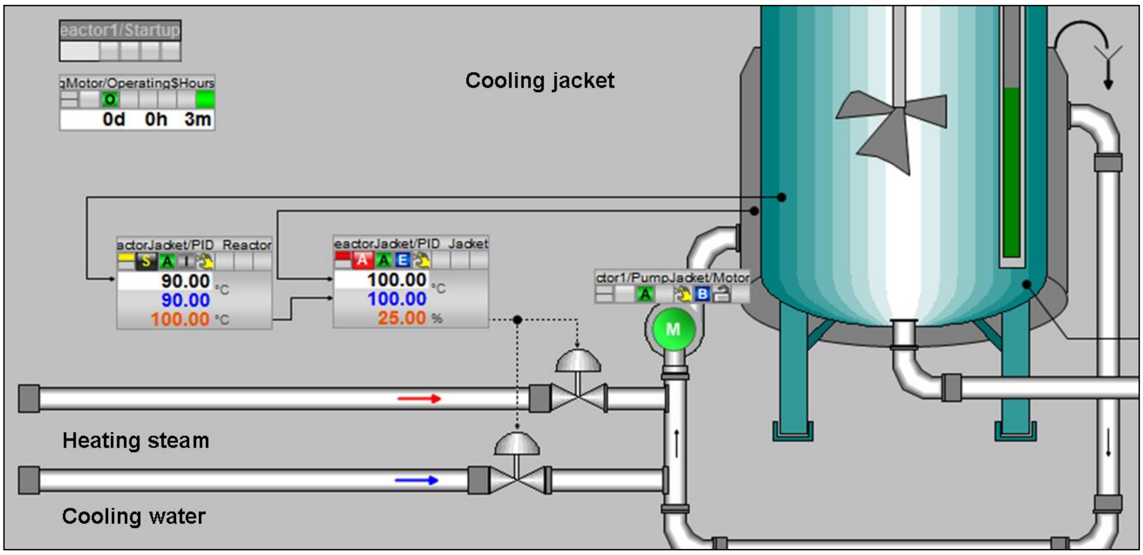


Figure 2-4: Temperature control of the inner reactor temperature via slave controller for the jacket temperature, with split-range-function for heating-and cooling valve, Source: \4\

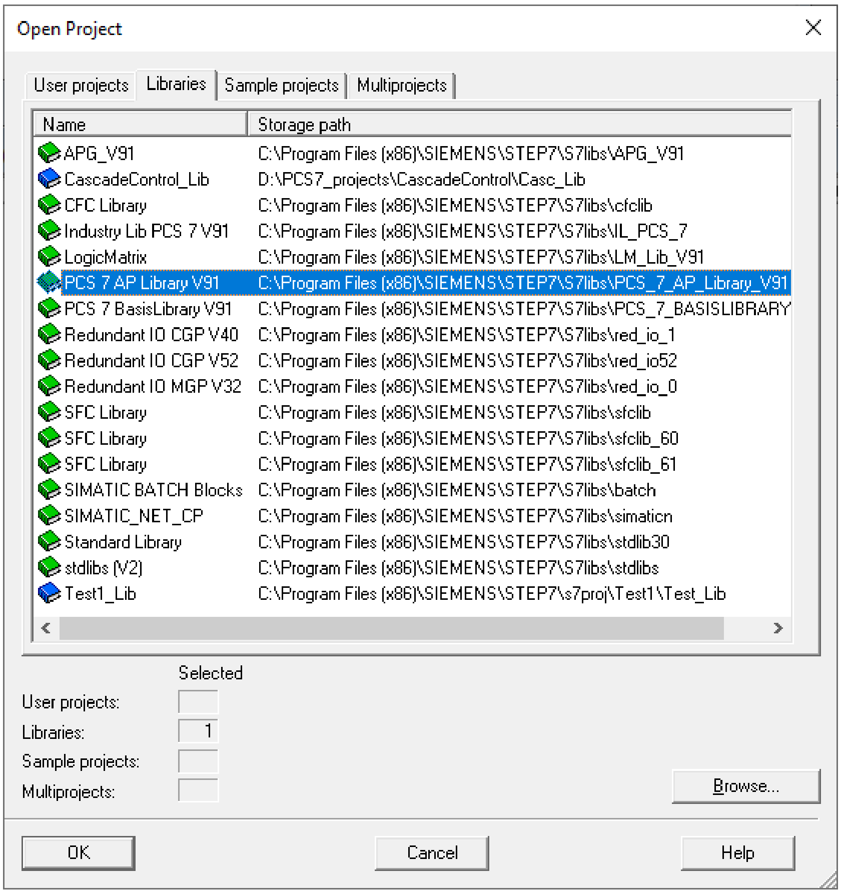
3. Configuration of Cascade Control

If you implement a new cascade control structure in your project, it is recommended to make use of the process tag type "CascadeControl" from the Advanced Process Library.

3.1. Configuration: Creating an Instance of the Process Tag Type

The following steps are carried out for the cascade control in the same way as for any other process tag type.

Open the Advanced Process Library with SIMATIC-Manager: "Open project > Library > PCS 7 AP Library V91".

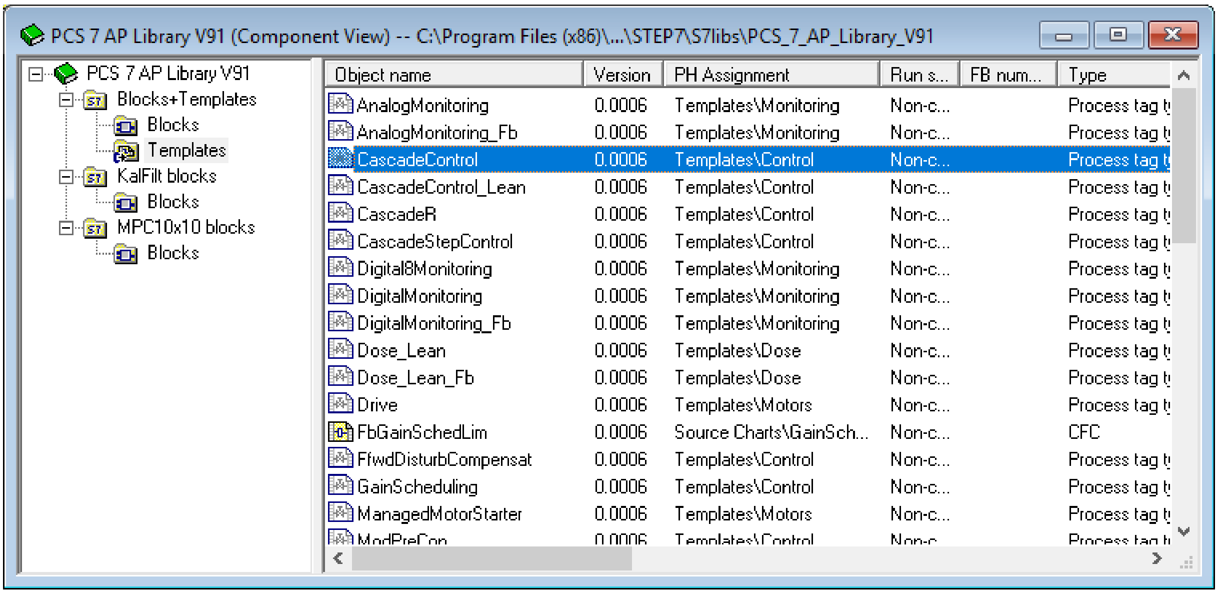


Name	Storage path
APG_V91	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\APG_V91
CascadeControl_Lib	D:\PCS7_projects\CascadeControl\Casc_Lib
CFC Library	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\cfclib
Industry Lib PCS 7 V91	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\IL_PCS_7
LogicMatrix	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\LM_Lib_V91
PCS 7 AP Library V91	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\PCS_7_AP_Library_V91
PCS 7 BasisLibrary V91	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\PCS_7_BASISLIBRARY
Redundant IO CGP V40	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\red_io_1
Redundant IO CGP V52	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\red_io52
Redundant IO MGP V32	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\red_io_0
SFC Library	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\sfclib
SFC Library	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\sfclib_60
SFC Library	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\sfclib_61
SIMATIC BATCH Blocks	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\batch
SIMATIC_NET_CP	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\simaticn
Standard Library	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\stdlib30
stdlibs (V2)	C:\Program Files (x86)\SIEMENS\STEP7\S7libs\stdlibs
Test1_Lib	C:\Program Files (x86)\SIEMENS\STEP7\s7proj\Test1\Test_Lib

	Selected
User projects:	
Libraries:	1
Sample projects:	
Multiprojects:	

Figure 3-1: Open the "PCS 7 AP Library V91"

Copy the process tag type "CascadeControl" from the subfolder "Templates" into the master data library of your PCS 7 multiproject and modify it if necessary, according to your general application requirements.



PCS 7 AP Library V91 (Component View) -- C:\Program Files (x86)\...\STEP7\S7libs\PCS_7_AP_Library_V91

- PCS 7 AP Library V91
  - Blocks+Templates
    - Blocks
    - Templates
  - KalFilt blocks
    - Blocks
  - MPC10x10 blocks
    - Blocks

Object name	Version	PH Assignment	Run s...	FB num...	Type
AnalogMonitoring	0.0006	Templates\Monitoring	Non-c...		Process tag t
AnalogMonitoring_Fb	0.0006	Templates\Monitoring	Non-c...		Process tag t
CascadeControl	0.0006	Templates\Control	Non-c...		Process tag t
CascadeControl_Lean	0.0006	Templates\Control	Non-c...		Process tag t
CascadeR	0.0006	Templates\Control	Non-c...		Process tag t
CascadeStepControl	0.0006	Templates\Control	Non-c...		Process tag t
Digital8Monitoring	0.0006	Templates\Monitoring	Non-c...		Process tag t
DigitalMonitoring	0.0006	Templates\Monitoring	Non-c...		Process tag t
DigitalMonitoring_Fb	0.0006	Templates\Monitoring	Non-c...		Process tag t
Dose_Lean	0.0006	Templates\Dose	Non-c...		Process tag t
Dose_Lean_Fb	0.0006	Templates\Dose	Non-c...		Process tag t
Drive	0.0006	Templates\Motors	Non-c...		Process tag t
FbGainSchedLim	0.0006	Source Charts\GainSch...	Non-c...		CFC
FfwdDisturbCompensat	0.0006	Templates\Control	Non-c...		Process tag t
GainScheduling	0.0006	Templates\Control	Non-c...		Process tag t
ManagedMotorStarter	0.0006	Templates\Motors	Non-c...		Process tag t
ModPreCon	0.0006	Templates\Control	Non-c...		Process tag t

Figure 3-2: Selection of process tag type

Copy the process tag type from the master data library to the application part <project name>_Prj of your multiproject, in the appropriate target folder (Process cell/Unit etc.) in the plant view. You obtain an instance of the process tag type i.e., a CFC chart, which indicates its origin by its symbolic representation.

Rename the new CFC chart and check if the cyclic interrupt OB is correct (in the CFC chart "Edit / Open run sequence").

3.2. Manual Connection of Master and Slave Controller

A manual connection is only required if you want to establish cascade structures in existing control loops in the PLC and therefore you are not prepared to apply the predefined process tag types.

Realize the following connections:

- Connect the output variable MV of the master controller to the input variable SP_Ext of the slave controller.
- Connect the output variable MV of the slave controller to the analog output driver block for the actuator.
- Provide the two controllers with the particular process variables, ranges of values and units from the analog input drivers

The manipulated variable range of the master controller must match the setpoint range (SP_ExtHiLim...SP_ExtLoLim) of the slave controller so that the anti-windup function of the master controller works properly. Therefore the manual MV limits of the master controller are connected to the setpoint limits of slave controller (green mark [1]). In many cases the "automatic limits" of the master controller are also connected to the same values.

If the slave controller reaches a limit of its manipulated variable, the integrator of the master controller is blocked (blue mark [2]).

If the master controller is not in the operating mode “cascade” (automatic mode with external setpoint) but in any other operating mode (e.g.: manual or automatic mode with local setpoint) and therefore not ready for commands by the master controller (shown by PID.CascaCut= true), the master controller must be taken into "tracking mode". An OR-block will be placed in front of the binary input variable PID.MVTrkOn, because the tracking mode has to be activated as well if there is a sensor fault. To assure bumpless transfer back to cascade mode the manipulated variable of the master controller PID.MVTrk must be tracking PV_Slave.PV_Out.

The cycle time of the slave controllers in cascades must be at least as fast as the cycle time of the master controller.

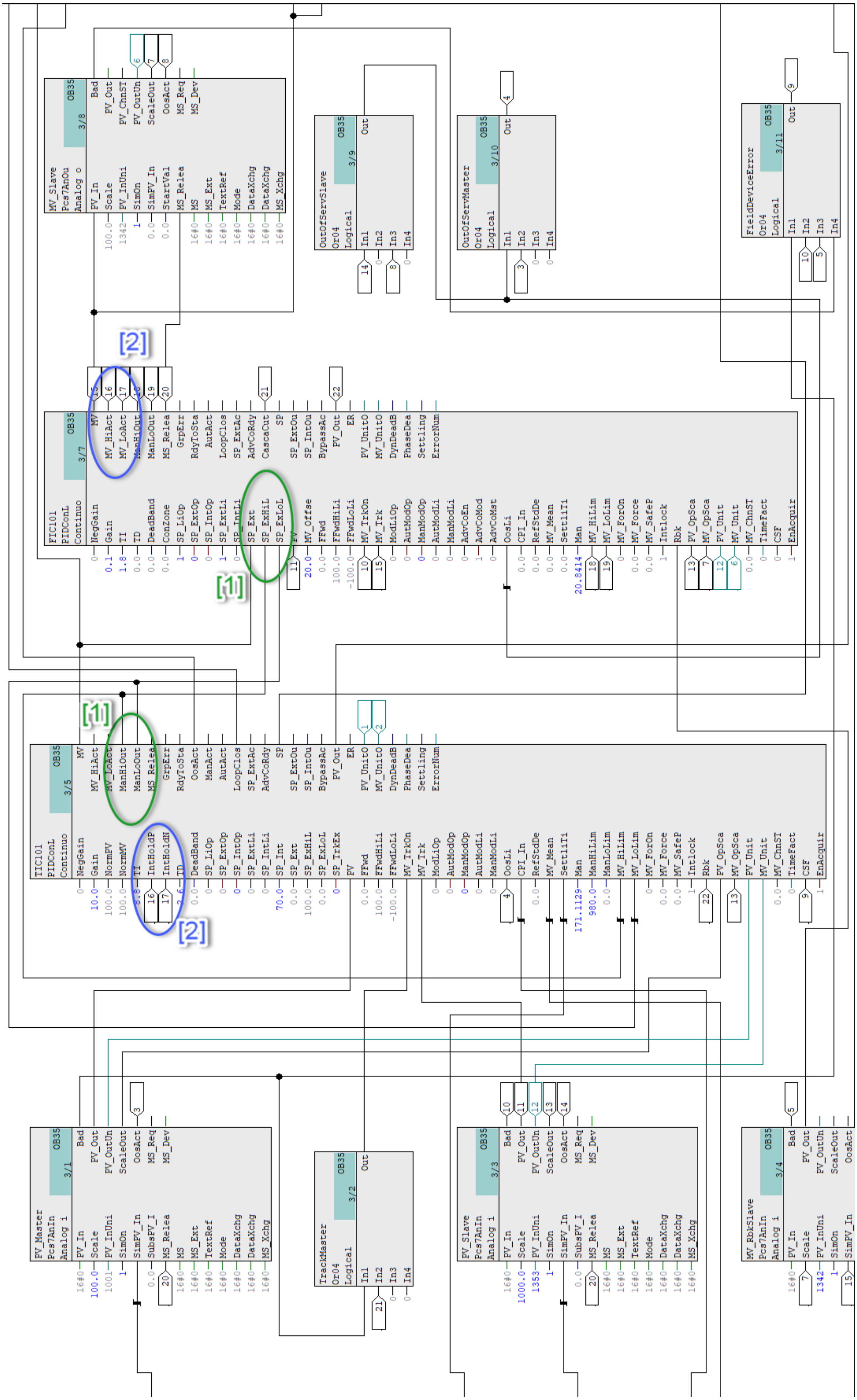


Figure 3-3: Connections between master and slave controller

3.3. Controller Design and Commissioning

The controller design and start of operation are performed "from inside to outside", i.e., first the slave controller is designed and taken in automatic mode. Then the slave controller is switched into cascade mode and the master controller is designed.

Please note that the closed inner control loop appears as the controlled system for the master controller. Therefore, the design of master controller depends on the parameter setting of the slave controller. The bigger the difference in dynamics of auxiliary and main control loop, the less attention is needed with respect to this interaction.

3.3.1. Controller Structures: PID → PI and PI → P Cascade

It is generally recommended that the slave controller of cascade structures is "simpler" than the master controller, i.e., that it has less different dynamic channels (proportional, integral and differential part), because the slave controller then "subordinates" itself to the master controller more easily.

PI → P Cascade

A steady-state deviation of the slave controller is normally not relevant for the application. In contrast the response time of the slave control loop is important because the time constants of the closed slave loop are a part of the controlled system for the master controller. Therefore, the structure PI→P cascade with a proportional-only slave controller is particularly recommended.

If (for these reasons) the slave controller is working without integral action, it is not recommended to specify the setpoint limits of the slave controller exactly to the physical range of the controlled variable of the slave controller, because in this case the operating range of the slave controller is not fully exploited due to the steady-state control error. Instead of that the setpoint limits of the slave controller and the corresponding MV limits of the master controller are adjusted generous.

In this case the anti-windup functions of the master controller are depending on the connections of IntHoldNeg and IntHoldPos. If the slave controller has no integral action, it also has no bumpless manual-automatic switching. Therefore, an MV_Offset should be specified approximating the typical MV value for the operating point of the process.

In \3\, Figure 4-3 and Figure 4-4 it is shown that the dynamic performance of a PI→P cascade is better than PI→PI cascade if the time constants of inner and outer controlled system are in the same order of magnitude.

PID → PI Cascade

The slave controller subordinates itself to the master controller also in a constellation with a PID master and PI slave controller. The slave controller now has an integral action and therefore no more steady-state control error. The master controller achieves a faster compensation of disturbances with the help of his differential action. However, these advantages are bought by higher actuator activity. If a valve is used as an actuator the issues of wear and tear, and the consumption of compressed air (energy) must be considered. The differential action therefore is rarely applied in combination with valves as actuators.

3.3.2. PID-Tuner

The PID-Tuner can be used for cascade control as usual, by calculating the parameters of the slave controller first. Afterwards the slave controller is taken into automatic mode with external setpoint, and the parameters found by the tuner. In the next step you determine the controller parameters of the master controller by application of the PID tuner.

3.4. Combined Structured

Cascade structures are often applied in combination with other control structures. The combination of cascade and split-range control is already mentioned in chapter 2.4. The combination with feedforward disturbance control or gain scheduling is also possible in a straightforward way. Cascade structures with a model based predictive controller as master controller and several PID slave controllers are popular and described in the application note "Multivariable Model Predictive Control – the Distillation Column as an Application Example".

In case of multi cascade structures each slave controller is connected to its directly superior master controller as described in chapter 3.2. Therefore, an operating mode switch will have effects from inner to outer loops: If the innermost controller is taken out of cascade mode, it will inform its master controller which then goes into tracking mode, i.e., the master controller also leaves the cascade mode and informs its master controller and so on.

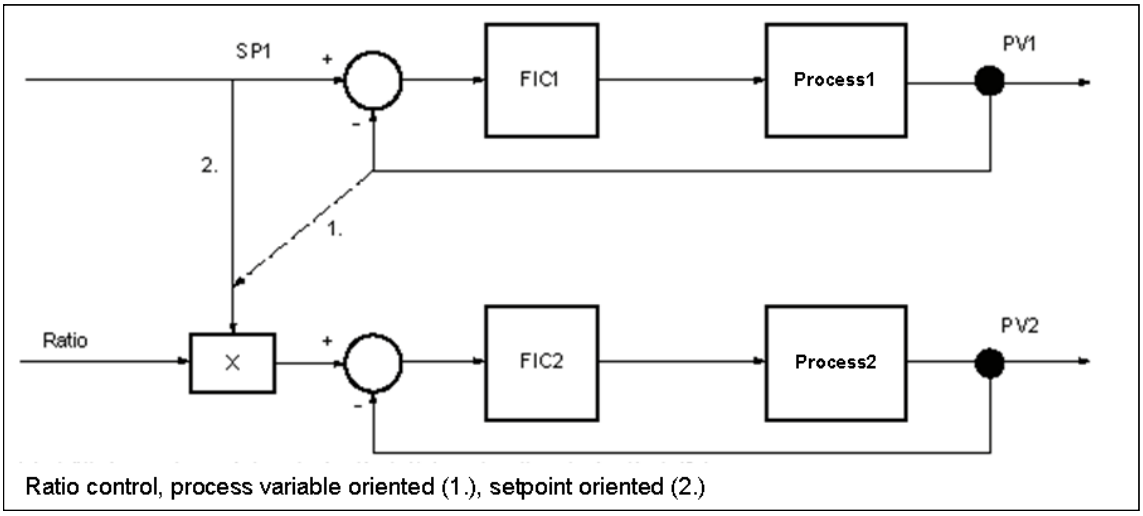

SP1
+
-
FIC1
Process1
PV1
2.
1.
Ratio
X
+
-
FIC2
Process2
PV2
Ratio control, process variable oriented (1.), setpoint oriented (2.)

Figure 3-4: Ratio control, FIC2.SP= FIC1.SP * Ratio, source: \7\

If several fluids or gasses are to be mixed in a specified ratio, this can be realized with a ratio control structure (Figure 3-4) consisting of several flow controllers and a ratio function block. Ratio control shows structural similarities to cascade control. The (external) set point of the flow controller FIC2 of the mixed component is derived from the set point or process variable of the master controller FIC1, in a defined ratio. If the slave controller leaves the cascade mode, corresponding tracking functions must be activated, in this case at the ratio block.

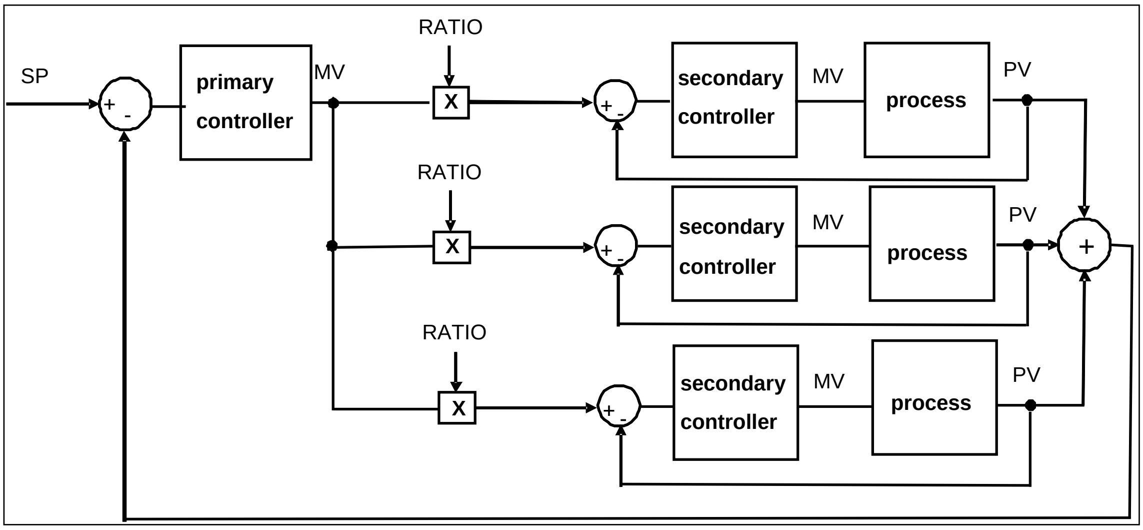


Figure 3-5: Mixture control, source: \8\, example 6

Mixture control (Figure 3-5) is a combination of cascade and ratio control. The master controller provides the total flow of the mixture. The setpoints of all slave controllers are calculated using scaling factors according to ratio control. Each slave controller is responsible for flow control of one component of the mixture. For such a structure it must be discussed in the context of the specific application what to do with the master controller if one of his slave controllers leaves the cascade mode.

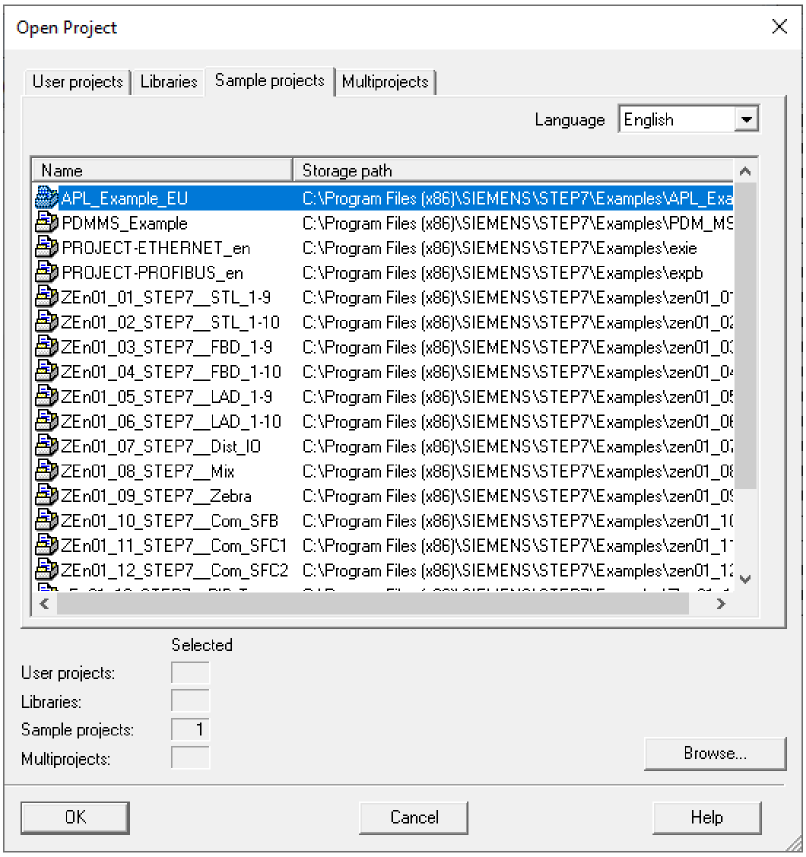
- Should the master controller try to keep the demanded overall flow of the mixture, because the manual intervention at a slave controller is a conscious, temporary manipulation of the mixture ratio?
- Should it stop production completely, because the specified mixture ratio can not be reached any more?

4. Simulation Example

The following example shows a simulation of a Temperature cascade via Heating Medium Flow.

Cascade temperature control via slave controller for heating and/or cooling medium flow is often used for tanks, reactors, or columns, sometimes in combination with heat exchangers or cooling jackets, see e.g., Figure 2-2.

The plant unit "Cascade" in the example project PL_Example_EU of PCS 7 Advanced Process Library contains simulation models for flow and temperature control systems with the parameters of the noise generator blocks. You can open the Sample Project via the task "Sample Projects" in the "Open Project"-dialogue, see Figure 4-1.



Language: English

Name	Storage path
APL_Example_EU	C:\Program Files (x86)\SIEMENS\STEP7\Examples\APL_Exa
PDMMS_Example	C:\Program Files (x86)\SIEMENS\STEP7\Examples\PDM_MS
PROJECT-ETHERNET_en	C:\Program Files (x86)\SIEMENS\STEP7\Examples\exie
PROJECT-PROFIBUS_en	C:\Program Files (x86)\SIEMENS\STEP7\Examples\expb
ZEn01_01_STEP7__STL_1-9	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_02_STEP7__STL_1-10	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_03_STEP7__FBD_1-9	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_04_STEP7__FBD_1-10	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_05_STEP7__LAD_1-9	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_06_STEP7__LAD_1-10	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_07_STEP7__Dist_IO	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_08_STEP7__Mix	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_09_STEP7__Zebra	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_0
ZEn01_10_STEP7__Com_SFB	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_1
ZEn01_11_STEP7__Com_SFC1	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_1
ZEn01_12_STEP7__Com_SFC2	C:\Program Files (x86)\SIEMENS\STEP7\Examples\zen01_1

	Selected
User projects:	
Libraries:	
Sample projects:	1
Multiprojects:	

Figure 4-1: Open "APL_Example_EU" sample project

Typical for this kind of applications is: The temperature control process is slower than the flow control process. The two time constants of the temperature control process are a far away from each other. There is an offset according to the environment temperature. The temperature control process shows less noise than the flow control process. Real world temperature control processes are typically much slower than this simulation, whereas the time constants of the flow control process are realistic.

In order to show how significant, the improvement by cascade control is compared to single loop control, the time lags of the temperature simulation model are multiplied by factor 10 compared to APL_Example_EU. The controller parameters of the master and slave controller are calculated by PID-tuner (see Table 4-1). Using these new controller parameters, a negative disturbance step change of 50 l/h was recorded.

ProcSimC	Flow control process	Temperature control process
Gain	8,0	0,3
TimeLag1	1,0 s	80,0 s
TimeLag2	1,0 s	10,0 s
PV0	0,0	20,0
Noise Variance	0,22	0,1

Table 4-1: Process parameters of the modified cascade control example project

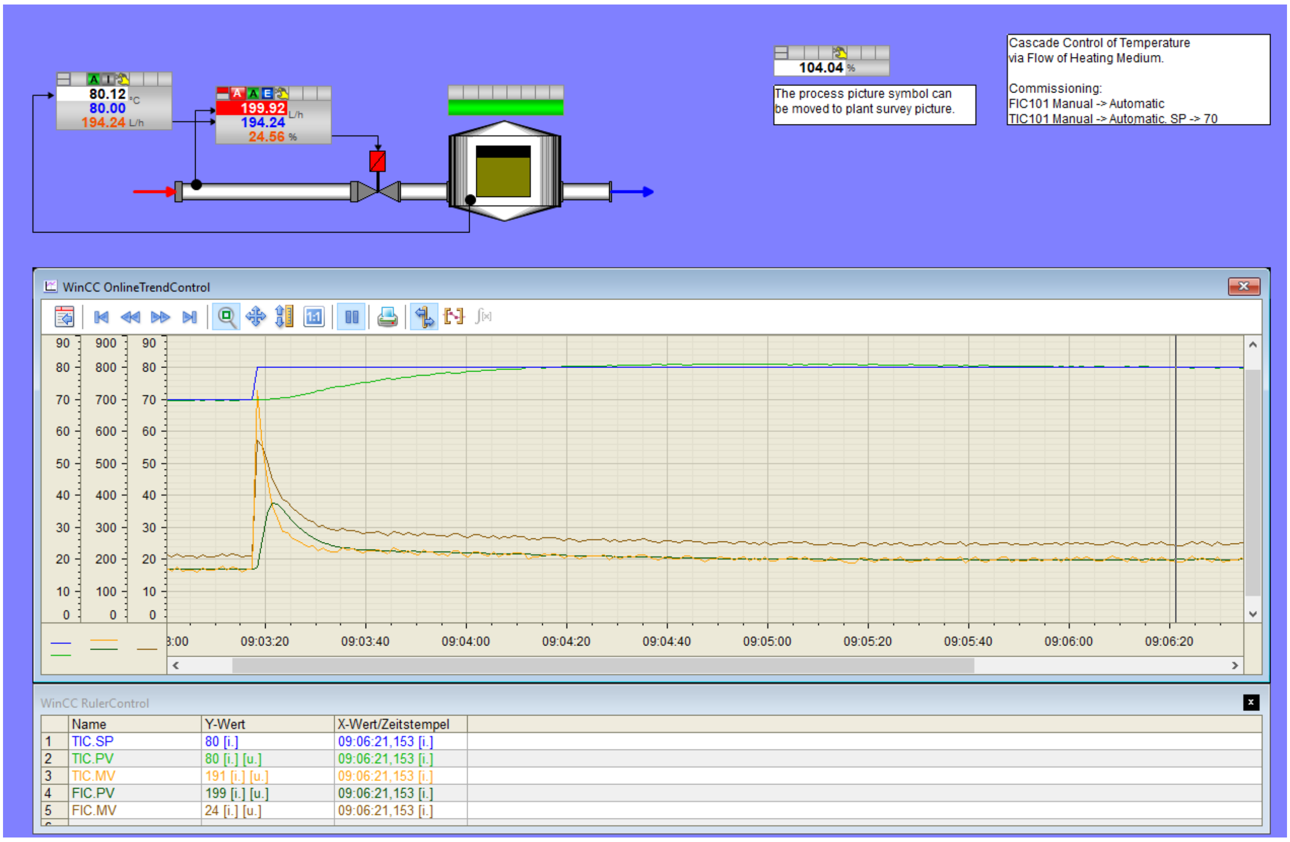
Transitions between different operation modes can be tested using the simulation model. In addition, the performance of different parameter sets for master and slave controller can be tested.

The parameters in Table 4-2 below result in fast control performance with small control errors but strong control action. The PID master controller, designed by the PID-tuner for "fast temperature control systems", had the Gain = 104. This control design is intended for fast temperature control systems with binary actuators, e.g. electrical heating with semiconductor relays and pulse width modulation.

Such an aggressive design however results in very fast fluctuations of the manipulated variable of the master controller i.e., fast fluctuations of the setpoint of the slave controller which cannot be followed fast enough. The slave control loop oscillates with a visible phase shift between setpoint and process variable. Therefore, such aggressive controller designs are not suitable for master controllers of cascades. In the simulation example the gain was reduced to 20. In general, this procedure (gain reduction by factor 2 to 5) is recommended if the PID-tuner for cascade master controllers suggests the "Design for fast temperature controllers".

PID	TIC-501	FIC-501
Gain	20,0	0,1
TI	43,3 s	1,8 s
TD	13,4 s	0,0 s

Table 4-2: Controller parameters for PID → PI cascade with fast dynamic performance



The process picture symbol can be moved to plant survey picture.

Cascade Control of Temperature via Flow of Heating Medium.

Commissioning:
FIC101 Manual -> Automatic
TIC101 Manual -> Automatic, SP -> 70

	Name	Y-Wert	X-Wert/Zeitstempel
1	TIC.SP	80 [i.]	09:06:21,153 [i.]
2	TIC.PV	80 [i.] [u.]	09:06:21,153 [i.]
3	TIC.MV	191 [i.] [u.]	09:06:21,153 [i.]
4	FIC.PV	199 [i.] [u.]	09:06:21,153 [i.]
5	FIC.MV	24 [i.] [u.]	09:06:21,153 [i.]

Figure 4-2: Setpoint step with PID→ PI Cascade

The parameters in the following tables have the advantage of smoother handling the actuator (e.g.: valve), because the master controller has no differential action.

PID	TIC-501	FIC-501
Gain	26,0	0,1
TI	45,5 s	0,0 s
TD	0,0 s	0,0 s

Table 4-3: Controller parameters for PI → P cascade with smooth actuator movements

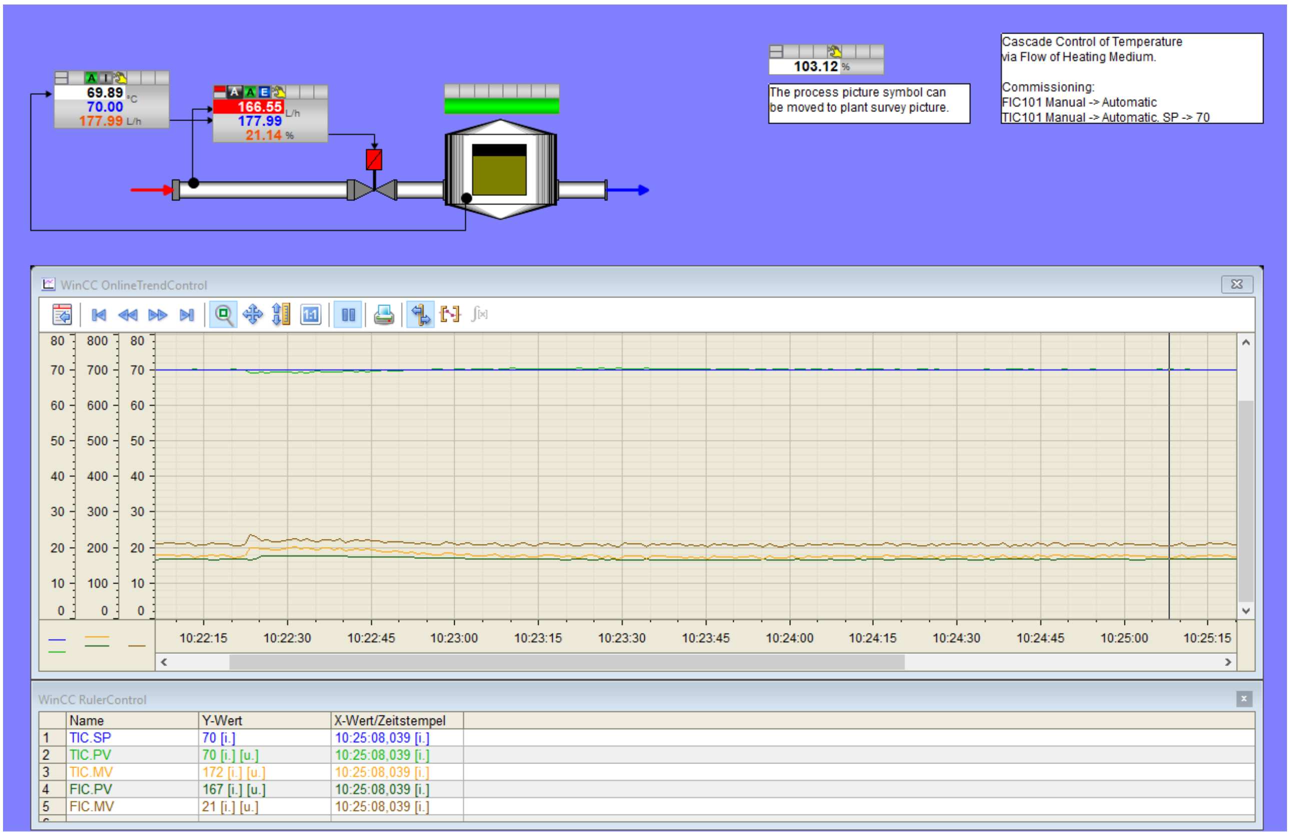
PID	TIC-501	FIC-501
Gain	26,0	0,1
TI	45,5 s	1,8 s
TD	0,0 s	0,0 s

Table 4-4: Controller parameters for PI → PI cascade with smooth actuator movements

In a further step the slave controller was bypassed, resulting in single loop PI control. With the help of the controller parameter were also determined by PID-Tuner (see Table 4-5) and the same disturbance step change was simulated as in the case of cascade control.

PID	TIC-501	FIC-501
Gain	0,82	
TI	56,9 s	-
TD	0,0 s	-

Table 4-5: Controller parameters for single loop PI-control



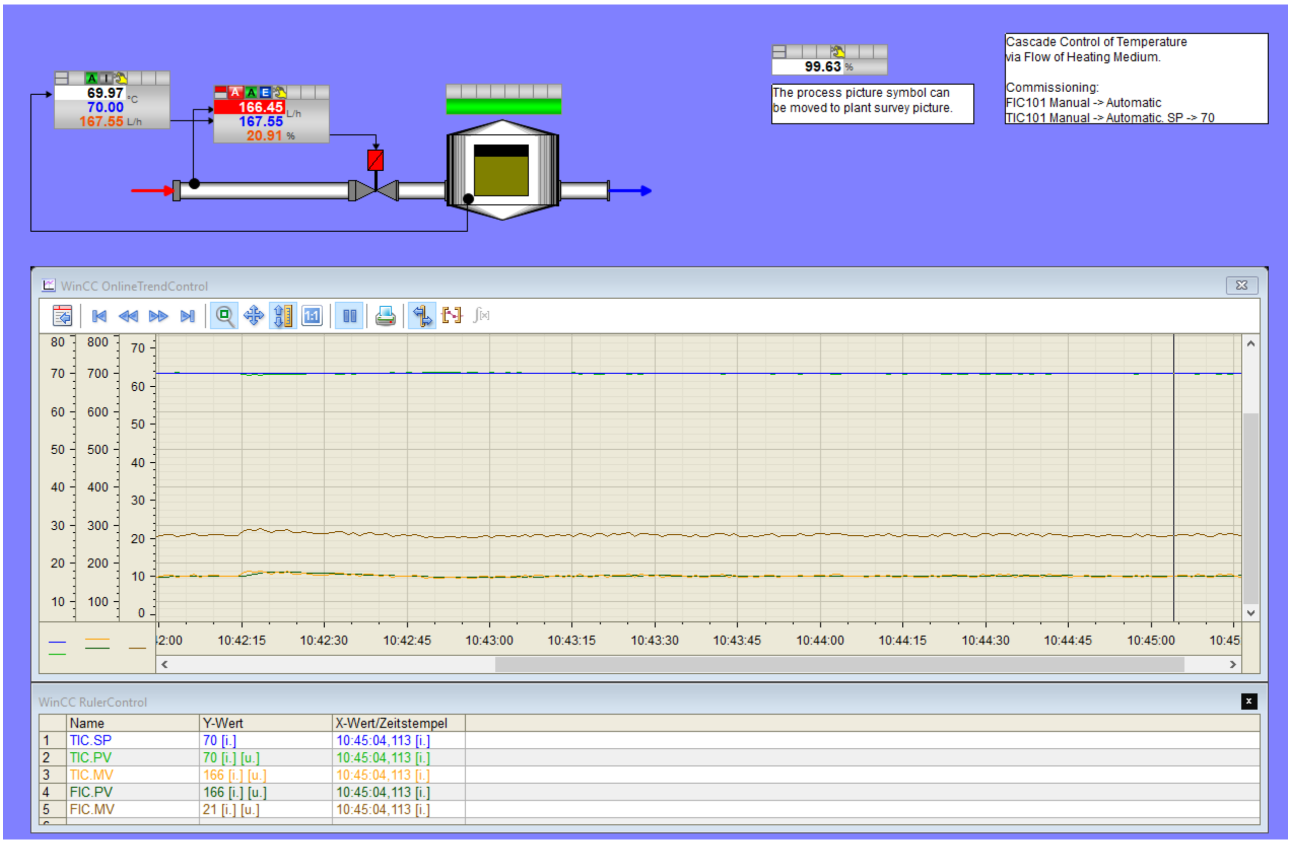
The process picture symbol can be moved to plant survey picture.

Cascade Control of Temperature
via Flow of Heating Medium.

Commissioning:
FIC101 Manual -> Automatic
TIC101 Manual -> Automatic. SP -> 70

	Name	Y-Wert	X-Wert/Zeitstempel
1	TIC.SP	70 [i.]	10:25:08,039 [i.]
2	TIC.PV	70 [i.] [u.]	10:25:08,039 [i.]
3	TIC.MV	172 [i.] [u.]	10:25:08,039 [i.]
4	FIC.PV	167 [i.] [u.]	10:25:08,039 [i.]
5	FIC.MV	21 [i.] [u.]	10:25:08,039 [i.]

Figure 4-3: P I→ P cascade control with flow disturbance step change, shows a steady state control error between TIC.MV=FIC.SP and FIC.PV



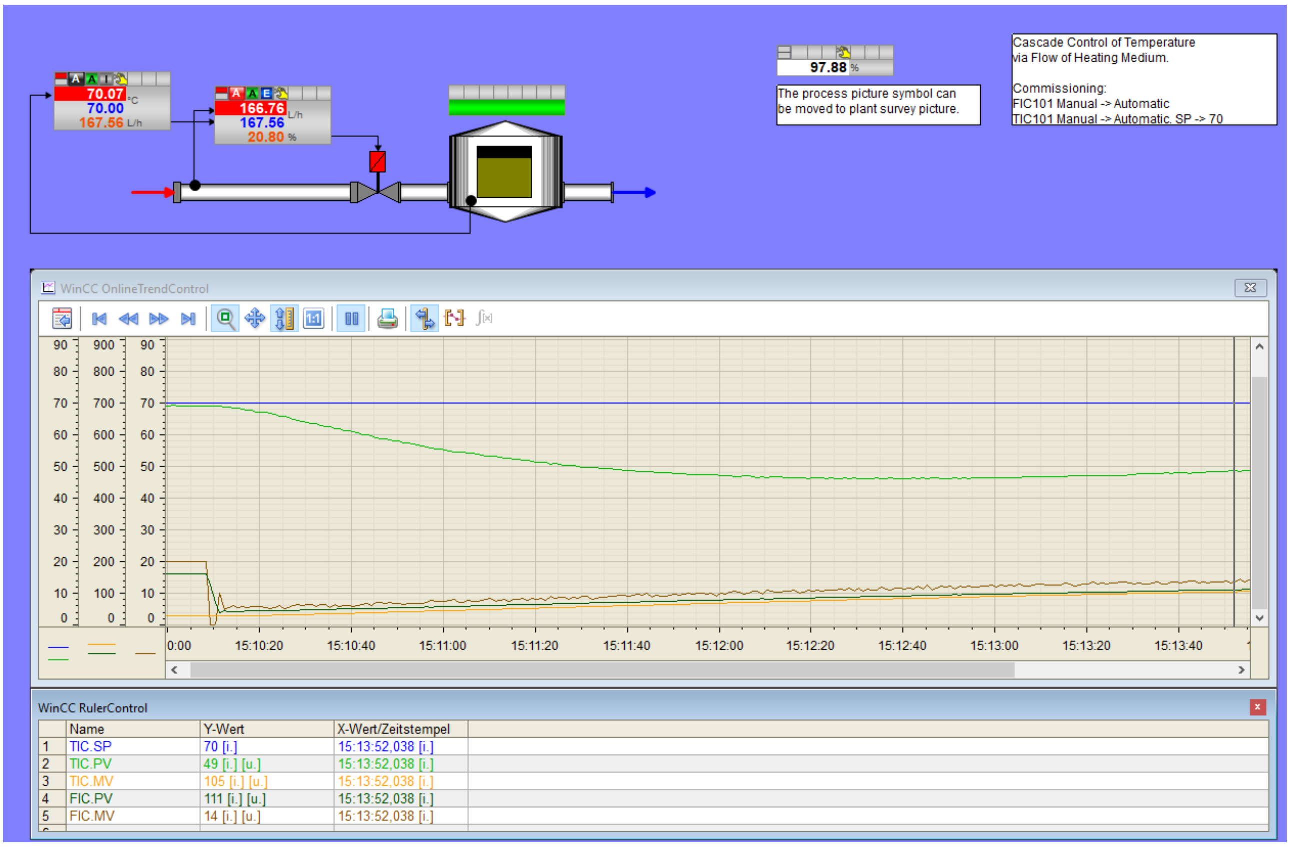
The process picture symbol can be moved to plant survey picture.

Cascade Control of Temperature via Flow of Heating Medium.

Commissioning:
FIC101 Manual -> Automatic
TIC101 Manual -> Automatic, SP -> 70

	Name	Y-Wert	X-Wert/Zeitstempel
1	TIC.SP	70 [i.]	10:45:04,113 [i.]
2	TIC.PV	70 [i.] [u.]	10:45:04,113 [i.]
3	TIC.MV	166 [i.] [u.]	10:45:04,113 [i.]
4	FIC.PV	166 [i.] [u.]	10:45:04,113 [i.]
5	FIC.MV	21 [i.] [u.]	10:45:04,113 [i.]

Figure 4-4: PI → PI cascade control with flow disturbance step change, shows nearly no deviation from setpoint



The process picture symbol can be moved to plant survey picture.

Cascade Control of Temperature via Flow of Heating Medium.

Commissioning:
FIC101 Manual -> Automatic
TIC101 Manual -> Automatic, SP -> 70

	Name	Y-Wert	X-Wert/Zeitstempel
1	TIC.SP	70 [i.]	15:13:52,038 [i.]
2	TIC.PV	49 [i.] [u.]	15:13:52,038 [i.]
3	TIC.MV	105 [i.] [u.]	15:13:52,038 [i.]
4	FIC.PV	111 [i.] [u.]	15:13:52,038 [i.]
5	FIC.MV	14 [i.] [u.]	15:13:52,038 [i.]

Figure 4-5: Single loop controller with flow disturbance step change, shows the biggest deviation from setpoint

PI → PI cascade control provides the best results with respect to compensation of a disturbance variable step change in the slave control loop in this example. The reason is that the time constant of the master control loop is much slower than that of the slave control loop. A disturbance inside the slave control loop is compensated by the integral action of this controller fast and precisely, such that the master control loop is not affected by the consequences of this disturbance. This is the desired effect of cascade control!

In case of the proportional-only slave controller however, the steady-state control error in the slave loop must be compensated by the master controller. Therefore, the effect of the disturbance is a little more visible. Conclusion: a proportional-only slave controller is only recommended if the time constants of master and slave control loop are in the same order of magnitude (see APL_Example_EU and \3\), because in such cases the integral actions of slave and master controller would interact with each other in an uncoordinated way.

5. Conclusion

If further process variables (auxiliary variables) besides the main controlled variable can be measured in a process, the application of cascade control can be advisable, because disturbances in slave control loops are recognized and compensated faster as this would be possible in the slower, main control loop.

6. Appendix

6.1. Service and support

SiePortal

The integrated platform for product selection, purchasing and support - and connection of Industry Mall and Online support. The SiePortal home page replaces the previous home pages of the Industry Mall and the Online Support Portal (SIOS) and combines them.

- Products & Services
  In Products & Services, you can find all our offerings as previously available in Mall Catalog.
- Support
  In Support, you can find all information helpful for resolving technical issues with our products.
- mySieportal
  mySiePortal collects all your personal data and processes, from your account to current orders, service requests and more. You can only see the full range of functions here after you have logged in.

You can access SiePortal via this address: sieportal.siemens.com

Industry Online Support

Industry Online Support is the previous address for information on our products, solutions and services.

Product information, manuals, downloads, FAQs and application examples - all information is available with just a few mouse clicks: support.industry.siemens.com

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts.
Please send queries to Technical Support via Web form: support.industry.siemens.com/cs/my/src

SITRAIN – Digital Industry Academy

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.
For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page: siemens.com/sitrain

Industry Online Support app

You will receive optimum support wherever you are with the "Industry Online Support" app. The app is available for iOS and Android:

Available on the App Store

ANDROID APP ON Google play

6.2. Links and literature

Nr.	Thema
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to this entry page of this application example https://support.industry.siemens.com/cs/ww/de/view/43033319
\3\	Föllinger, O., Regelungstechnik 8, überarbeitete Auflage Hüthig-Verlag, Heidelberg, 1994 ISBN 3-77852336-8, Kap. 7.10.1.
\4\	Dittmar, R., Pfeiffer, B-M., Modellbasierte prädiktive Regelung - Eine Einführung für Ingenieure. Oldenbourg Verlag, München, 2004 ISBN 3-486-27523-2, Kap. 1.2.2.
\5\	Müller, J., Pfeiffer, B-M., Hunger, V., Regeln mit Simatic – Praxisbuch für Regelungen mit Simatic S7 und PCS 7. Zweite, erweiterte Auflage Publicis MCD Verlag, Erlangen, 2002 ISBN 3-89578-195-9
\6\	Pfeiffer, B-M., PCS 7 Solution-Template für Rührkesselreaktoren mit Mantelkühlung I IA&DT ATS 32 Karlsruhe, Feb. 2010.
\7\	Online-Hilfe PCS 7 Advanced Process Library Kapitel: Templates / Messstellentypen.
\8\	Handbuch SIMATIC Modular PID-Control Siemens A&D AS, Nov. 2003.

Table 6-1

6.3. Change documentation

Version	Date	Modification
V1.0	05/2010	First version
V2.0	08/2023	Revision and Update to PCS 7 V9.1 SP2

Table 6-2