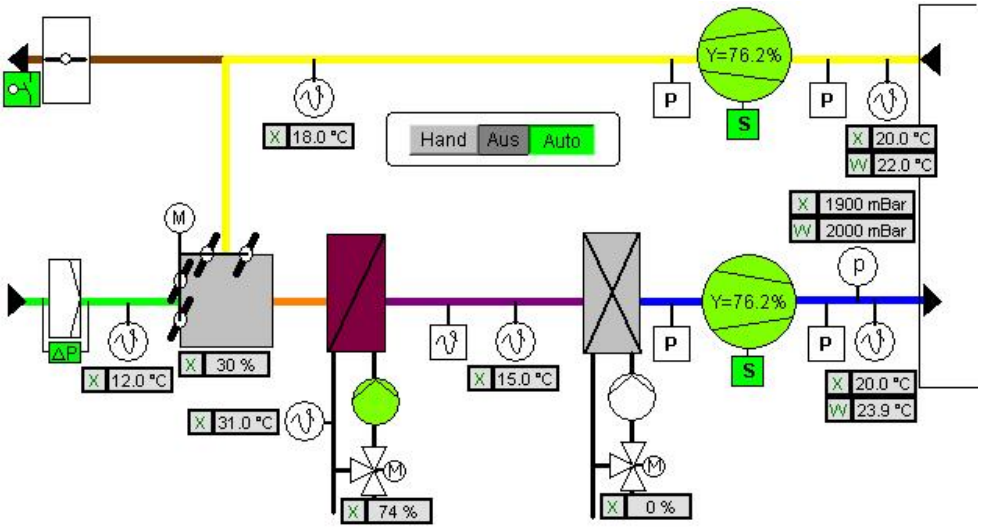
WAGO I/O SYSTEM 750

Libraries for Building Automation



Function Block Descriptions for HVAC Functions

Last update: 27.11.2019

Copyright © 2019 by WAGO Kontakttechnik GmbH & Co. KG
All rights reserved.

WAGO Kontakttechnik GmbH & Co. KG
Hansastraße 27
D-32423 Minden

Phone: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69

E-mail: info@wago.com

Web: http://www.wago.com

Technical Support
Phone: +49 (0) 571/8 87 – 4 45 55
Fax: +49 (0) 571/8 87 – 84 45 55

E-mail: support@wago.com

Every conceivable measure has been taken to ensure the accuracy and completeness of this documentation. However, as errors can never be fully excluded, we always appreciate any information or suggestions for improving the documentation.

We wish to point out that the software and hardware terms, as well as the trademarks of companies used and/or mentioned in the present document, are generally protected by trademark or patent.

WAGO-I/O-PRO Library for Building Automation

Contents

Important Notes 6

Copyright ... 6
Personnel Qualification ... 6
Intended Use ... 6
Scope of Validity ... 7

01 System Monitoring 8

Collective Malfunction (FbCollectiveMalfunction) ... 8
Global Collective Malfunction (FbGlobalMalfunction) ... 11
Start/Stop (FbStartStop) ... 12
Start/Stop Optimization (FbStartStopOptimization) ... 13
Start/Stop Heating Circuit Control (FbStartStopHeatingCircuit) ... 18
Filter Monitoring (FbFilterMonitoring) ... 23
Summer Night Ventilation (FbSummerNightVentilation) ... 24

02 Antifreeze Protection 27

Air-Side Antifreeze Protection (FbAntifreezeAir) ... 27
Water-Side Antifreeze Protection (FbAntifreezeWater) ... 29

03 Damper Control 33

Actuation of 2-Point Dampers (Fb2PointDamper) ... 33
Actuation of Continuous Dampers (FbContinuousDamper) ... 36

04 Fan Control 39

Fan, 1-Level (FbFan_1Level) ... 39
Fan, 2-Level (FbFan_2Level) ... 42
Fan, 3-Level (FbFan_3Level) ... 46
Fan with Frequency Converter (FbFan_FC) ... 47

05 Controllers 51

PID Controller (FbPIDController) ... 51
PID Controller wit Two Sets of Control Parameters (FbPIDController2PIDSets) ... 54
PI Limit Controller (FbLimitController) ... 58
Cascade Controller (FbCascadeController) ... 60
Sequence Controller (FbSequenceController) ... 63
Two-Point Controller for DHW Preparation (FbDHWController) ... 66

06 Sequences 70

Cooling Sequence (FbSequenceCooling) ... 70
Energy Recovery Sequence (FbSequenceEnergyRecovery) ... 72
Mixed Air Sequence (FbSequenceMixedAir) ... 74
Heating Sequence (FbSequenceHeating) ... 77

Dehumidifying Sequence (FbSequenceDehumidifying) 79
Humidifying Sequence (FbSequenceHumidifying)..... 81

07 Heat Exchangers 83

Plate-Type Heat Exchanger (FbPlateHeatExchanger)..... 83
Rotary Heat Exchanger (FbRotaryHeatExchanger)..... 86
Run-Around Coil (FbRunAroundCoil)..... 89

08 Pumps and Valves 92

Pump (FbPump)..... 92
Pump with Frequency Converter (FbPumpFC) 95
Valve and Pump (FbValveAndPump) 99
Humidifier (FbHumidifier) 104

09 Analog Signals 106

Scaling of Input Values 0 to 32767 (FuAI) 106
Scaling of Temperature Values in °C (FuAI_Temp)..... 107
Scaling of Output Values 0 to 32767 (FuAO) 108
1st Order Low-Pass Filter (FbLowPassFilter) 109
1st Order Low-Pass Filter for AI 0 - 32767 (FbLowPassFilterAI)..... 111
1st Order Low-Pass Filter for Temperatures (FbLowPassFilterTemp) ... 113
1st Order Low-Pass Filter for Bus Signals (FbLowPassFilterBus) 115

10 Temperature Evaluation 117

Enthalpy (FbEnthalpy)..... 117
Averaged Outside Temperature (FbAveragedOutsideTemperature)..... 119
Damped Temperature (FbDampedTemperature)..... 121

11 Set Value Adjustment 123

Heating Characteristics (FbHeatingCharacteristics)..... 123
Calculation of Supply Temperature (FbCalculatedSupplyTemperature) 126
Anti-Legionnaire's Disease Function (FbAntiLegionella)..... 129
Overheating and Condensation Protection (FbTemperatureOverride) 132
Optimized Supply Temperature (FbOptimizedSupplyTemperature) 135
Summer Compensation (FuSummerCompensation) 137
Reduced Minimum Fresh Air (FbMinFreshAir)..... 139

12 Boiler Control 141

Boiler Strategy for Two Boilers (Fb2BoilerStrategy) 141
Boiler Module for 2-Level Boiler (Fb2LevelBoiler)..... 145
Boiler Module for Modulating Boiler (FbModulatingBoiler) 158

13 Single Room Control 173

PID Single Room Controller (FbPIDSingleRoomController)..... 173
2-Point Individual Room Controller (Fb2PointSingleRoomController) . 179

14 General Drivers 183

Control of 2-Point Drivers (Fb2PointDriver) 183

Control of Continuous Drivers (FbContinuousDriver) 186
PWM Output (FbPWM) 189
Analog 3-Point Signal (FbAnalog3Point) 191

15 Volume Flow Regulator 193

Volume Flow Regulator (FbVolumeFlowRegulator) 193

16 Additional Functions 198

Blinker (FbBlinker) 198
Blocking Protection for Continuous Drivers (FbBlockingProtectionAnalog) 199
Comfort zone controller (FbHxControl) 201
Hysteresis (FbHysteresis) 205
Impulse Counter (FbImpulseCounter) 207
Min. Value, Avg. Value and Max. Value (FbMinMidMax) 209
Operating Hours Counter (FbOperatingHours_01) 210
Ramp (FbRamp) 211
Status Indication as STRING (FuStatus) 213
Unit monitoring (FbRedundancyUnits) 214

17 Characteristic Curves 217

KTY Characteristic Curve (FuKTY) 217
Two-Point Characteristic (Fu2Point) 218
Four-Point Characteristic Curve (Fu4Point) 219

Important Notes

To ensure fast installation and start-up of the units, we strongly recommend that the following information and explanations are carefully read and adhered to.

Copyright

This document, including all figures and illustrations contained therein, is subject to copyright protection. Any use of this document that infringes upon the copyright provisions stipulated herein is prohibited. Reproduction, translation, electronic and phototechnical filing/archiving (e.g., photocopying), as well as any amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG, Minden, Germany. Non-observance will entail the right of claims for damages.

WAGO Kontakttechnik GmbH & Co. KG reserves the right to make any alterations or modifications that serve to increase the efficiency of technical progress. WAGO Kontakttechnik GmbH & Co. KG owns all rights arising from granting patents or from the legal protection of utility patents. Third-party products are always mentioned without any reference to patent rights. Thus, the existence of such rights cannot be excluded.

Personnel Qualification

The use of the product detailed in this document is geared exclusively to specialists having qualifications in PLC programming, electrical specialists or persons instructed by electrical specialists who are also familiar with the valid standards. WAGO Kontakttechnik GmbH & Co. KG assumes no liability resulting from improper action and damage to WAGO products and third-party products due to non-observance of the information contained in this document.

Intended Use

For each individual application, the components are supplied from the factory with a dedicated hardware and software configuration. Modifications are only permitted within the framework of the possibilities documented in this document. All other changes to the hardware and/or software and the non-conforming use of the components entail the exclusion of liability on part of WAGO Kontakttechnik GmbH & Co. KG.

Please send your requests for modified and new hardware or software configurations directly to WAGO Kontakttechnik GmbH & Co. KG.

Scope of Validity

This application note is based on the stated hardware and software from the specific manufacturer, as well as the associated documentation. This application note is therefore only valid for the described installation.
New hardware and software versions may need to be handled differently.

Please note the detailed description in the specific manuals.

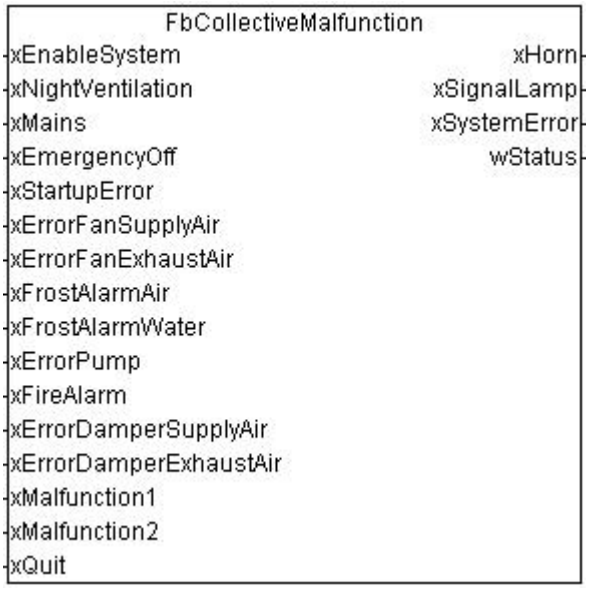
01 System Monitoring

Collective Malfunction (FbCollectiveMalfunction)

WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbCollectiveMalfunction	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnableSystem	BOOL	Enable fault monitoring
xNightVentilation	BOOL	Enable fault monitoring during summer night ventilation
xMains	BOOL	Mains fault
xEmergencyOff	BOOL	Emergency OFF signal
xStartupError	BOOL	Error Preflushing
xErrorFanSupplyAir	BOOL	Fault Incoming air fan
xErrorFanExhaustAir	BOOL	Fault Exhaust air fan
xFrostAlarmAir	BOOL	Antifreeze monitor
xFrostAlarmWater	BOOL	Antifreeze water signal
xErrorPump	BOOL	Error pump heating register
xFireAlarm	BOOL	Fire alarm
xErrorDamperSupplyAir	BOOL	Fault Incoming air damper
xErrorDamperExhaustAir	BOOL	Fault Exhaust air damper
xMalfunction1	BOOL	Error signal 1
xMalfunction2	BOOL	Error signal 2
xQuit	BOOL	Error acknowledgement
Return value:	**Data type:**	**Comment:**
xHorn	BOOL	Horn
xSignalLamp	BOOL	Error indicator lamp
xSystemError	BOOL	System error

wStatus	WORD	Indication of current error message 0 = OK 10 = Error Pump 11 = Main voltage off 12 = Emergency off 13 = Fire alarm 14 = Frost monitor 15 = Frost alarm 20 = Error fan Supply air 21 = Error fan Exhaust air 22 = Malfunction 1 23 = Malfunction 2 27 = Error Supply air damper 28 = Error Preflush 41 = Error Exhaust air damper

Graphical illustration:



Function description:

This function block has been designed to only collect serious errors that would cause a system shutdown.

If the ***"xEnableSystem"*** or ***"xNightVentilation"*** input is activated and one of the inputs ***"xMains", "xEmergencyOff", "xStartupError", "xErrorFanSupplyAir", "xErrorFanExhaustAir", "xFrostAlarmAir", "xFrostAlarmWater", "xErrorPump", "xFireAlarm", "xErrorDamperSupplyAir", "xErrorDamperSupplyAir", "xMalfunction1"*** or ***"xMalfunction2"*** is set to TRUE, an alarm is issued.

The error messages can be either visual or audible messages. An audible error message can be triggered via the ***"xHorn"*** output until the error is acknowledged via the ***"xQuit"*** input. The visual error message can be triggered via the ***"xSignalLamp"*** output. With every error message that appears, the error indicator lamp starts to blink with a frequency of 1 Hz and the horn is activated.

If the error is acknowledged via the *"xQuit"* input, the error indicator lamp will be lit continuously. Only if there is no longer an error at the inputs is it possible to delete the the error message via the *"xQuit*" input.

At the same time, the ***"xSystemError"*** output issues a collective malfunction alarm (non-blinking) that shuts down the system via the **FbStartStop** function block.

The ***"wStatus"*** output provides the specific error message in the order of priority.

Note:

1.) If you also want to receive error messages when the system is turned off, "EnableSystem" should be permanently set *to TRUE.*

2.) The **FuStatus** function converts the "wStatus" status message into a plain text message.

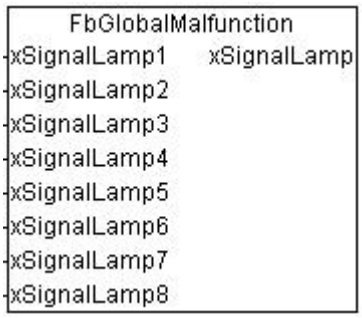
Global Collective Malfunction (FbGlobalMalfunction)

WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbCollectiveMalfunction		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		

Input parameters:	**Data type:**	**Comment:**
xSignalLamp1	BOOL	Signal lamp for FbCollectiveMalfunction
xSignalLamp2	BOOL	Signal lamp for FbCollectiveMalfunction
xSignalLamp3	BOOL	Signal lamp for FbCollectiveMalfunction
xSignalLamp4	BOOL	Signal lamp for FbCollectiveMalfunction
xSignalLamp5	BOOL	Signal lamp for FbCollectiveMalfunction
xSignalLamp6	BOOL	Signal lamp for FbCollectiveMalfunction
xSignalLamp7	BOOL	Signal lamp for FbCollectiveMalfunction
xSignalLamp8	BOOL	Signal lamp for FbCollectiveMalfunction

Return value:	**Data type:**	**Comment:**
xSignalLamp	BOOL	Error indicator lamp

Graphical illustration:

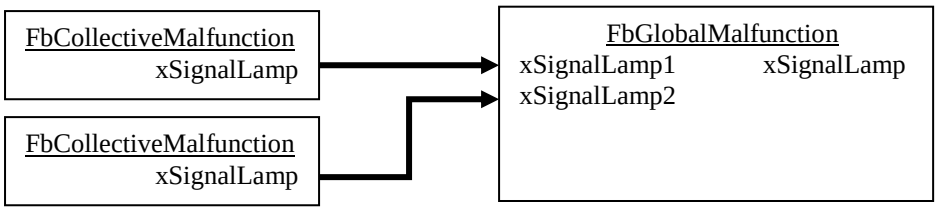


Function description:

The **FbGlobalMalfunction** function block evaluates the error messages from up to eight collective malfunction modules and generates a global collective malfunction message from these.

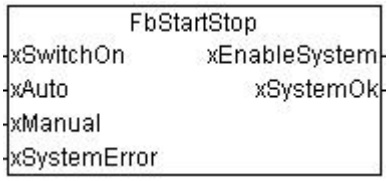
The output signal *"xSignalLamp"* for the FbCollectiveMalfunction function block is linked to the ***"xSignalLampX"*** input for evaluation of the error message.

If one of the collective malfunction modules signals an error, this is indicated at the ***"xSignalLamp"*** output.



Start/Stop (FbStartStop)

WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbStartStop		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
xSwitchOn	BOOL	Switch-on signal in Automatic mode (e.g., from a timer program)	
xAuto	BOOL	Automatic mode	
xManual	BOOL	Manual mode	
xSystemError	BOOL	Collective Malfunction	
Return value:	**Data type:**	**Comment:**	
xEnableSystem	BOOL	Enabling system	
xSystemO.K.	BOOL	System is OK	
Graphical illustration:			



Function description:

This function block serves for switching a HVAC system on and/or off.

The input signals ***"xAuto"*** and ***"xManual"*** are operated by a rotary switch on the switch cabinet and are locked against each other. The rotary switch has the positions: Auto − Off − Manual.

During manual operation, the HVAC system is switched on directly via the ***"xEnableSystem"*** output. During automatic operation, the *"xEnableSystem"* output is switched via the ***"xSwitchOn"*** input (e.g. enabling a clock timer).

If a system malfunction is reported via the ***"xSystemError"*** input, the ***"xEnableSystem"*** and "*xSystemOk"* outputs are set to FALSE. If the malfunction has been corrected and the *"xSystemError"* input is FALSE, the *"xSystemOk"* output is automatically set to TRUE.

Start/Stop Optimization (FbStartStopOptimization)

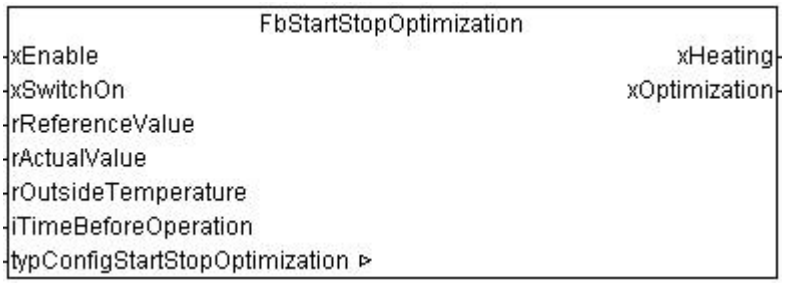
WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbStartStopOptimization	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Library used:	Scheduler_03.lib	

Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable Start/Stop optimization
xSwitchOn	BOOL	Switching signal from the timer program
rReferenceValue	REAL	Reference value room temperature
rActualValue	REAL	Actual value of room temperature
rOutsideTemperature	REAL	Outside temperature
iTimeBeforeOperation	INT	Time before use (+) or duration of use (-)

Input/output parameters:	**Data type:**	**Comment:**
typConfigStartStop Optimization	←	Configuration parameters:
.tStartLowTemperature	TIME	Starting time for outside temperatures >= -10 °C Default setting = t#50m [min/°C]
.tStartHighTemperature	TIME	Starting time for outside temperatures >= 10 °C Default setting = t#20m [min/°C]
.tStopLowTemperature	TIME	Stopping time for outside temperatures >= -10°C Default setting = t#0m [min/°C]
.tStopHighTemperature	TIME	Stopping time for outside temperatures >=10°C Default setting = t#20m [min/°C]
.rVariation	REAL	Tolerance for optimization Default setting = 0.5 [K]
.tMaxTimeBefore Operation	TIME	Max. starting time before operation Default setting = t#9h
.rHolidayOffset	REAL	Percentage increase of starting time after extended outages Default setting = 30
.xAutoCalibration	BOOL	Activate automatic calibration Default setting = TRUE
.xStartOptimization	BOOL	Activate start optimization Default setting = TRUE
.xStopOptimization	BOOL	Activate stop optimization Default setting = FALSE

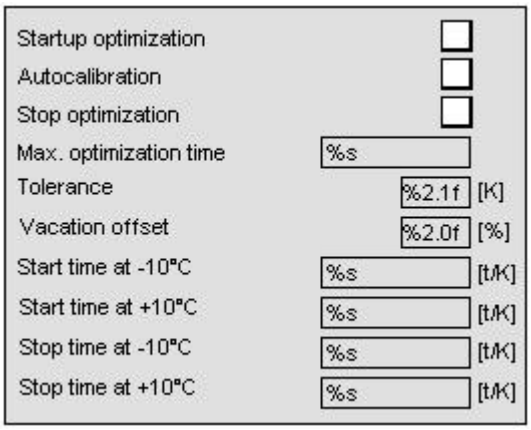
Return value:	Data type:	Comment:
xHeating	BOOL	Enable Heating
xOptimization	BOOL	Display of optimization operating mode

Graphical illustration:



Visualization object:

ConfigStartStop Optimization



Function description:

The **FbStartStopOptimization** function block calculates the optimal start and stop times of a heating installation.

The start time optimization aims to reach the required temperature at the beginning of the service period by starting up the heating on time. The stop time optimization switches the heating off before the end of service. In this process, the temperature may not be/fall below the defined specified temperature.

The optimization function can be deactivated by setting the ***"xEnable"*** input to FALSE signal. In this case, the ***"xSwitchOn"*** is linked directly to the ***"xHeating"*** output.

The time remaining until the service period ***"iTimeBeforeOperation"*** begins, or the remaining time up to the end of the service period, is determined by the **FbScheduler** function block from the *"Scheduler_03.lib"* library.

Configuration parameters:

The configuration structure ***"typConfigStartStopOptimization"*** contains the following parameters:

- ***".xAutoCalibration"*** enables automatic adaptation of the characteristic curve.
- ***".xStartOptimization"*** enables the "Start optimization" function.
- ***".xStopOptimization"*** enables the "Stop optimization" function. Stop optimization only functions when Start optimization has been enabled.
- ***".tStartLowTemperature"*** is the bottom grid point in the Start optimization curve and is adapted (calibrated) automatically by Self-optimization.
- ***".tStartHighTemperature"*** is the top grid point in the Start optimization curve and is adapted (calibrated) automatically by Self-optimization.
- ***".tStopLowTemperature"*** is the bottom grid point in the Stop optimization curve and is adapted automatically by Self-optimization.
- ***".tStopHighTemperature"*** is the top grid point in the Stop optimization curve and is adapted automatically by Self-optimization.
- ***".rVariation"*** indicates the tolerance for deviation.
- ***".tMaxTimeBeforeOperation"*** indicates the maximum time before beginning of service for Start optimization.
- ***".rHolidayOffset"*** adds an offset percentage to the calculated starting time (see Holiday Effect).

Start Time Optimization

If the beginning of the normal start time has not been reached yet, the function block calculates the optimum start time according to the characteristic curve shown in Fig. 1.

The characteristic curve gives the starting time per degrees Kelvin of deviation between the ***"rReferenceValue"*** reference and the actual value ***"rActualValue"***. The dependency on the current outside temperature ***"rOutsideTemperature"*** is also taken into account.

Example: navigation of level 3.

tStartLowTemperature = t#50m min/K
tStartHighTemperature = t#10m min/K
rOutsideTemperature = 0°C
rActualValue = 18°C
rReferenceValue = 20°C

For example, a start time of 30 min. is yielded from the characteristic curve (Fig. 1) at an outside temperature of 0°C.

$$\Longrightarrow \text{ Start time} = (20\ °C - 18\ °C) * 30 \frac{min}{°C} = \underline{60\ min}$$

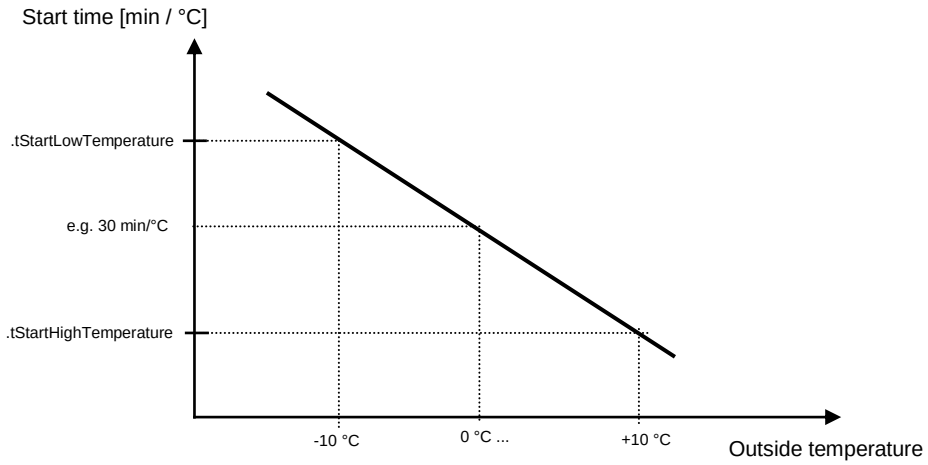


Fig.1: Start time in relation to the outside temperature

By starting up the heating installation on time, the required reference temperature can be reached at the beginning of the service period. If the remaining time until the service period starts is smaller than the calculated start time, the "*xHeating"* and ***"xOptimization"*** outputs are switched to TRUE (see Fig. 2).

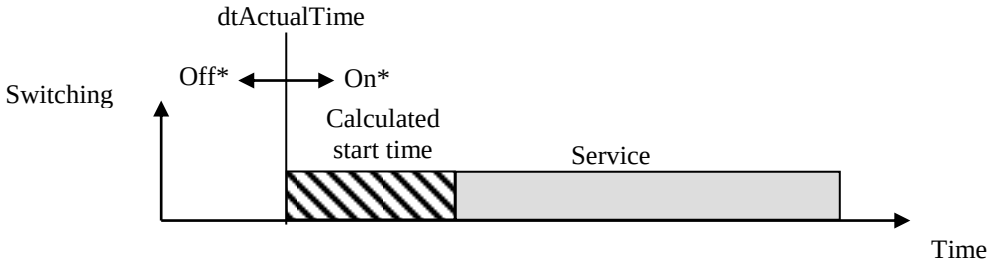


Fig. 2: Start-up time in relation with the calculated start time

The ***"xOptimization"*** output signal is reset to FALSE when the required reference temperature, minus the *".rVariation"* tolerance, is reached, or when the "normal" service period begins. This shows that the start time optimization is finished.

Ideally, the reference temperature is reached when the service period begins. If the room temperature is reached too early or too late, the characteristic curve can be adjusted automatically by shifting the grid points. This ensures that the thermal characteristics of the building are identified by the function block.

Automatic correction of the grid points will not be performed if the installation is switched off for more than 20 hours (see Holiday Effect).

Public Holiday Offset

If the installation is switched off for more than 20 hours (e.g. on weekends or public holidays), a percentage offset is added to the calculated start time as a longer heat-up phase is required.

The percentage offset is calculated from a characteristic curve (Fig. 3). The maximum value for the offset is reached after the heating installation has been switched off for 48 hours.

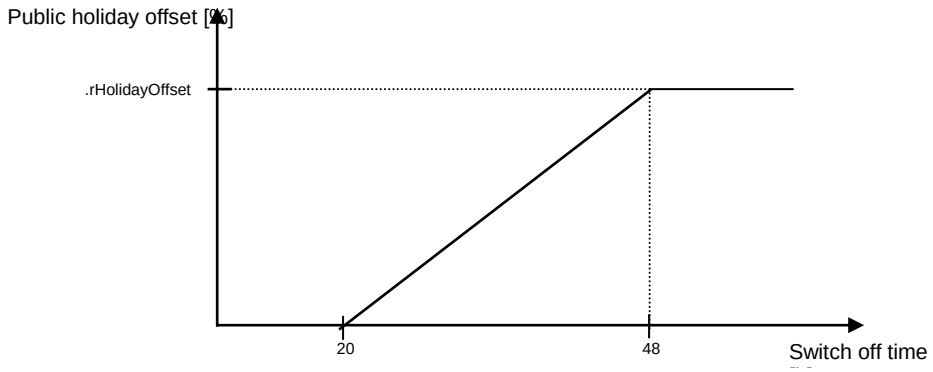


Fig.3: Public holiday offset as a function of the off time

Stop Time Optimization

The stop time optimization aims to save energy by switching off the heating installation before the service period is finished. In this case, the room temperature must not fall beyond a defined value during the period of use. The limit for the room temperature at the end of the service period is yielded from the reference value, minus the tolerance value (*"rReferenceValue" - ".rVariation"*).

The stop time is calculated from the characteristic curve shown in Fig. 4, which describes the relation between the outside temperature and the stop time. The characteristic curve indicates the stop time per Kelvin of deviation between the current room temperature and the room temperature limit at the end of the service period.

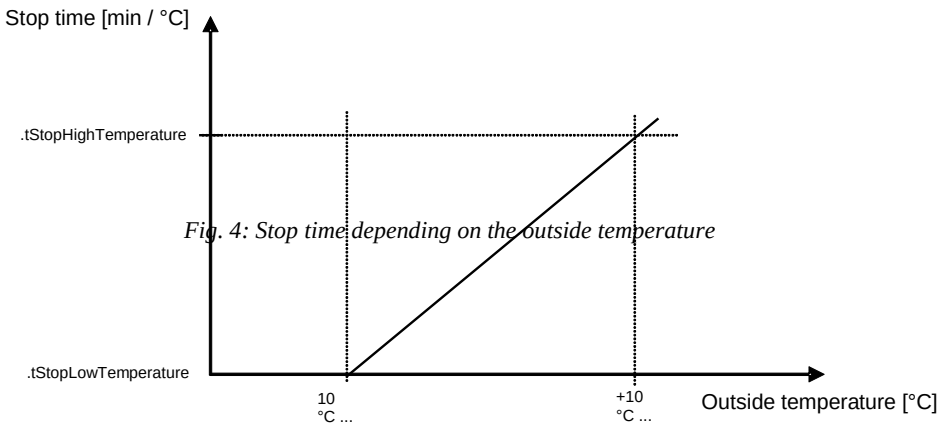


Fig. 4: Stop time depending on the outside temperature

Start/Stop Heating Circuit Control (FbStartStopHeatingCircuit)

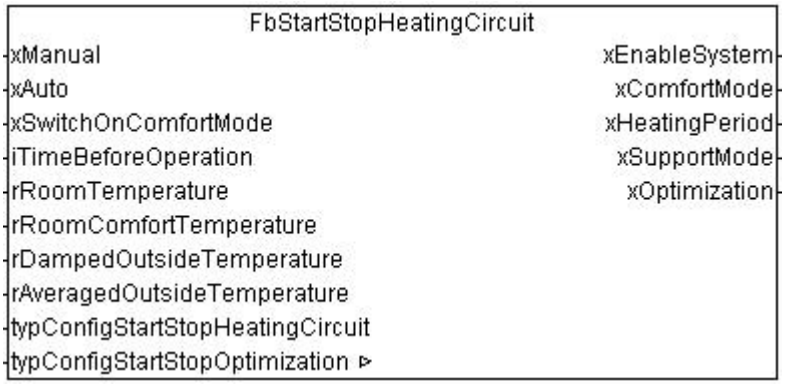
WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbStartStopHeatingCircuit		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Library used:	Scheduler_03.lib		

Input parameters:	**Data type:**	**Comment:**
xManual	BOOL	Manual mode
xAuto	BOOL	Automatic mode
xSwitchOnComfortMode	BOOL	Switch-on signal in Automatic mode (e.g., from a timer program)
iTimeBeforeOperation	INT	Time before use (+) or duration of use (-)
rRoomTemperature	REAL	Actual value room temperature [°C]
rRoomComfort Temperature	REAL	Reference value for room temperature, day-time mode [°C] Default setting = 20
rDampedOutside Temperature	REAL	Damped outside temperature [°C]
rAveragedOutside Temperature	REAL	Averaged outside temperature [°C]
typConfigStartStopHeating Circuit	←	Configuration parameters:
.rLimitDampedOutside TemperatureComfort	REAL	Limit for damped outside temperature, Comfort mode [°C] Default setting = 18
.rLimitDampedOutside TemperatureStandby	REAL	Limit for damped outside temperature, Standby mode [°C] Default setting = 18
.rLimitAveragedOutside TemperatureComfort	REAL	Limit for averaged outside temperature, Comfort mode [°C] Default setting = 16
.rLimitAveragedOutside TemperatureStandby	REAL	Limit for averaged outside temperature, Standby mode [°C] Default setting = 16
.rMinRoomTemperature	REAL	Limit for room temperature for support mode [°C] Default setting = 13
.rHysteresisMinRoom Temperature	REAL	Hysteresis for support mode [K] Default setting = 2
.xEconomyMode	BOOL	Night-time economy mode or overnight shutdown Default setting = TRUE (Night-time economy mode)
.xRoomTemperature Sensor	BOOL	Room temperature sensor present Default setting = TRUE

Input/output parameters:	Data type:	Comment:
typConfigStartStop Optimization	←	Configuration parameters for Start/Stop optimization (see description for FbStartStopOptimization function block)

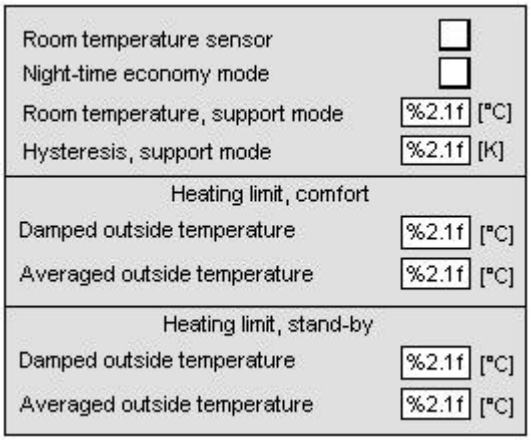
Return value:	Data type:	Comment:
xEnableSystem	BOOL	Enable for heating circuit control system
xComfortMode	BOOL	Heating circuit in Comfort mode
xHeatingPeriod	BOOL	Heating limit not attained (heating period)
xSupportMode	BOOL	Heating circuit in the Support mode
xOptimization	BOOL	Heating circuit in Optimization mode

Graphical illustration:

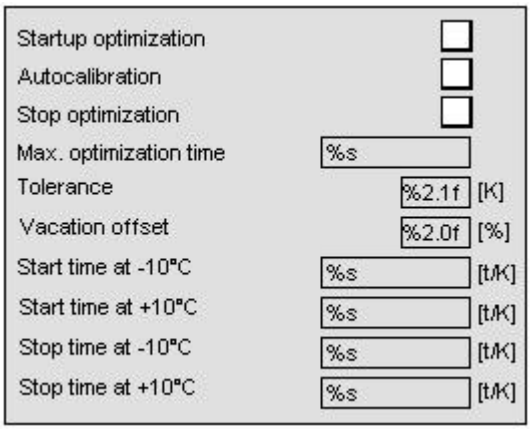


Visualization objects:

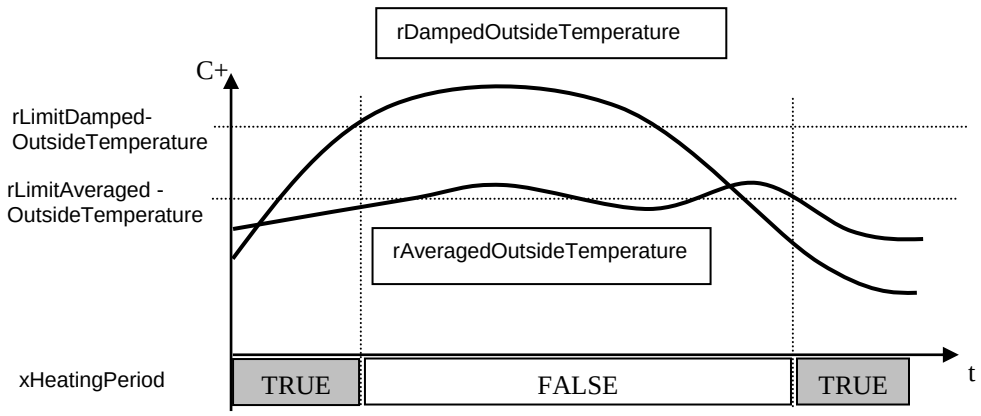
ConfigStartStopHeating Circuit



ConfigStartStop Optimization



Time referenced behavior:



Function description:

The **FbStartStopHeatingCircuit** function block is used for switching the heating circuit on/off. Start optimization, a heating limit based on the outside temperature and a Support mode have been implemented to determine the optimal on/off times.

The input signals ***"xAuto"*** and ***"xManual"*** are operated by a rotary switch on the switch cabinet and are locked against each other. The rotary switch has the positions: Auto − Off − Manual.

The heating circuit is enabled via the ***"xEnableSystem"*** output when one of the following conditions is met:

1.) *"xManual"*

2.) *"xAuto"*, ***"xHeatingPeriod"*** and *"xSwitchOnComfortMode"*

3.) *"xAuto", "xHeatingPeriod"* and ***".xEconomyMode"*** (Night-time economy mode)

When the heating circuit has been enabled, the Comfort mode is activated via the ***"xComfortMode"*** output when one of the following conditions is met:

1.) *"xManual"*

2.) *"xAuto"* and *"xSwitchOnComfortMode"* (e.g., enabled from a timer program)

The set Comfort mode temperature is specified via the ***"rRoomComfortTemperature"*** input.

A room temperature sensor is required to determine an optimal starting time. The characteristic curve for start optimization is adjusted automatically at the beginning of use as a function of the difference between the current room temperature ***"rRoomTemperature"*** and the reference room temperature *"rRoomComfortTemperature"*. Automatic adjustment of the characteristic curve cannot take place for start optimization without a room temperature sensor.

Premature activation of the heating circuit by start optimization is indicated at the ***"xOptimization"*** output.

The time remaining until the beginning of use is communicated to the block via the ***"iTimeBeforeOperation"*** input. A more detailed description of Start optimization is given in the documentation for the **FbStartStopOptimization** function block.

The heating circuit is switched on in the Support mode if the room temperature *"rRoomTemperature"* drops below the minimum room temperature "*rMinRoomTemperature*". The Support mode is indicated at the ***"xSupportMode"*** output.

Two different values are taken into account for detecting a heating period. If the averaged outside temperature ***"rAveragedOutsideTemperature"*** and the damped outside temperature ***"rDampedOutsideTemperature"*** fall below the defined limit, the heating period is enabled via the *"xHeatingPeriod"* output. The heating period is terminated if one of these two values rise above the defined limit again.

Configuration parameters:

The configuration structure ***"typConfigStartStopHeatingCircuit"*** contains the following parameters:

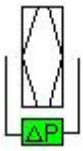
- ***".rLimitDampedOutsideTemperatureComfort"*** is the limit for the damped outside temperature for determining the heating period (Comfort mode).
- ***".rLimitDampedOutsideTemperatureStandby"*** is the limit for the damped outside temperature for determining the heating period (Standby mode).
- ***".rLimitAveragedOutsideTemperatureComfort"*** is the limit for the averaged outside temperature for determining the heating period (Comfort mode).
- ***".rLimitAveragedOutsideTemperatureStandby"*** is the limit for the averaged outside temperature for determining the heating period (Standby mode).
- ***".rMinRoomTemperature"*** is the limit for the Support mode.
- ***".rHysteresisMinRoomTemperature"*** is the hysteresis for the *".rMinRoomTemperature"* limit.
- ***".xEconomyMode"*** indicates which economy mode is to be used. TRUE = Night-time economy mode, FALSE = Overnight shutdown.
- ***".xRoomTemperatureSensor"*** indicates whether a room temperature sensor is present for minimum room temperature monitoring (Support mode) or for Start optimization.

The configuration structure ***"typConfigStartStopOptimization"*** is described in the documentation for the **FbStartStopOptimization** function block.

Note:

When the parameter *".xRoomTemperatureSensor"* is activated, the parameter ***„typConfigStartStopOptimization.xAutoCalibration“*** is activated at the same time.

Filter Monitoring (FbFilterMonitoring)

WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbFilterMonitoring		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
xFilter	BOOL	Filter maintenance signal	
tOnDelay	TIME	Response delay Default setting: t#10s	
xQuit	BOOL	Error message acknowledgement	
Return value:	**Data type:**	**Comment:**	
xMaintenance	BOOL	Warning signal filter dirty	
Graphical illustration:			
Visualization objects:			
ConfigFilterMonitoring			
AirFilter	 You can select for visualization of the filter whether the filter is to be located in the supply (incoming) air or in the exhaust air.		

Function description:

The filters are normally monitored using differential pressure monitors. The differential pressure monitors report fouling of the filter system via the inputs ***"xFilter"***.

An On-delay ***"tOnDelay"*** can be defined for the *"xFilter"* input to prevent fouling from being signaled in the duct when pressure fluctuations occur.

Fouling of the filter is indicated via the ***"xMaintenance"*** output.

If the differential pressure no longer report fouling of the filter, the message can be acknowledged via a flank at the ***"xQuit"*** input.

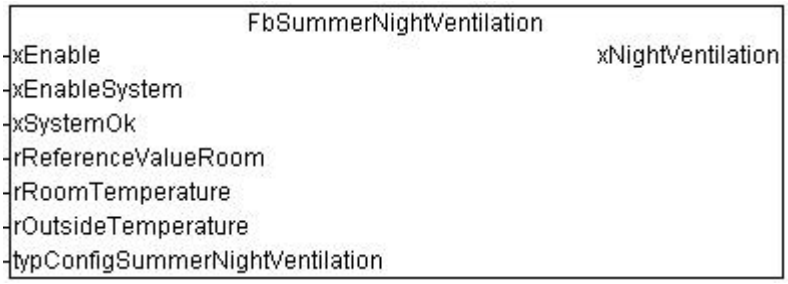
Summer Night Ventilation (FbSummerNightVentilation)

WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbSummerNightVentilation	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	

Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enabling summer night ventilation
xEnableSystem	BOOL	Enable FbStartStop
xSystemOK	BOOL	System check-back signal from FbStartStop
rReferenceValueRoom	REAL	Reference value room temperature [°C] Default setting = 20 °C
rRoomTemperature	REAL	Actual value room temperature [°C]
rOutsideTemperature	REAL	Actual value outside temperature [°C]
typConfigSummerNight Ventilation	←	Configuration parameters:
.rMinDiffRoom	REAL	Minimum difference between reference and actual value of room temperature [K] Default setting = 2 K
.rMinDiffRoomOutside	REAL	Minimum difference between room temperature and outside temperature [K] Default setting = 5 K
.rMinOutside Temperature	REAL	Minimum temperature for summer night ventilation [°C] Default setting = 12 °C
.rHysteresis	REAL	Hysteresis for limit values Default setting = 1 K

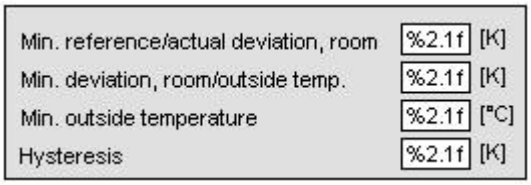
Return value:	**Data type:**	**Comment:**
xNightVentilation	BOOL	Output signal of the summer night ventilation

Graphical illustration:

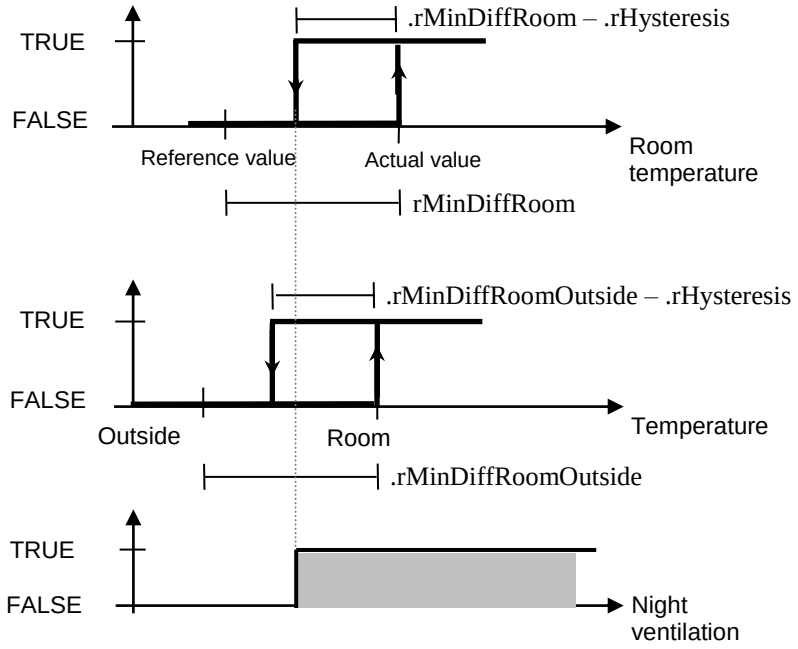


Visualization object:

ConfigSummerNight Ventilation



Time referenced behavior:



Function description:

Summer often offers the possibility of cooling down the room temperature with the cool night air. The **FbSummerNightVentilation** function block is used to utilize the possibility of effective night cooling to control the unit components necessary for cooling.

Configuration parameters:

The configuration structure ***"typConfigSummerNightVentilation"*** contains the following parameters:

- ***".rMinDiffRoom"*** is the minimum difference between the reference and actual room temperature.
- ***".rMinDiffRoomOutside"*** is the minimum difference between the room temperature and the outside temperature.
- ***".rMinOutsideTemperature"*** is the minimum outside temperature for using summer night air for cooling.
- ***".rHysteresis"*** is the hysteresis for these three limits.

Starting conditions for night ventilation:

The following points must all be fulfilled before night cooling (ventilation) is enabled via ***"xNightVentilation"***:

- ***"xEnable"*** = TRUE
- ***"xEnableSystem"*** = FALSE
- ***"xSystemOK"*** = TRUE
- The difference between the specified temperature ***"rReferenceValueRoom"*** and the actual temperature ***"rRoomTemperature"*** must be greater than the limit *".rMinDiffRoom".*
- The difference between the room temperature *"rRoomTemperature"* and the outside temperature *"rOutsideTemperature"* must be greater than the limit value *"rMinDiffRoomOutside"*
- The outside temperature *"rOutsideTemperature"* must be greater than *"rMinOutsideTemperature".*

Stop conditions for night ventilation:

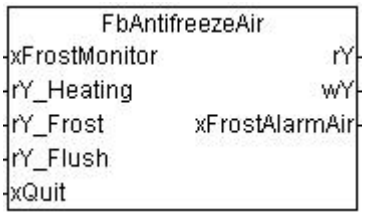
Night ventilation is terminated when one of the following conditions is fulfilled:

- *"xEnable"* = FALSE
- *"xEnableSystem"* = TRUE
- *"xSystemOK"* = FALSE
- The difference between the room temperature *"rRoomTemperature"* and the outside temperature *"rOutsideTemperature"* is less than *".rMinDiffRoomOutside"*, minus *".rHysteresis".*
- The difference between the specified temperature *".rReferenceValueRoom"* and the actual temperature *"rRoomTemperature"* is less than *".rMinDiffRoom ".*
- The outside temperature *"rOutsideTemperature"* is less than *"rMinOutsideTemperature"*, minus *".rHysteresis".*

02 Antifreeze Protection

Air-Side Antifreeze Protection (FbAntifreezeAir)

WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbAntifreezeAir		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
xFrostMonitor	BOOL	Signal of the freeze protection test point Default setting = TRUE	
rY_Heating	REAL	Set value from controller [%] Value range = 0 – 100	
rY_Frost	REAL	Set value antifreeze control [%] Value range = 0 – 100	
rY_Flush	REAL	Set value for antifreeze water (preflushing) [%] Value range = 0 – 100	
xQuit	BOOL	Error acknowledgement	
Return value:	**Data type:**	**Comment:**	
rY	REAL	Set value heating valve [%] Value range: 0 −100	
wY	WORD	Set value for heating valve Value range = 0 – 32767	
xFrostAlarmAir	BOOL	Frost alarm is active	
Graphical illustration:			



Function description:
The **FbAntifreezeAir** function block controls the temperature in the air intake by means of a freeze protection device and determines the maximum setting value for the heating register. If the air-side antifreeze ***"xFrostMonitor"*** is activated, the valve for the heating register is opened 100%. In a non-faulted state, the maximum value for inputs ***"rY_Heating"***, "***rY_Flush"*** and ***"rY_Frost"*** arrive at the ***"rY"*** output. The output value ***"wY"*** has the same meaning as the *"rY"* output, only the output has standardized values between 0 – 32767. The ***"xFrostAlarmAir"*** output ensures that the HVAC system is switched off via the **FbCollectiveMalfunction** function block and that the pump for the heating register is switched on as a frost protection measure. If the antifreeze protection device no longer reports an error, the warning message can be acknowledged via a flank at the ***"xQuit"*** input.

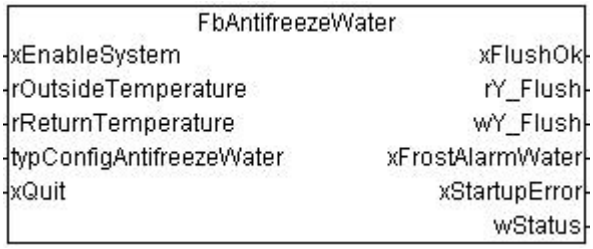
WAGO Kontakttechnik GmbH & Co. KG
P:O. box 2880 • D-32385 Minden
Hansastr. 27 • D-32423 Minden
Phone: 05 71 / 8 87-0
Fax.: +49 (0) 5 71 / 8 87-169
E-Mail: info@wago.com
Web: http://www.wago.com

Water-Side Antifreeze Protection (FbAntifreezeWater)

WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbAntifreezeWater	
Type:	Function ☐	Function block ☒ Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnableSystem	BOOL	Enable antifreeze system water Default setting = TRUE
rOutsideTemperature	REAL	Actual value outside temperature [°C]
rReturnTemperature	REAL	Actual value return temperature [°C]
xQuit	BOOL	Error acknowledgement
typConfigAntifreezeWater	←	Configuration parameters:
.rLimitOutside TemperatureFlush	REAL	Maximum outside temperature for flush activation [°C] Default setting = 10 °C
.rY_Standby	REAL	Valve opening for preventive frost protection [%] Default setting = 5
.rLimitFrostAlarm	REAL	Limiting value water temperature for frost alarm [°C] Default setting = 5 °C
.tMaxFlushPeriod	TIME	Maximum flush time Default setting = t#15 m
.rLimitValueWaterStart Up	REAL	Minimum return temperature for terminating preflushing [°C] Default setting = 30 °C
.rReferenceValueReturn	REAL	Reference value for antifreeze protection controller return temperature [°C] Default setting = 15 °C
.tFollowUpTime	TIME	Off-delay for reducing the set value to zero Default setting = t#10 m
.rPercentAfterStartUp	REAL	Starting value for the gradient after terminating preflushing [%] Default setting = 50%
.rLimitOutside TemperatureStandby	REAL	Outside temperature limit for the Standby set value [%] Default setting = 2
.xReturnSensor	BOOL	Return temperature sensor present Default setting = FALSE
Return value:	**Data type:**	**Comment:**
xFlushOK	BOOL	Flush process completed
rY_Flush	REAL	Set value for flush process [%] Value range: 0 −100

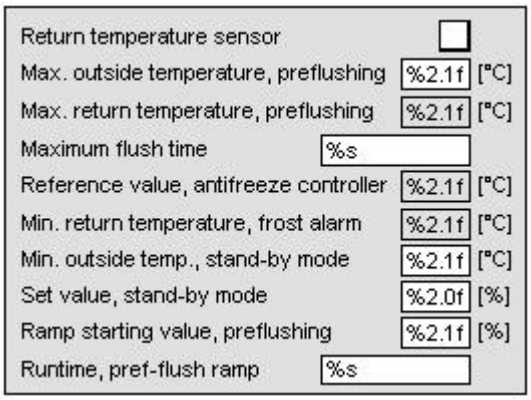
wY_Flush	WORD	Set value for the flush process Value range = 0 – 32767
xFrostAlarmWater	BOOL	The return temperature has fallen below the frost alarm limit
xStartupError	BOOL	Flush process error
wStatus	WORD	Display current status 0 = OK 18 = No hot water 24 = Preflushing completed 25 = Preflushing of heating elements 15 = Frost alarm

Graphical illustration:

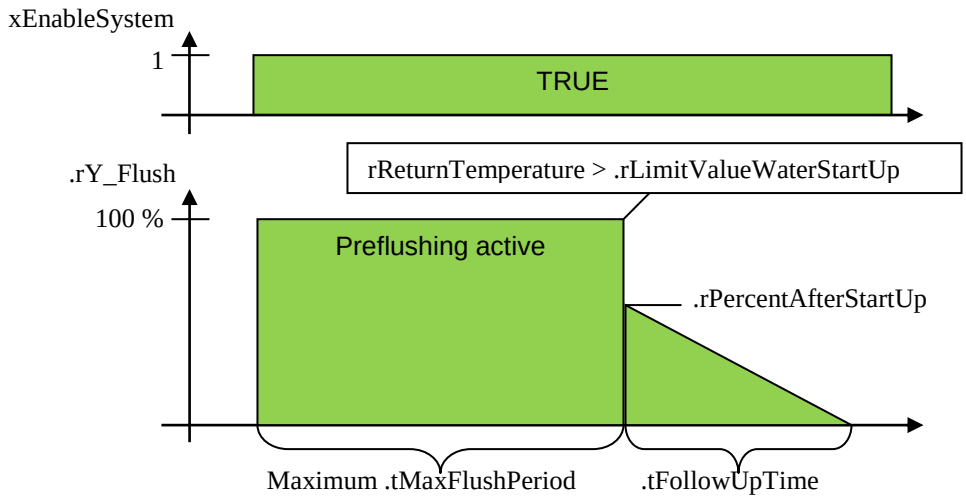


Visualization object:

ConfigAntifreezeWater



Time referenced behavior:



Function description:

The antifreeze water serves as a preventive frost protection by flushing the preheater and sends an error message in case of freeze danger (only with return sensor).

Configuration parameters:

The configuration structure ***"typConfigAntifreezeWater"*** contains the following parameters:

- ***".rLimitOutsideTemperatureFlush"*** specifies the limit, starting from which the heating register is to be flushed.
- ***".tMaxFlushPeriod"*** defines the maximum time for flushing when a return temperature is present. If no return temperature is present this time will be used for the duration of the flushing process.
- ***".rPercentAfterStartUp"*** defines the start value for "rY_Flush" after flushing. This start value is reduced to 0% via a ramp.
- ***".tFollowUpTime"*** defines the time after flushing in which the set value is reduced from *".rPercentAfterStartUp"* to zero (ramp).
- ***".xReturnSensor"*** indicates whether a return temperature sensor is present. Flushing is performed time-controlled if no return temperature sensor is present.
- ***".rLimitValueWaterStartUp"*** specifies the limit after which flushing is terminated. At the same time, this limit is also used for enabling the antifreeze controller.
- ***".rReferenceValueReturn"*** indicates the reference value for the antifreeze controller.
- ***".rLimitOutsideTemperatureStandby"*** specifies the limit after which the valve is moved to the standby position.
- ***".rY_Standby"*** indicates the valve position for the heating register when the system is shut down and the outside temperature falls below the limit "*.rLimitOutsideTemperatureStandby"*.
- ***".rLimitFrostAlarm"*** indicates the minimum return temperature for the frost alarm.

Antifreeze water is activated via the ***"xEnableSystem"*** input.

Flushing of the heating register is carried out only when the outside temperature ***"rOutsideTemperature"*** falls below the set limit for flushing. If the outside temperature is above the limit temperature flushing is not performed and the ***"xFlushOK"*** output is directly activated.

During flushing of the heating register, the ***"rY_Flush"*** output is set to 100% until the adjustable limit temperature limit in the return line is exceeded. When this limit temperature is exceeded, the "*xFlushOK"* output is activated.

If the return temperature fails to reach the limit temperature within the set delay period (no hot water), an error message is issued at the ***"xStartupError"*** output and the valve opened 100%.

After flushing, the "*rY_Flush"* output is set to a defined value and reduced to 0% via a definable ramp.

Even when it is switched off the antifreeze controller regulates the return temperature to a minimum reference value. The antifreeze controller is active as long as the return temperature remains below the limit for terminating the flushing process.

If the return temperature falls below the limit for the frost alarm, there is a risk of freezing and the alarm ***"xFrostAlarmWater"*** is issued. Additionally, the set value for the heating register *"rY_Flush"* is set to 100%.

The output value ***"wY_Flush"*** has the same meaning as the *"rY_Flush"* output, except that the output has standardized values between 0 – 32767.

The current status for antifreeze water protection is indicated via the ***"wStatus"*** output.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

Note:

1.) If no return temperature sensor is present, flushing is performed in a time-controlled manner.

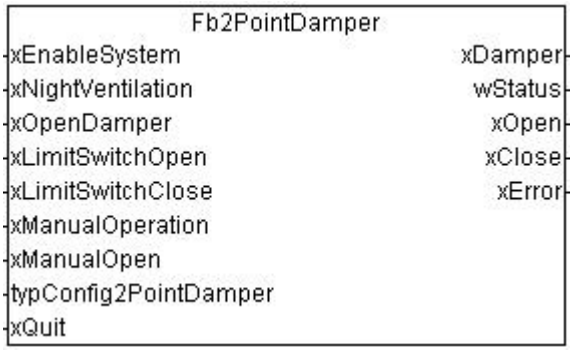
2.) The **FuStatus** function converts the "wStatus" status message into a plain text message.

03 Damper Control

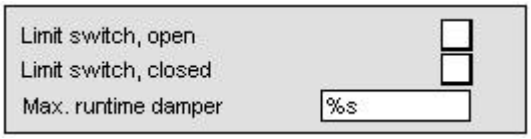
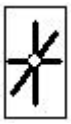
Actuation of 2-Point Dampers (Fb2PointDamper)

WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	Fb2PointDamper	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnableSystem	BOOL	Enable FbStartStop Default setting = TRUE
xNightVentilation	BOOL	Opening dampers during night ventilation
xOpenDamper	BOOL	Open the damper Default setting = TRUE
xLimitSwitchOpen	BOOL	Check-back signal from limit switch (open)
xLimitSwitchClose	BOOL	Check-back signal from limit switch (closed)
xManualOperation	BOOL	Enable manual operation
xManualOpen	BOOL	Open or close damper manually Open = TRUE
typConfig2PointDamper	←	Configuration parameters:
.tMaxRuntime	TIME	Maximum runtime of the damper Default setting: t#30s
.xLimitSwitchOpen	BOOL	Activate control of check-back signal from limit switch (open) Default setting = TRUE
.xLimitSwitchClose	BOOL	Activate control of check-back signal from limit switch (closed) Default setting = FALSE
xQuit	BOOL	Error message acknowledgement
Return value:	**Data type:**	**Comment:**
xDamper	BOOL	Damper control
wStatus	WORD	Display current status 0 = OK 3 = Open 4 = Closed 36 = In motion 42 = Error Damper position
xOpen	BOOL	Damper is open
xClose	BOOL	Damper is closed
xError	BOOL	Damper fault

Graphical illustration:



Visualization objects:

Config2PointDamper	
Config2PointDamperRWT	
TwoPointDamper	

Function description:

The **Fb2PointDamper** function block is used to control 2-point dampers with optional limit switches.

<u>Configuration parameters:</u>

The configuration structure ***"typConfig2PointDamper"*** contains the following parameters:

- ***".tMaxRuntime"*** monitors the maximum runtime for the damper when limit switches are provided. If no limit switches are present, this parameter is used for the runtime of the damper.
- ***".xLimitSwitchOpen"*** indicates whether a limit switch is available for "Damper open".
- ***".xLimitSwitchClose"*** indicates whether a limit switch is available for "Damper closed".

The damper is opened in the Automatic mode when the system has been enabled via ***"xEnableSystem"*** and the ***"xOpenDamper"*** input has been activated.

During night ventilation, the damper can also be opened independently of this enable via the ***"xNightVentilation"*** input.

When the Manual mode is activated via the ***"xManualOperation"***, the damper is controlled via the ***"xManualOpen"*** input.

The damper adjusting motor is controlled via the ***"xDamper"*** output.

The runtime of the damper is monitored when limit switches are provided for each direction of movement. When the maximum runtime is exceeded, the damper is closed and the ***"xError"*** output activated.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

The ***"xOpen"*** and ***"xClose"*** outputs indicate the status of the damper (opened/closed).

The current status for the damper is indicated via the ***"wStatus"*** output.

Note:

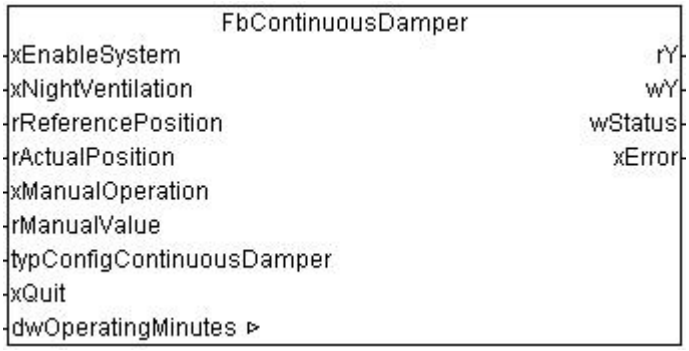
1.) If no limit switch is provided, the damper position is determined over time.

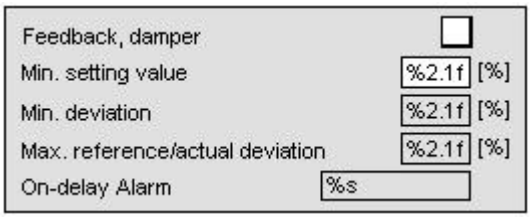
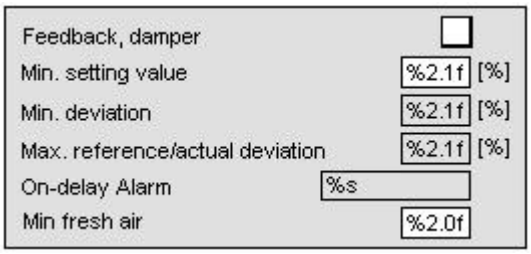
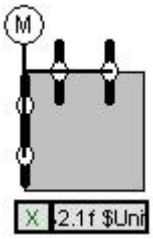
2.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

Actuation of Continuous Dampers (FbContinuousDamper)

WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbContinuousDamper	
Type:	Function ☐	Function block ☒ Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnableSystem	BOOL	Enable FbStartStop Default setting = TRUE
xNightVentilation	BOOL	Opening damper during night ventilation
rReferencePosition	REAL	Reference position of the continuous damper [%]
rActualPosition	REAL	Actual position of the continuous damper [%]
xManualOperation	BOOL	Enable manual operation
rManualValue	REAL	Set value manual operation [%] Value range = 0 – 100%
typConfigContinuous Damper	←	Configuration parameters:
.tOnDelayAlarm	TIME	ON-delay for alarm Default setting = t#5 m
.rMinDeviationMovement	REAL	Minimum deviation for detection of motion [%] Default setting = 5
.rTolerance	REAL	Max. deviation, specified/actual position [%] Default setting = 3
.rY_Min	REAL	Minimum setting value for damper [%] Default setting = 3
.xFeedbackDamper	BOOL	Check-back signal Damper position present Default setting = FALSE
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating minutes for continuous damper
Return value:	**Data type:**	**Comment:**
rY	REAL	Set value for damper Value range = 0 – 100
wY	WORD	Set value for dampler Value range: 0 −32767

wStatus	WORD	Display current status 0 = OK 1 = On 2 = Off 36 = In motion 42 = Error Damper position
xError	BOOL	Damper fault
Graphical illustration:		



Visualization objects:	
ConfigContinuous Damper	
ConfigMixedAir Damper	
MixedAirDamper	

Function description:

The **FbContinuousDamper** function block is used for controlling continuous dampers. As an option, the damper position can also be monitored by the function block.

Configuration parameters:

The configuration structure ***"typConfigContinuousDamper"*** contains the following parameters:

- ***".tOnDelayAlarm"*** defines the time period that can elapse until an alarm is issued for a permanent deviation of the driver position.
- ***".rMinDeviationMovement"*** defines the minimum deviation between the specified position ***"rReferencePosition"*** and the actual position ***"rActualPosition"*** for movement detection.
- ***".rTolerance"*** defines the permissible deviation between specified/actual position for position monitoring
- ***".rY_Min"*** defines the set value that must at least be reached to change the damper position.
- ***".xFeedbackDamper"*** indicates whether a continuous damper check-back signal is provided.

Damper control is enabled via the ***"xEnableSystem"*** input.

During night ventilation, the damper can also be opened independently of this enable via the ***"xNightVentilation"*** input.

When the Manual mode is activated via the input ***"xManualOperation"***, the damper is controlled via the ***"rManualValue"*** input.

The damper adjusting motor is controlled via the ***"rY"*** output.

The output value ***"wY"*** has the same meaning as the *"rY"* output, the output just has the standardized values between 0 – 32767.

When the position check-back signal is present with a permanent position deviation, the damper is closed and the ***"xError"*** is activated when the delay period is exceeded.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

The current status for the damper is indicated via the ***"wStatus"*** output.

The input/output variable ***"dwOperatingMinutes"*** indicates the operating time in minutes for the continuous damper. The operating minutes are counted when *"rY" is greater than ".rY_Min".*

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

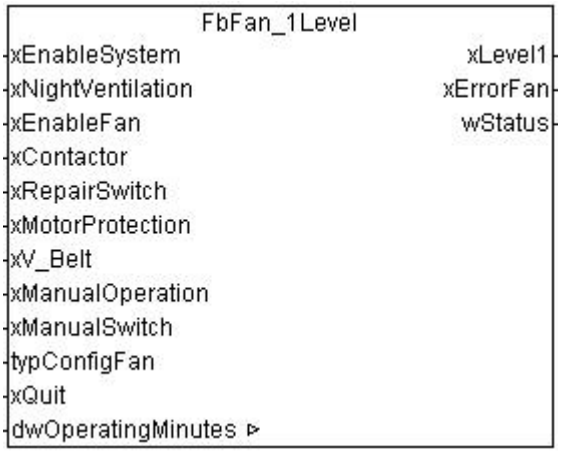
04 Fan Control

Fan, 1-Level (FbFan_1Level)

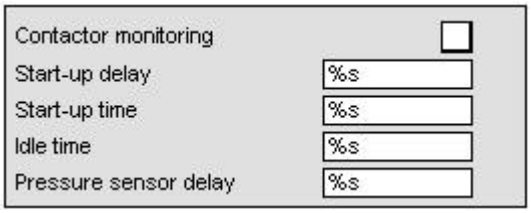
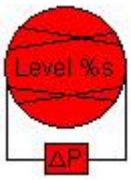
WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbFan_1Level	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnableSystem	BOOL	Fan control enabled by FbStartStop Default setting = TRUE
xNightVentilation	BOOL	Switch on fan during night ventilation
xEnableFan	BOOL	Switch on fan
xContactor	BOOL	Contactor monitoring via auxiliary contact
xRepairSwitch	BOOL	Repair switch Default setting = TRUE
xMotorProtection	BOOL	Motor protection switch Default setting = TRUE
xV_Belt	BOOL	V-belt monitoring of the fan Default setting = TRUE
xManualOperation	BOOL	Enable manual operation
xManualSwitch	BOOL	Switch on fan manually
typConfigFan	←	Configuration parameters:
.tOnDelay	TIME	Delay time of the fan Default setting: t#0s
.tStartUpPeriod	TIME	Startup time of the fan Default setting: t#5s
.tSwitchOverTime	TIME	Without function
.tPressureVariation	TIME	Error message delay time during pressure fluctuations Default setting: t#5s
.xAuxiliaryContact	BOOL	Auxiliary contact for contactor monitoring present Default setting = FALSE
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating minutes of the fan
Return value:	**Data type:**	**Comment:**
xLevel1	BOOL	Switch-on signal for the fan
xErrorFan	BOOL	Error message fan

wStatus	WORD	Display current status 0 = OK 16 = Repair switch 17 = Motor protection switch 19 = V-belt 26 = Error Contactor contact

Graphical illustration:



Visualization objects:

ConfigFan_xLevel	
Fan_xLevel	 Attention: The idle time is not required with a single-level fan!

Function description:

The **FbFan_1Level** function block controls and monitors a 1-level fan.

Configuration parameters:

The configuration structure ***"typConfigFan"*** contains the following parameters:

- ***".tOnDelay"*** defines the On-delay for the fan.
- ***".tStartUpPeriod"*** defines the runup time for the fan. During this time period V-belt monitoring is not performed.
- ***".tPressureVariation"*** defines the response delay for V-belt monitoring.
- ***".xAuxiliaryContact"*** indicates whether a check-back signal from the power contactor auxiliary contact is present.

The fan is switched on the Automatic mode when the system has been enabled via ***"xEnableSystem"*** and the ***"xEnableFan"*** input has been activated.

During night ventilation, the fan can also be switched on independently of this enable via the ***"xNightVentilation"*** iniput.

When the Manual mode is activated via the input ***"xManualOperation"***, the fan is actuated via the ***"xManualSwitch"*** input.

The fan is controlled via the ***"xLevel1"*** output.

The safety chain of the fan must operate error-free for proper control of the fan.

The safety chain consists of the inputs:
- ***"xRepairSwitch"*** (repair switch (break contact)),
- ***"xMotorProtection"*** (motor protection switch (break contact))
- ***"xV_Belt"*** (runtime monitoring, V-belt monitoring)
- ***"xContactor"*** (contactor monitoring)

If there is a malfunction in the safety chain, the fan is switched off and the ***"xErrorFan"*** output is activated. A more detailed description of the malfunction is provided by the ***"wStatus"*** output.

Runtime monitoring is only activated after an adjustable startup time, which is required by the motor to achieve the rated speed. In order to avoid a false alarm in the case of pressure fluctuations in the duct while the system is running, an additional response delay can be defined.

The *"xContactor"* input monitors for correct functioning of the power contactor. For this purpose the *"xLevel1"* output is compared with the feedback signal of the contactor. If the switch status of the contactor differs from the *"xLevel1"* output for more than one second, there is a contactor malfunction.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

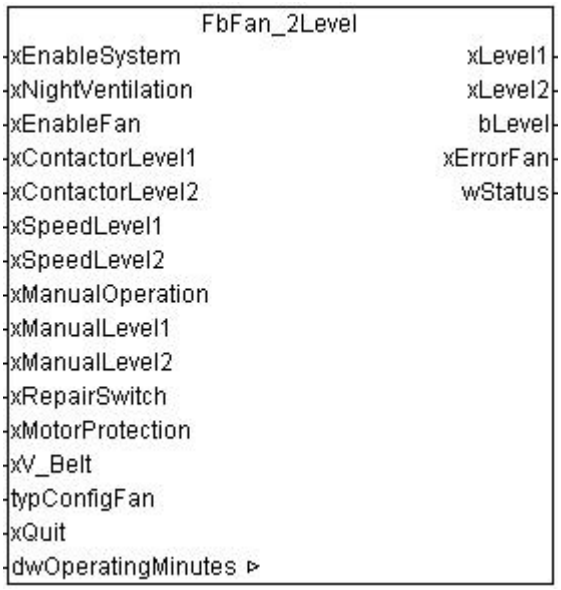
2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

Fan, 2-Level (FbFan_2Level)

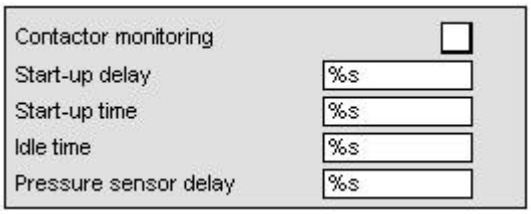
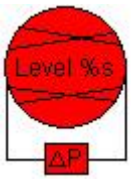
WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbFan_2Level	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnableSystem	BOOL	Fan control enabled by FbStartStop Default setting = TRUE
xNightVentilation	BOOL	Switch on fan during night ventilation
xEnableFan	BOOL	Switch on fan
xContactorLevel1	BOOL	Contactor monitoring via auxiliary contact level 1
xContactorLevel2	BOOL	Contactor monitoring via auxiliary contact level 2
xSpeedLevel1	BOOL	Speed level 1 in automatic mode
xSpeedLevel2	BOOL	Speed level 2 in automatic mode
xManualOperation	BOOL	Enable manual operation
xManualLevel1	BOOL	Fan level 1 in manual mode
xManualLevel2	BOOL	Fan level 2 in manual mode
xRepairSwitch	BOOL	Repair switch Default setting = TRUE
xMotorProtection	BOOL	Motor protection switch Default setting = TRUE
xV_Belt	BOOL	V-belt monitoring of the fan Default setting = TRUE
typConfigFan	←	Configuration parameters:
.tOnDelay	TIME	Delay time of the fan Default setting: t#0s
.tStartUpPeriod	TIME	Startup time of the fan Default setting: t#5s
.tPressureVariation	TIME	Error message delay time during pressure fluctuations Default setting: t#5s
.tSwitchOverTime	TIME	"Idle time" of the fan, when switching from level 2 to level 1. Default setting = t#2s
.xAuxiliaryContact	BOOL	Auxiliary contact for contactor monitoring present Default setting = FALSE
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating minutes of the fan

Return value:	Data type:	Comment:
xLevel1	BOOL	Level 1 of the fan
xLevel2	BOOL	Level 2 of the fan
bLevel	BYTE	Indication of the current fan level
xErrorFan	BOOL	Error message of the fan
wStatus	WORD	Display current status 0 = OK 16 = Repair switch 17 = Motor protection switch 19 = V-belt 26 = Error Contactor contact

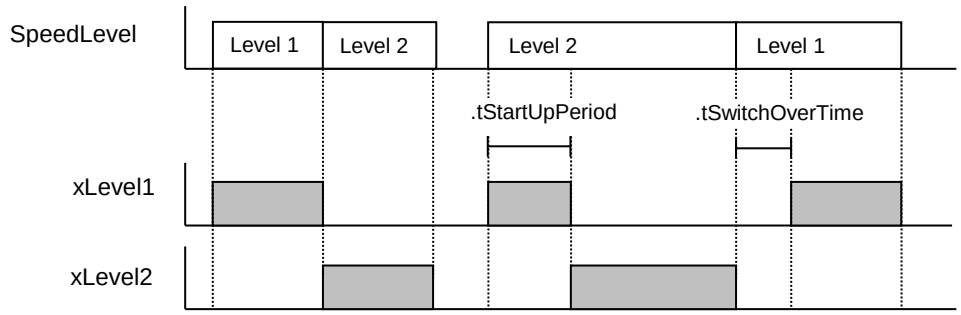
Graphical illustration:



Visualization objects:

ConfigFan_xLevel	
Fan_xLevel	

Time referenced behavior:



Function description:

The **FbFan_2Level** function block controls and monitors a 2-level fan.

Configuration parameters:

The configuration structure ***"typConfigFan"*** contains the following parameters:

- ***".tOnDelay"*** defines the On-delay for the fan.
- ***".tStartUpPeriod"*** defines the runup time for the fan. During this time period V-belt monitoring is not performed.
- ***".tSwitchOverTime"*** indicates the time the fan needs to switch over from speed level 2 to speed level 1. Both levels are deactivated in this state.
- ***".tPressureVariation"*** defines the response delay for V-belt monitoring.
- ***".xAuxiliaryContact"*** indicates whether a check-back signal from the power contactor auxiliary contact is present.

The fan is switched on the Automatic mode when the system has been enabled via ***"xEnableSystem"*** and the ***"xEnableFan"*** input has been activated.

In the automatic mode you can specify the desired fan level via the ***"xSpeedLevel1"*** and ***"xSpeedLevel2"*** inputs. If you select both fan levels, the fan remains in its last valid level.

If level 2 is specified immediately during fan startup, the fan starts with level 1 and changes to level 2 after the startup time has expired. At the same time, the runtime monitoring is activated. If level 2 is specified immediately during fan startup, the fan starts with level 1 and changes to level 2 after the startup time has expired. At the same time, the runtime monitoring is activated.

When night ventilation is enabled via the ***"xNightVentilation"*** input, the fan is controlled independently of *"xEnableSystem"* via the *"xEnableFan"* and *"xSpeedLevel1"* or *"xSpeedLevel2"* inputs.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the fan is switched via the ***"xManualLevel1"*** and ***"xManualLevel2"*** inputs.

The fan is controlled via the ***"xLevel1"*** and ***"xLevel2"*** outputs.

The safety chain of the fan must operate error-free for proper control of the fan.

The safety chain consists of the inputs:

- ***"xRepairSwitch"*** (repair switch (break contact)),

- ***"xMotorProtection"*** (motor protection switch (break contact))

- ***"xV_Belt"*** (runtime monitoring, V-belt monitoring)

The following inputs are added to the safety chain when the *".xAuxiliaryContact"* parameter is activated:

- ***"xContactorLevel1"*** (monitoring of power contactor, Level 1)

- ***"xContactorLevel2"*** (monitoring of power contactor, Level 2)

These inputs monitor for correct functioning of the power contactor. The switching outputs are compared with the check-back signal from the contator for this. If the switch status of the contactor differs from the respective output for more than one second, there is a contactor malfunction.

If there is a malfunction in the safety chain, the fan is switched off and the ***"xErrorFan"*** output is activated. A more detailed description of the malfunction is provided by the ***"wStatus"*** output.

Runtime monitoring is only activated after an adjustable startup time, which is required by the motor to achieve the rated speed. In order to avoid a false alarm in the case of pressure fluctuations in the duct while the system is running, an additional response delay can be defined.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

Note:

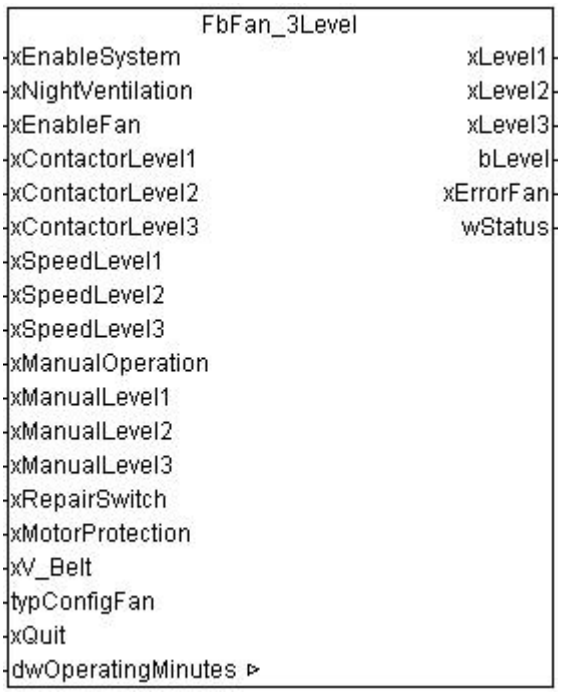
1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

Fan, 3-Level (FbFan_3Level)

WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbFan_3Level		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		

Graphical illustration:



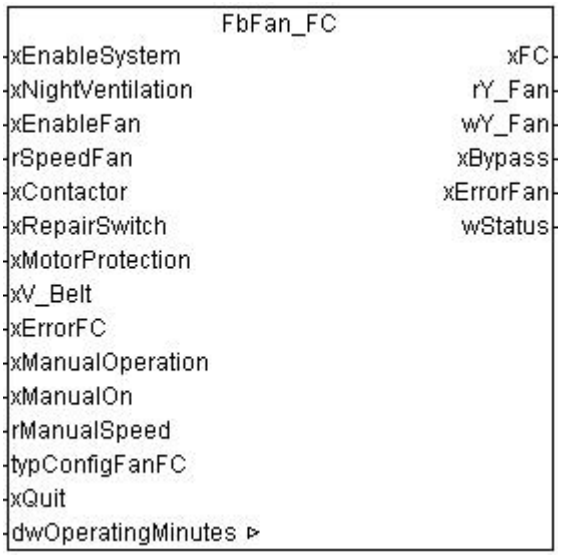
Function description:

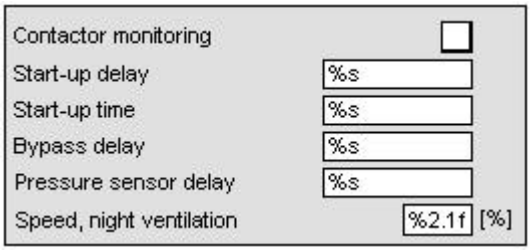
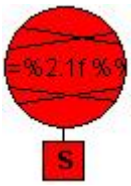
Refer to function description for FbFan_2Level.

Fan with Frequency Converter (FbFan_FC)

WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbFan_FC		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
xEnableSystem	BOOL	Fan control enabled by FbStartStop Default setting = TRUE	
xNightVentilation	BOOL	Enable Night ventilation Default setting = FALSE	
xEnableFan	BOOL	Switch on fan	
rSpeedFan	REAL	Fan speed in Automatic mode [%] Value range = 0 – 100 Default setting = 50	
xContactor	BOOL	Contactor monitoring via auxiliary contact	
xRepairSwitch	BOOL	Repair switch Default setting = TRUE	
xMotorProtection	BOOL	Motor protection switch Default setting = TRUE	
xV_Belt	BOOL	V-belt monitoring of the fan Default setting = TRUE	
xErrorFC	BOOL	Fault check-back from frequency converter	
.xManualOperation	BOOL	Enable manual operation	
.xManualOn	BOOL	Switch on frequency converter manually	
rManualSpeed	REAL	Fan speed in Manual mode [%] Value range = 0 – 100 Default setting = 50	
typConfigFanFC	←	Configuration parameters:	
.tOnDelay	TIME	Delay time of the fan Default setting: t#0s	
.tStartUpPeriod	TIME	Startup time of the fan Default setting = t#5s	
.tPressureVariation	TIME	Error message delay time during pressure fluctuations Default setting: t#5s	
.tBypassDelay	TIME	Bypass protection delay time on frequency converter error Default setting: t#5s	
.rSpeedFanNight Ventilation	REAL	Speed during night ventilation [%] Value range: 0 −100 Default setting = 25	

.xAuxiliaryContact	BOOL	Auxiliary contact for contactor monitoring present Default setting = FALSE
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating minutes of the fan
Return value	**Data type:**	**Comment:**
xFC	BOOL	Switch on frequency converter
rY_Fan	REAL	Set value for frequency converter [%] Value range: 0 −100
wY_Fan	WORD	Set value for frequency converter Value range = 0 – 32767
xBypass	BOOL	Bypass protection switching signal
xErrorFan	BOOL	Fan fault
wStatus	WORD	Display current status 0 = OK 16 = Repair switch 17 = Motor protection switch 19 = V-belt 26 = Error Contactor contact 38 = Error FC
Graphical illustration:		



Visualization objects:	
ConfigFan_FC	
Fan_FC	

Function description:

The **FbFan_FC** function block controls and monitors a fan with actuation using frequency converters.

<u>Configuration parameters:</u>

The configuration structure ***"typConfigFanFC"*** contains the following parameters:

- ***".tOnDelay"*** defines the On-delay for the fan.
- ***".tStartUpPeriod"*** defines the runup time for the fan. During this time period V-belt monitoring is not performed.
- ***".tPressureVariation"*** defines the response delay for V-belt monitoring.
- ***".tBypassDelay"*** defines the delay period for bypass switchover.
- ***".rSpeedFanNightVentilation"*** defines the fan speed during night ventilation.
- ***".xAuxiliaryContact"*** indicates whether a check-back signal from the power contactor auxiliary contact is present.

The fan is switched on the Automatic mode when the system has been enabled via ***"xEnableSystem"*** and the ***"xEnableFan"*** input has been activated.

The frequency converter (FC) is controlled via the ***"xFC"*** output.

In the Automatic mode, the required speed from the ***"rSpeedFan"*** input is output directly at the ***"rY_Fan"*** output.

The output value ***"wY_Fan"*** has the same meaning as the *"rY_Fan*" output, the output just has standardized values between 0 – 32767.

When night ventilation is enabled via the "xNightVentilation" input, the fan is switched on independently of "xEnableSystem" via the "xEnableFan" input. In this case, the set value *".rSpeedFanNightVentilation"* is output at the *"rY_Fan"* output.

The safety chain of the fan must operate error-free for proper control of the fan.

The safety chain consists of the inputs:

- ***"xRepairSwitch"*** (repair switch (break contact)),

- ***"xMotorProtection"*** (motor protection switch (break contact))

- ***"xV_Belt"*** (runtime monitoring, V-belt monitoring)

The following input is added to the safety chain when the *".xAuxiliaryContact"* parameter is activated:

- ***"xContactor"*** (contactor monitoring)

This input monitors for correct functioning of the power contactor. The switch output is compared with the check-back signal from the contactor for this. If the switch status of the contactor differs from the respective output for more than one second, there is a contactor malfunction.

If there is a malfunction in the safety chain, the fan is switched off and the ***"xErrorFan"*** output is activated. A more detailed description of the malfunction is provided by the ***"wStatus"*** output.

Runtime monitoring is only activated after an adjustable startup time, which is required by the motor to achieve the rated speed. In order to avoid a false alarm in the case of pressure fluctuations in the duct while the system is running, an additional response delay can be defined.

A bypass contactor can be used in the event of a frequency converter malfunction.

If the frequency converter reports a malfunction via the ***"xErrorFC"*** input, the frequency converter is disconnected from the fan via a contactor. When contactor monitoring reports the open (disconnected) status, the bypass contactor is activated with a time delay via the ***"xBypass"*** output.

When the frequency converter malfunction is rectified, the bypass contactor is first opened and the contactor for the frequency converter re-activated with a time delay.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the fan is switched on via the ***"xManualOn"*** input and controlled via the ***"rManualSpeed"*** input.

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

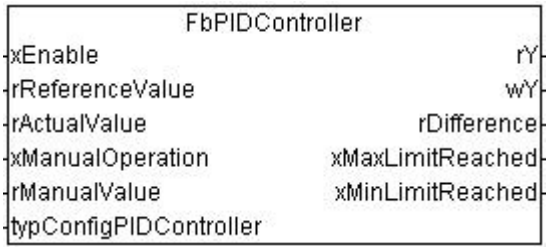
2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

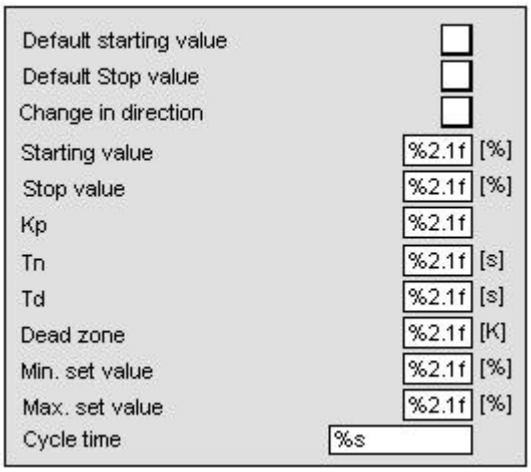
05 Controllers

PID Controller (FbPIDController)

WAGO-I/O-*PRO* Library Elements		
Category:	Building Automation	
Name:	FbPIDController	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable PID controller Default setting = TRUE
rReferenceValue	REAL	Reference value [°C]
rActualValue	REAL	Actual value [°C]
xManualOperation	BOOL	Enable manual operation
rManualValue	REAL	Set value in manual mode [°C]
typConfigPIDController	←	Configuration parameters:
.xChangeInDirection	BOOL	Operating direction of the controller FALSE = heating; TRUE = cooling
.xPresetON	BOOL	Release start value on activation of the controller
.xPresetOFF	BOOL	Release stop value on deactivation of the controller
.rPresetValueON	REAL	Setting value of the controller when switched on
.rPresetValueOFF	REAL	Setting value of the controller when switched off
.tCycleTime	TIME	Cycle time for the controller Default = t#100ms
.rOutputMin	REAL	Minimum value of the set value (rY)
.rOutputMax	REAL	Maximum value of the set value (rY) Default setting = 100
.rKp	REAL	Proportional gain (P portion) Default setting = 10
.rTn	REAL	Reset time (I part) [s] Default setting = 60 s
.rTd	REAL	Rate time (D portion) [s]
.rDeadZone	REAL	Dead zone +/- [K] Default setting = 0
Return value:	**Data type:**	**Comment:**
rY	REAL	Set value of controller [%]
wY	WORD	Set value of the controller Value range = 0 – 32767

rDifference	REAL	Deviation of the set value from the actual value
xMaxLimitReached	BOOL	Maximum set value reached
xMinLimitReached	BOOL	Minimum set value reached
Graphical illustration:		



Visualization objects:	
ConfigPIDController	

Function description:

The **FbPIDController** function block is a standard PID controller with freely configurable Start and Stop values. Additionally, the function block offers the possibility to change the operating direction of the controller.

Configuration parameters:

The configuration structure ***"typConfigPIDController"*** contains the following parameters:

- ***".rKp"*** defines the proportional gain for the controller.
- ***".rTn"*** defines the reset time of the controller.
- ***".rTd"*** defines the derivative time of the controller.
- ***".rDeadZone"*** defines the range around the reference value in which the set value may not be changed (dead zone).
- ***".rOutputMin"*** defines the minimum setting value for the controller.
- ***".rOutputMax"*** defines the maximum setting value for the controller.
- ***".tCycleTime"*** defines the cycle time for the controller.
- ***".xChangeInDirection"*** allows the operating direction of the controller to be changed.
- ***".xPresetOn"*** ensures that the controller starts with the set value ***".rPresetValueOn"***.
- ***".xPresetOff"*** ensures that the controller outputs the set value ***".rPresetValueOff"*** when it is switched off. If *".xPresetOff"* is not activated, the controller outputs the set value of zero when it is switched off.

If the ***"xEnable"*** input is activated, the input values ***"rActualValue"*** and ***"rReferenceValue"*** are used to calculate the set value ***"rY"***.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the reference value from the ***"rManualValue"*** input is output at the *"rY"* output.

The output value ***"wY"*** has the same meaning as the *"rY"* output, except that the output has standardized values between 0 – 32767.

The output value ***"wY"*** has the same meaning as the output *"rY"* and depends on *".rOutputMin"* and *".rOutputMax"*. The values of *"wY"* are scaled from 0-32767 instead of 0-100 as in *"rY"*. *„wY“* is usable as long as ".rOutputMax" <= 100.

When the controller reaches its maximum set value ("***xMaxLimitReached"*** = TRUE) or its minimum set value (***"xMinLimitReached"*** = TRUE), the I portion of the controller is inhibited to prevent the set value from being integrated further (anti-wind-up).

The ***"rDifference"*** output indicates the difference between the specified and actual values.

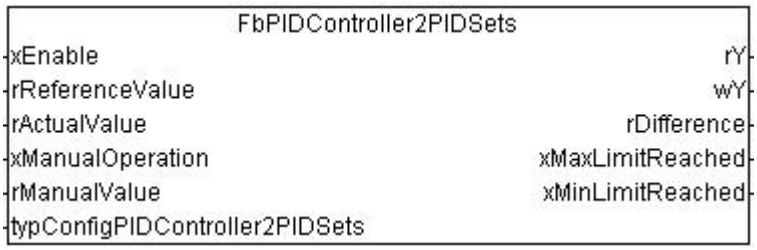
PID Controller wit Two Sets of Control Parameters (FbPIDController2PIDSets)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbPIDController2PIDSets	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable PID controller Default setting = TRUE
rReferenceValue	REAL	Reference value [°C]
rActualValue	REAL	Actual value [°C]
xManualOperation	BOOL	Enable manual operation
rManualValue	REAL	Set value in manual mode [°C]
typConfigPIDController 2PIDSets	←	Configuration parameters:
.xChangeInDirection	BOOL	Operating direction of the controller FALSE = heating; TRUE = cooling
.xPresetON	BOOL	Release start value on activation of the controller
.xPresetOFF	BOOL	Release stop value on deactivation of the controller
.rPresetValueON	REAL	Setting value of the controller when switched on
.rPresetValueOFF	REAL	Setting value of the controller when switched off
.tCycleTime	TIME	Cycle time for the controller Default = t#100ms
.rOutputMin	REAL	Minimum value of the set value (rY)
.rOutputMax	REAL	Maximum value of the set value (rY) Default setting = 100
.rKp1	REAL	Proportional gain (P portion) for the first set of controller parameters Default setting = 2.5
.rTn1	REAL	Reset time of the controller for the first set of controller parameters [s] Default setting = 80 s
.rTd1	REAL	Derivative time (D portion) for the first set of controller parameters [s] Default setting = 0 s
.rKp2	REAL	Proportional gain (P portion) for the second set of controller parameters Default setting = 2.5
.rTn2	REAL	Reset time of the controller for the second set of controller parameters [s] Default setting = 300 s

.rTd2	REAL	Derivative time (D portion) for the second set of controller parameters [s] Default setting = 0 s
.rDeviation	REAL	Maximum deviation between specified/actual values for the second set of controller parameters [%] Default setting = 5
.rDeadZone	REAL	Dead zone +/- [K] Default setting = 0

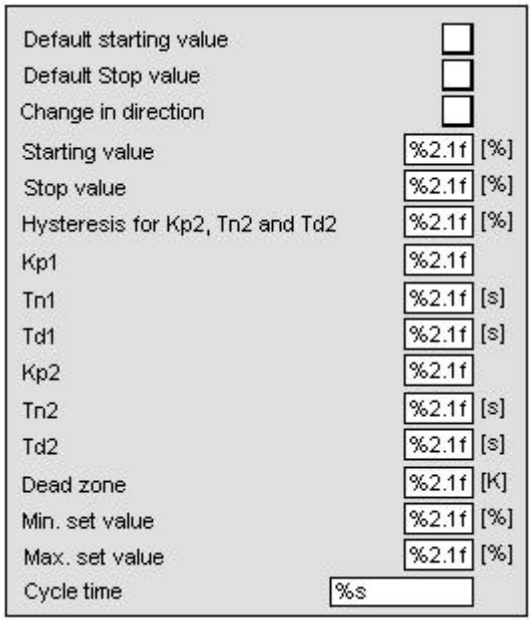
Return value:	**Data type:**	**Comment:**
rY	REAL	Set value of controller [%]
wY	WORD	Set value of the controller Value range = 0 – 32767
rDifference	REAL	Deviation of the set value from the actual value
xMaxLimitReached	BOOL	Maximum set value reached
xMinLimitReached	BOOL	Minimum set value reached

Graphical illustration:



Visualization objects:

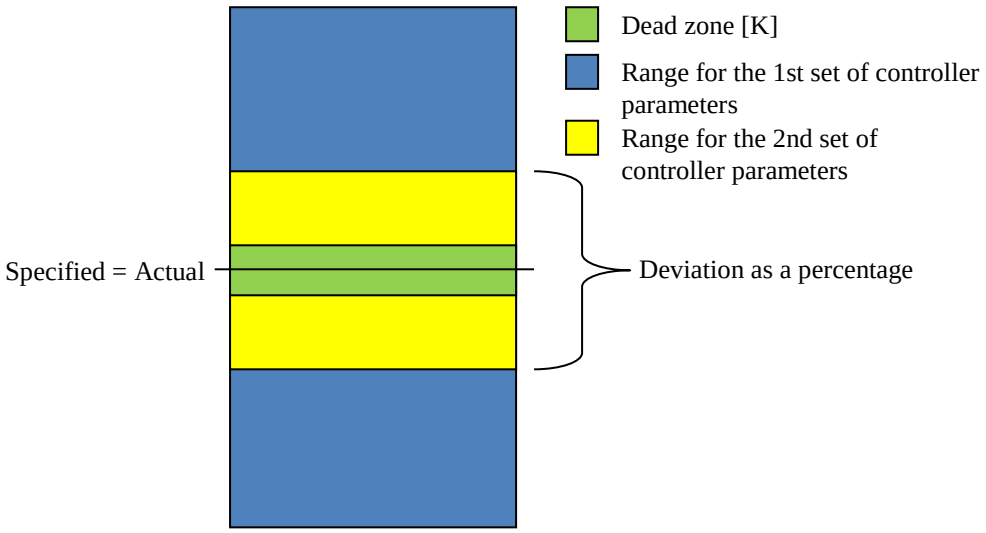
ConfigPIDController2PID Sets



ConfigPIDController2PID Sets	Hysteresis for Kp2, Tn2 and Td2 %2.1f [%] Kp1 %2.1f Tn1 %2.1f [s] Td1 %2.1f [s] Kp2 %2.1f Tn2 %2.1f [s] Td2 %2.1f [s] Dead zone %2.1f [K]
ConfigPIDPressure Controller	Reference value [mBar] %2.0f Hysteresis for Kp2, Tn2 and Td2 %2.1f [%] Kp1 %2.1f Tn1 %2.1f [s] Td1 %2.1f [s] Kp2 %2.1f Tn2 %2.1f [s] Td2 %2.1f [s] Dead zone %2.1f [K]

Diagram:

Deviation, specified/actual values:



Function description:

Besides the options offered with the standard PID controller, the **FbPIDController2PIDSets** also provides the function for switching back and forth between two sets of controller parameters.

Configuration parameters:

The configuration structure ***"typConfigPIDController2PIDSets"*** contains the following parameters:

- ***".rKp1"*** and "***.rKp2"*** define the proportional gain for the controller.
- ***".rTn1"*** and ***".rTn2"*** define the reset time of the controller.
- ***".rTd1"*** and ***".rTd2"*** define the derivative time of the controller.
- ***".rDeviation"*** defines the maximum deviation between the specified/actual values for the second set of parameters.
- ***".rDeadZone"*** defines the range around the reference value in which the set value may not be changed (dead zone).
- ***".rOutputMin"*** defines the minimum setting value for the controller.
- ***".rOutputMax"*** defines the maximum setting value for the controller.
- ***".tCycleTime"*** defines the cycle time for the controller.
- ***".xChangeInDirection"*** allows the operating direction of the controller to be changed.
- ***".xPresetOn"*** ensures that the controller starts with the set value ***".rPresetValueOn"***.
- ***".xPresetOff"*** ensures that the controller outputs the set value ***".rPresetValueOff"*** when it is switched off. If *".xPresetOff"* is not activated, the controller outputs the set value of zero when it is switched off.

If the ***"xEnable"*** input is activated, the input values ***"rActualValue"*** and ***"rReferenceValue"*** are used to calculate the set value ***"rY"***.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the reference value from the ***"rManualValue"*** input is output at the *"rY"* output.

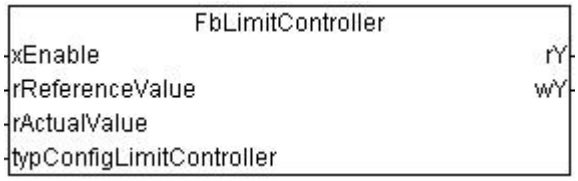
The output value ***"wY"*** has the same meaning as the *"rY"* output, except that the output has standardized values between 0 – 32767.

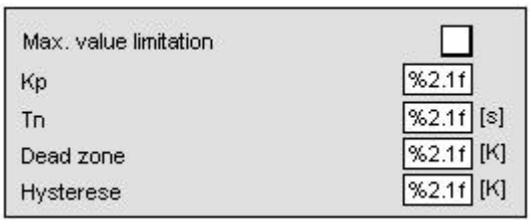
When the controller reaches its maximum set value ("***xMaxLimitReached"*** = TRUE) or its minimum set value (***"xMinLimitReached"*** = TRUE), the I portion of the controller is inhibited to prevent the set value from being integrated further (anti-wind-up).

The ***"rDifference"*** output indicates the difference between the specified and actual values.

PI Limit Controller (FbLimitController)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbLimitController	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable PID controller Default setting = TRUE
rReferenceValue	REAL	Reference value [°C]
rActualValue	REAL	Actual value [°C]
typConfigLimitController	←	Configuration parameters:
.xLimitMaxValue	BOOL	TRUE = The controller limits at maximum value (return limit) FALSE = The controller limits a minimum value (antifreeze)
.rHysteresis	REAL	Hysteresis for deactivation of the controller
.rKp	REAL	Proportional gain (P portion) Default setting = 10
.rTn	REAL	Reset time (I part) [s] Default setting = 60 s
.rDeadZone	REAL	Dead zone +/- [K] Default setting = 0
Return value:	**Data type:**	**Comment:**
rY	REAL	Set value of controller [%]
wY	WORD	Set value of the controller Value range = 0 – 32767
Graphical illustration:		



Visualization objects:	
ConfigLimitController	

ConfigAntifreeze Controller	Reference value %2.1f [°C] Kp %2.1f Tn %2.1f [s] Dead zone %2.1f [K] Hysteresis %2.1f [K]

Function description:

The **FbLimitController** function block serves as a limit controller to prevent a reference lower value (e.g., antifreeze controller) or a reference upper value (e.g., return temperature temperature limit controller) from being violated.

Configuration parameters:

The configuration structure ***"typConfigLimitController"*** contains the following parameters:

- ***".rKp"*** defines the proportional gain for the controller.
- ***".rTn"*** defines the reset time of the controller.
- ***".rDeadZone"*** defines the range around the reference value in which the set value may not be changed (dead zone).
- ***".xLimitMaxValue"*** prevents the reference value from being exceeded. When *".xLimitMaxValue"* is deactivated, the limit controller ensures that no values fall below the reference value.
- ***".rHysteresis"*** defines the hysteresis for deactivating the controller. When, for example, the limit controller ensures that the reference value is not exceeded, the controller is deactivated when *"rActualValue"* is less than *"rReferenceValue" – "rHysteresis".*

If the ***"xEnable"*** input is activated, the input values ***"rActualValue"*** and ***"rReferenceValue"*** are used to calculate the set value ***"rY"***.

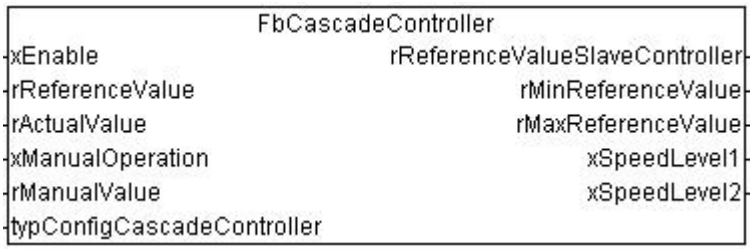
The output value ***"wY"*** has the same meaning as the *"rY"* output, except that the output has standardized values between 0 – 32767.

Cascade Controller (FbCascadeController)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbCascadeController	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable Cascade controller Default setting = TRUE
rReferenceValue	REAL	Reference value [°C] Default setting = 22 °C
rActualValue	REAL	Actual value [°C]
xManualOperation	BOOL	Enable manual operation
rManualValue	REAL	Set value in manual mode [°C] Default setting = 22 °C
typConfigCascade Controller	←	Configuration parameters:
.rKp	REAL	Proportional gain (P portion) Default setting = 2.5
.rTn	REAL	Reset time of the controller [s] Default setting = 300 s
.rDeadZone	REAL	Dead zone +/- [K] Default setting = 0
.rOffsetMaxReference Value	REAL	Offset for the maximum reference value for the slave controller as a function of the reference value [K] Default setting = 6 K
.rOffsetMinReference Value	REAL	Offset for the minimum reference value for the slave controller as a function of the reference value [K] Default setting = 4 K
.tDelaySwitchOver	TIME	Delay time for switching between the two fan levels Default setting = t#5 m
.xControlSpeedLevel	BOOL	Enable for determining the required fan level Default setting = FALSE
Return value:	**Data type:**	**Comment:**
rReferenceValueSlave Controller	REAL	Reference value for slave controller [°C]
rMinReferenceValue	REAL	Minimum reference value for slave controller
rMaxReferenceValue	REAL	Maximum reference value for slave controller

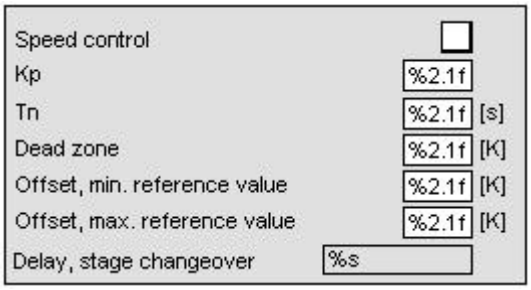
xSpeedLevel1	BOOL	Request fan level 1
xSpeedLevel2	BOOL	Request fan level 2

Graphical illustration:

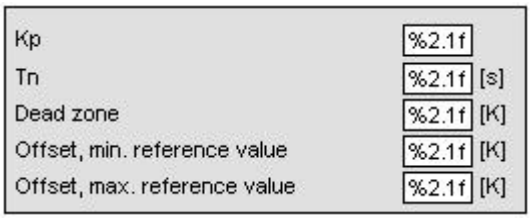


Visualization objects:

ConfigCascadeController

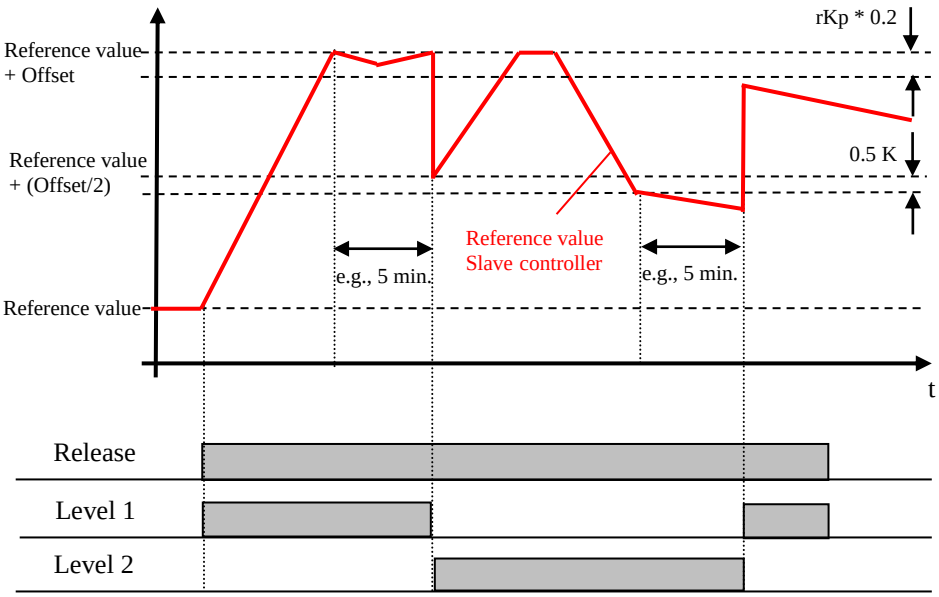


ConfigCascadeController 1



Time referenced behavior:

Determination of the required fan level during heating:



Function description:

The cascade controller (master controller) **FbCascadeController** function block determines the reference value for the slave controller. The function block also evaluates the required fan level as an option.

Configuration parameters:

The configuration structure ***"typConfigCascadeController"*** contains the following parameters:

- ***".rKp"*** defines the proportional gain for the controller.
- ***".rTn"*** defines the reset time of the controller.
- ***".rDeadZone"*** defines the range around the reference value in which the set value may not be changed (dead zone).
- ***".rOffsetMaxReferenceValue"*** defines the maximum setting value for the controller as a function of the reference value.
- ***".rOffsetMinReferenceValue"*** defines the minimum setting value for the controller as a function of the reference value.
- ***".tDelaySwitchOver"*** defines the delay time for switching between fan levels.
- ***".xControlSpeedLevel"*** enables the function for determining the required fan level.

If the controller is enabled via the ***"xEnable"*** input, the reference value for the slave controller ***"rReferenceValueSlaveController"*** is calculated from the input values ***"rActualValue"*** and ***"rReferenceValue"***.

The outputs ***"rMinReferenceValue"*** and ***"rMaxReferenceValue"*** indicate the minimum and maximum reference value for the slave controller. This range applies both for the Automatic mode and for manual override.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the reference value from the ***"rManualValue"*** input is output at the *"rReferenceValueSlaveController"* output.

If the function for determining the required fan level is enabled, the required fan level is specified at the ***"xSpeedLevel1"*** and ***"xSpeedLevel2"*** outputs.

At first, the fans run at level 1 as long until the set value for the slave controller has reached its maximum value when heating or its minimum value when cooling. After a defined delay time, the fan is switched to level 2. To prevent the actual value from rising due to the double volume flow rate, the reference value for the slave controller is reduced (heating) or increased (cooling) at the particular switching point.

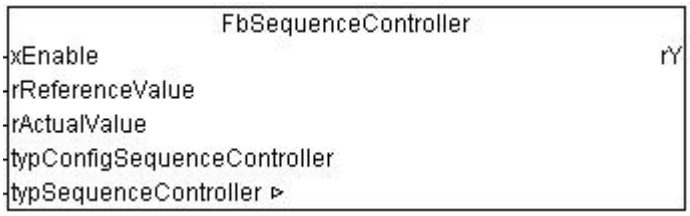
If the reference value for the slave controller falls below its limit (for heating), or rises above its limit (for cooling) again (Offset/2) plus the hysteresis (0.5 K), the fans are switched back to level 1 with a time delay. When switching back to level 1, the reference value for the slave controller is raised or lowered again as required.

Sequence Controller (FbSequenceController)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbSequenceController	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable sequence controller Default setting = TRUE
rReferenceValue	REAL	Reference value [°C] Default setting = 22 °C
rActualValue	REAL	Actual value [°C]
typConfigSequence Controller	←	Configuration parameters:
.rDeadZone	REAL	Dead zone +/- [K] Default setting = 0
.rKp1	REAL	Proportional gain (P portion) for the first set of controller parameters Default setting = 2.5
.rTn1	REAL	Reset time of the controller for the first set of controller parameters [s] Default setting = 300 s
.rTd1	REAL	Derivative time (D portion) for the first set of controller parameters [s] Default setting = 0 s
.rKp2	REAL	Proportional gain (P portion) for the second set of controller parameters Default setting = 2.5
.rTn2	REAL	Reset time of the controller for the second set of controller parameters [s] Default setting = 300 s
.rTd2	REAL	Derivative time (D portion) for the second set of controller parameters [s] Default setting = 0 s
.rDeviation	REAL	Maximum deviation between specified/actual values for the second set of controller parameters [%] Default setting = 0 (no switchover)
.bMaxSequenceNumber	BYTE	Max. number of sequences used Value range = 2 – 4 Default setting = 2
typSequenceController	←	Data structure for data exchange between the individual sequences and the sequence controller

Return value:	Data type:	Comment:
rY	REAL	Set value for sequences [%]

Graphical illustration:



Visualization objects:

ConfigSequence Controller

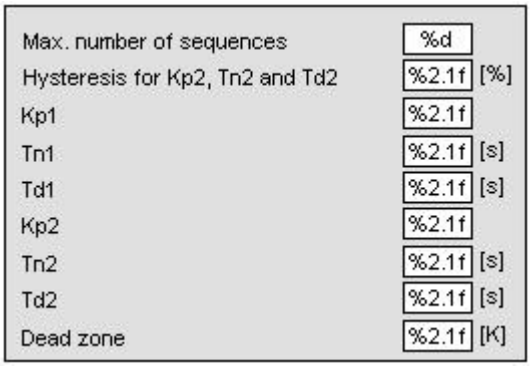
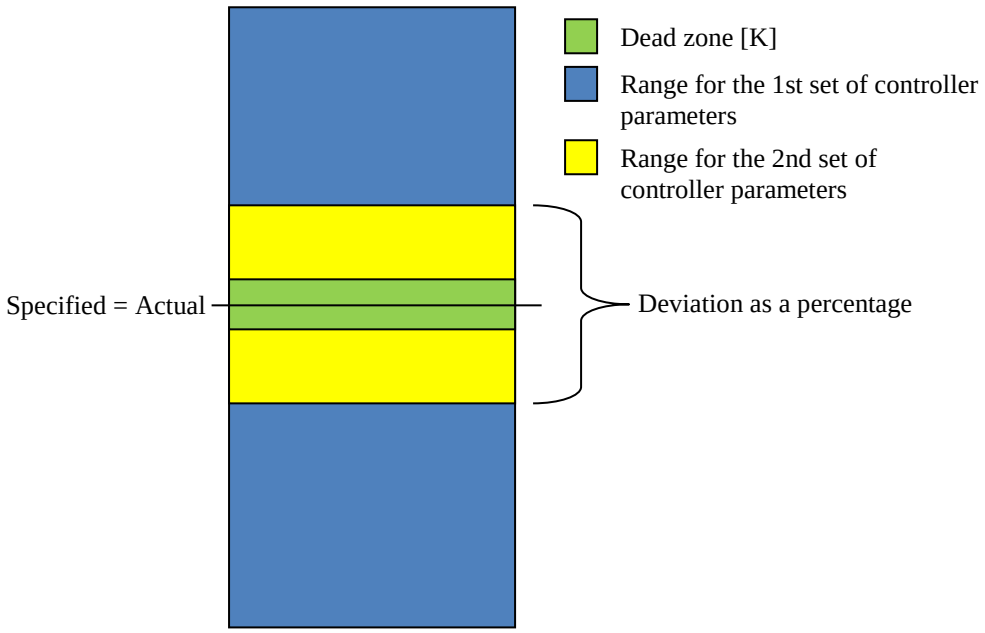


Diagram:

Deviation (specified/actual):



Function description:

The **FbSequenceController** sequence controller can support up to four (4) sequences. If a malfunction occurs in one of these sequences, the sequence concerned is automatically disabled for the control system.

Configuration parameters:

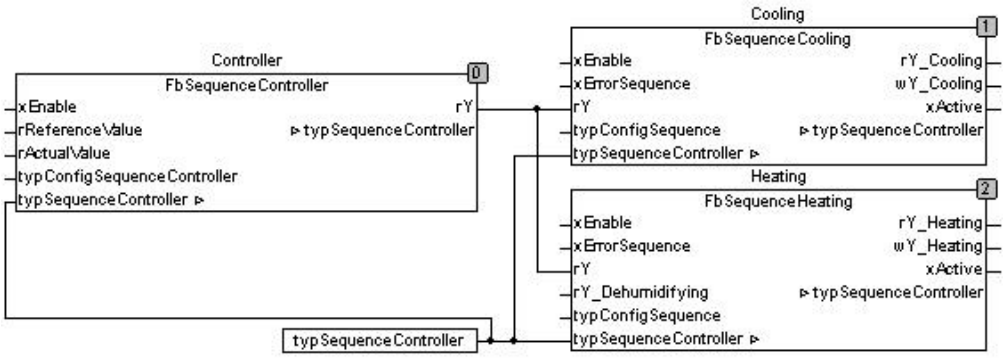
The configuration structure ***"typConfigSequenceController"*** contains the following parameters:

- ***".rDeadZone"*** defines the range around the reference value in which the set value may not be changed (dead zone).
- ***".rKp1"*** and "***.rKp2"*** define the proportional gain for the controller.
- ***".rTn1"*** and ***".rTn2"*** define the reset time of the controller.
- ***".rTd1"*** and ***".rTd2"*** define the derivative time of the controller.
- ***".rDeviation"*** defines the maximum deviation between the specified/actual values for the second set of parameters.
- ***".bMaxSequenceNumber"*** defines the number of sequences used.

If the controller is enabled via the ***"xEnable"*** input, the set value for ***"rY"*** for the sequences is calculated from the input values ***"rActualValue"*** and ***"rReferenceValue"***.

Note:

1.) The structure ***"typSequenceController"*** is required for check-back by the individual sequences. Therefore, this structure must be linked to all of the sequences, which are to be controlled via this module.

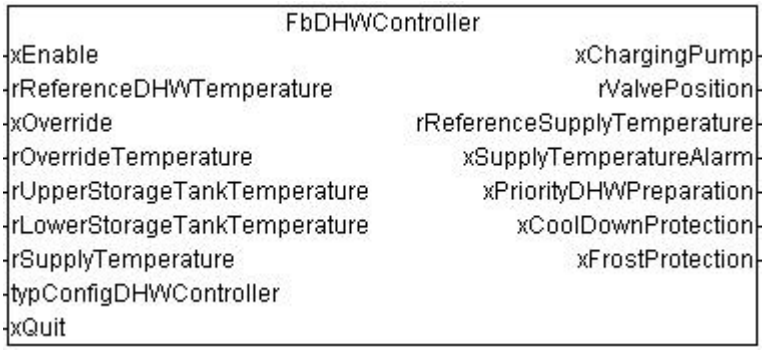


Two-Point Controller for DHW Preparation (FbDHWController)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbDHWController	
Type:	Function ☐	Function block ☒ Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable domestic hot water preparation Default setting = TRUE
rReferenceDHW Temperature	REAL	Reference value for domestic hot water (DHW) [°C]
xOverride	BOOL	Override reference value (e.g., overheating protection)
rOverrideTemperature	REAL	Overriden specified temperature [°C]
rUpperStorageTank Temperature	REAL	Upper storage tank temperature sensor [°C]
rLowerStorageTank Temperature	REAL	Lower storage tank temperature sensor [°C]
rSupplyTemperature	REAL	Actual value of supply temperature for hot water conditioning [°C]
typConfigDHWController	←	Configuration parameters:
.rHysteresis	REAL	Hysteresis for 2-point controller [K] Default setting = 5
.rMinTemperature FrostProtection	REAL	Min. supply or storage temperature for antifreeze protection [°C] Default setting = 5
.rTempDropPriorityDHW Preparation	REAL	Temperature drop for DHW priority Default setting = 20
.rTempDiffPrimary SecondarySystem	REAL	Temperature difference between supply temperature and reference value value for DHW charging [K] Default setting = 10
.tMaxStartUpSupply Temperature	TIME	Maximum time until set supply temperature is reached Default setting = t#30 m
.xThreeWayValve	BOOL	3-way valve present Default setting = TRUE
.xAutoQuit	BOOL	Automatic acknowledgement of supply temperature alarm Default setting = FALSE
.xEnableCoolDown Protection	BOOL	Enable cool-down protection Default setting = FALSE
.xSupplyTemperature Sensor	BOOL	Supply temperature sensor present Default setting = TRUE
xQuit	BOOL	Acknowledge the supply temperature alarm

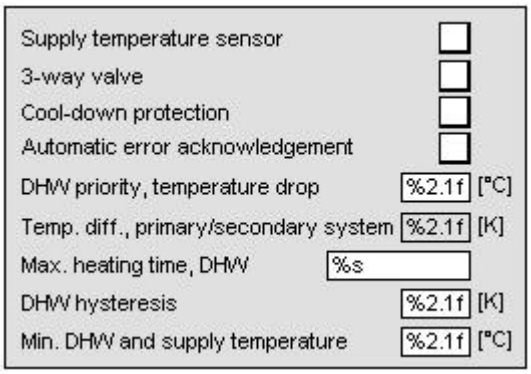
Return value:	Data type:	Comment:
xChargingPump	BOOL	Switching signal for charging pump
rValvePosition	REAL	Valve position [%]
rReferenceSupply Temperature	REAL	Supply temperature reference value for hot water conditioning [°C]
xSupplyTemperatureAlarm	BOOL	Supply temperature alarm for hot water conditioning
xPriorityDHWPreparation	BOOL	Hot water priority request
xCoolDownProtection	BOOL	Cool-down protection
xFrostProtection	BOOL	Antifreeze function

Graphical illustration:

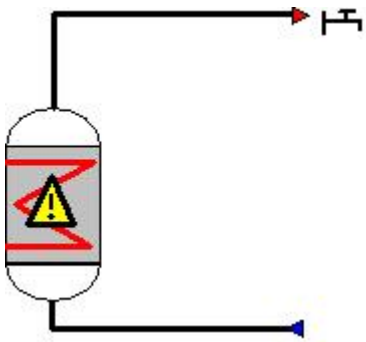


Visualization objects:

ConfigDHW Controller



DHWController



Function description:

The **FbDHWController** 2-point controller regulates the domestic hot water (DHW) temperature for the storage tank using an upper and lower storage tank temperature sensor.
When a supply temperature sensor is provided, the hot water storage tank is additionally protected against forced cooling.

Configuration parameters:

The configuration structure ***"typConfigDHWController"*** contains the following parameters:

- ***".rHysteresis"*** defines the hysteresis for the 2-point controller.
- ***".rMinTemperatureFrostProtection"*** defines the minimum supply temperature and the minimum upper storage tank temperature for forced charging.
- ***".rTempDropPriorityDHWPreparation"*** defines the minimum temperature drop for the domestic hot water priority function.
- ***".rTempDiffPrimarySecondarySystem"*** defines the offset to the DHW reference value for the specified supply temperature.
- ***".tMaxStartUpSupplyTemperature"*** defines the maximum time that may elapse until the required supply temperature is reached.
- ***".xThreeWayValve"*** defines whether a 3-way valve is used.
- ***".xAutoQuit"*** acknowledges the supply temperature alarm automatically when the required supply temperature is reached.
- ***".xEnableCoolDownProtection"*** enables the cooling protection function. The cooling protection function protects the hot water storage tank against forced cooling and must always have a supply temperature sensor for this.
- ***".xSupplyTemperatureSensor"*** defines whether a supply temperature sensor is provided.

Domestic hot water preparation (DHW preparation) is enabled via the input ***"xEnable"***.

The hot water storage tank is charged when the upper storage tank temperature ***"rUpperStorageTankTemperature"*** , minus the hysteresis, is situated below the ***"rReferenceDHWTemperature"***. For charging, the valve ***"rValvePosition"*** is opened and the pump ***"xChargingPump"*** enabled.

The output ***„xPriorityDHWPreparation“*** for domestic hot water priority will be activated, when following requirements are fulfilled:

1.) The upper storage temperature "rUpperStorageTankTemperature" is below the limit for the domestic hot water priority function.

2.) The supply temperature "rSupplyTemperature" falls within the upper storage temperature "rUpperStorageTankTemperature" plus the adjusted offset to the DHW reference value.

The valve is closed and pump enable canceled when the upper storage tank temperature *"rUpperStorageTankTemperature"* and the lower storage tank temperature ***"rLowerStorageTankTemperature"*** is greater than the *"rReferenceDHWTemperature"* .

Forced charging of the storage tank takes place when the storage tank temperature *"rUpperStorageTankTemperature"* or *"rLowerStorageTankTemperature" falls below the defined limit. This also happens, when the storage tank hasn't reached the reference temperature and* the supply temperature ***"rSupplyTemperature"*** falls below the defined limit. Forced charging of the storage tank is indicated via the ***"xFrostProtection"*** output.

The specified supply temperature ***"rReferenceSupplyTemperature"*** is calculated using an offset to the DHW reference value *"rReferenceDHWTemperature"* and ensures sufficient heat transfer.

For cooling protection, domestic hot water preparation is not enabled until the supply temperature *"rSupplyTemperature"* is greater than the upper storage tank temperature *"rUpperStorageTankTemperature"*.

If the supply temperature *"rSupplyTemperature"* does not achieve the required temperature within the defined time period, an alarm is issued via the output ***"xSupplyTemperatureAlarm"***.

When automatic acknowledgement is activated, the malfunction is canceled automatically when the specified supply temperature is reached. The alarm can also be reset via the ***"xQuit"*** input.

If there is a risk of overheating of heating units, the 2-point controller can be enabled via the *"xOverride"* input, independently of the ***"xEnable"*** input. In this case, the ***"rOverrideTemperature"*** is used as the specified storage tank temperature.

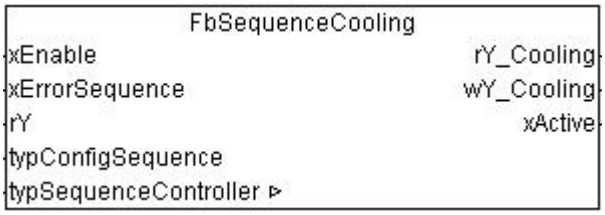
Note:

- Supply to the sensor can be blocked off when the valve is closed and the pump is shut down, depending on where the supply temperature sensor is installed. In this case the cool-down protection function must be deactivated.
- If a 2-way valve is used in place of a 3-way valve, the charging pump will not be switched on as long as the full-way valve is closed.
- When a 3-way valve is installed, a shorter overtravel time should be selected for the charging pump, as the hot water is routed directly into the return line and this could, under some circumstances, result in the return temperature being increased excessively.
- If only the upper storage tank temperature sensor is present the measured value must be linked both to the input for the upper and for the lower storage tank temperature sensor.

06 Sequences

Cooling Sequence (FbSequenceCooling)

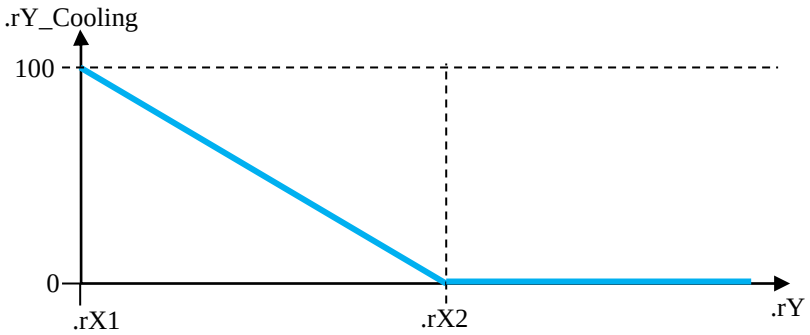
WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbSequenceCooling	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable sequence Default setting = TRUE
xErrorSequence	BOOL	Error check-back signal from cooling register
rY	REAL	Set value for sequence controller [%]
typConfigSequence	←	Configuration parameters:
.rX1	REAL	Min. setting value for cooling sequence [%]
.rX2	REAL	Max. setting value for cooling sequence [%]
.bSequenceNumber	BYTE	Sequence number (from cooling toward heating) Value range = 1 – 4
typSequenceController	←	Data structure for data exchange between the individual sequences and the sequence controller
Return value:	**Data type:**	**Comment:**
rY_Cooling	REAL	Valve position cooling register
wY_Cooling	WORD	Valve position cooling register Value range = 0 – 32767
xActive	BOOL	Active indicator for sequence
Graphical illustration:		



Visualization objects:

ConfigSequence

Diagram:



Function description:

The **FbSequenceCooling** function block converts the set value for the sequence controller into a setting value for the cooling elements.

Configuration parameters:

The configuration structure ***"typConfigSequence"*** contains the following parameters:

- ***".rX1"*** defines the minimum set value for the sequence controller for the cooling sequence.
- ***".rX2"*** defines the maximum set value for the sequence controller for the cooling sequence.
- ***".bSequenceNumber"*** defines the number of the sequence. Sequence numbers are assigned in the HVAC system in order from cooling toward heating.

The sequence is enabled via the ***"xEnable"*** input. The output set value ***"rY_Cooling"*** is then calculated from the set value from the sequence controller ***"rY"***.

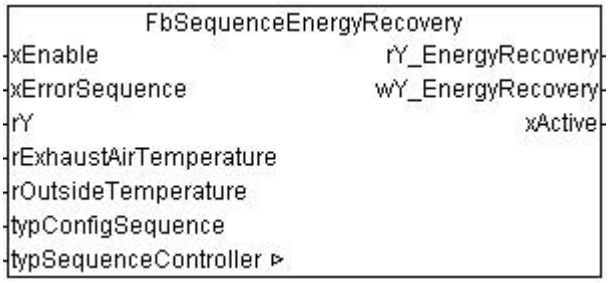
The output value ***"wY_Cooling"*** has the same meaning as the *"rY_Cooling"* output, except that the output has standardized values between 0 – 32767.

The ***"xActive"*** output indicates whether the set value for the cooling register is greater than zero.

When the *"xEnable"* input is not activated, or when a malfunction is signaled at the ***"xErrorSequence"*** input, the variable ***"typSequenceController"*** notifies the sequence controller that this sequence must be skipped.

Energy Recovery Sequence (FbSequenceEnergyRecovery)

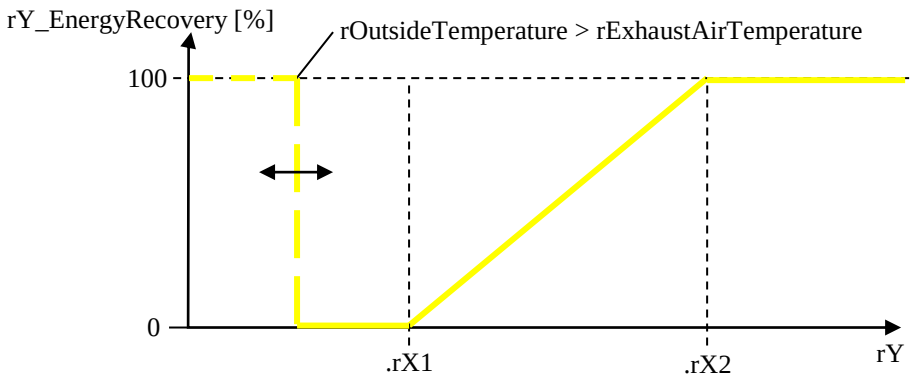
WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbSequenceEnergyRecovery	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable sequence Default setting = TRUE
xErrorSequence	BOOL	Error check-back signal from energy recovery
rY	REAL	Set value for sequence controller [%]
rExhaustAirTemperature	REAL	Actual value for exhaust air temperature [°C]
rOutsideTemperature	REAL	Actual value outside temperature [°C]
typConfigSequence	←	Configuration parameters:
.rX1	REAL	Min. setting value for energy recovery sequence [%]
.rX2	REAL	Max. setting value for energy recovery sequence [%]
.bSequenceNumber	BYTE	Sequence number (from cooling toward heating) Value range = 1 – 4
typSequenceController	←	Data structure for data exchange between the individual sequences and the sequence controller
Return value:	**Data type:**	**Comment:**
rY_EnergyRecovery	REAL	Set value for energy recovery [%]
wY_EnergyRecovery	WORD	Set value for energy recovery Value range = 0 – 32767
xActive	BOOL	Active indicator for sequence
Graphical illustration:		



Visualization objects:

ConfigSequence

Diagram:



Function description:

The **FbSequenceEnergyRecovery** function block converts the set value of the sequence controller into a setting value for energy recovery (rotary heat exchangers, plate-type heat exchangers or run-around coil system).

Configuration parameters:

The configuration structure ***"typConfigSequence"*** contains the following parameters:

- ***".rX1"*** defines the minimum set value for the sequence controller for the energy recovery sequence.
- ***".rX2"*** defines the maximum set value for the sequence controller for the energy recovery sequence.
- ***".bSequenceNumber"*** defines the number of the sequence. Sequence numbers are assigned in the HVAC system in order from cooling toward heating.

The sequence is enabled via the ***"xEnable"*** input. The output set value ***"rY_EnergyRecovery"*** is then calculated from the set value from the sequence controller ***"rY"***.

The output value "***wY_EnergyRecovery"*** has the same meaning as the *"rY_EnergyRecovery"* output, except that the output has standardized values between 0 – 32767.

When the outside temperature ***"rOutsideTemperature"*** is higher than the exhaust air temperature ***"rExhaustAirTemperature"***, the set value for energy recovery is switched to maximum output (summer function).
A hysteresis of 1 K is taken into account for the summer function.

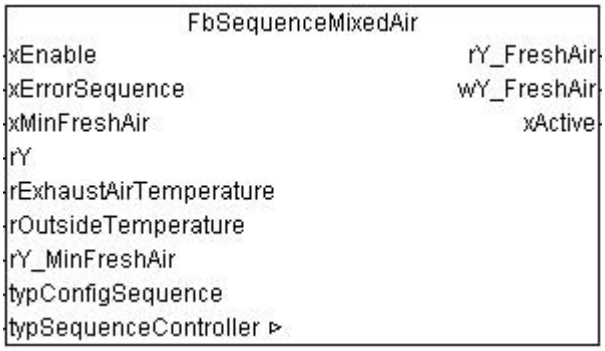
The ***"xActive"*** output indicates whether the set value for energy recovery is greater than zero.

When the *"xEnable"* input is not activated, or when a malfunction is signaled at the ***"xErrorSequence"*** input, the variable ***"typSequenceController"*** notifies the sequence controller that this sequence must be skipped.

Mixed Air Sequence (FbSequenceMixedAir)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbSequenceMixedAir	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable sequence Default setting = TRUE
xErrorSequence	BOOL	Error check-back signal from the mixed air damper
xMinFreshAir	BOOL	Override mixed air damper using the minimum fresh air rate
rY	REAL	Set value for sequence controller [%]
rExhaustAirTemperature	REAL	Actual value for exhaust air temperature [°C]
rOutsideTemperature	REAL	Actual value outside temperature [°C]
rY_MinFreshAir	REAL	Minimum fresh air rate [%]
typConfigSequence	←	Configuration parameters:
.rX1	REAL	Min. setting value for mixed air sequence [%]
.rX2	REAL	Max. setting value for mixed air sequence [%]
.bSequenceNumber	BYTE	Sequence number (from cooling toward heating) Value range = 1 – 4
typSequenceController	←	Data structure for data exchange between the individual sequences and the sequence controller
Return value:	**Data type:**	**Comment:**
rY_FreshAir	REAL	Set value mixed air damper [%]
wY_FreshAir	WORD	Set value for mixed air damper Value range = 0 – 32767
xActive	BOOL	Active indicator for sequence

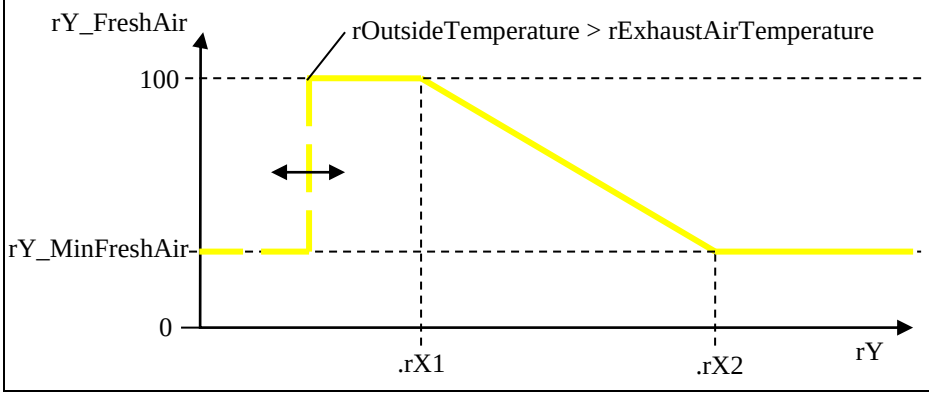
Graphical illustration:



Visualization objects:

ConfigSequence	Sequence number %d X1 sequence %2.1f [%] X2 sequence %2.1f [%]

Diagram:



Function description:

The **FbSequenceMixedAir** function block converts the set value for the sequence controller into a setting value for the mixed air damper.

Configuration parameters:

The configuration structure ***"typConfigSequence"*** contains the following parameters:

- ***".rX1"*** defines the minimum set value for the sequence controller for the mixed air sequence.
- ***".rX2"*** defines the maximum set value for the sequence controller for the mixed air sequence.
- ***".bSequenceNumber"*** defines the number of the sequence.
  Sequence numbers are assigned in the HVAC system in order from cooling toward heating.

The sequence is enabled via the ***"xEnable"*** input. The output set value ***"rY_FreshAir"*** is then calculated from the set value from the sequence controller ***"rY"***. When enabled, the minimum fresh air rate ***"rY_MinFreshAir"*** is always maintained.

The output value ***"wY_FreshAir"*** has the same meaning as the *"rY_FreshAir"* output, except that the output has standardized values between 0 – 32767.

When the outside temperature ***"rOutsideTemperature"*** is higher than the exhaust air temperature ***"rExhaustAirTemperature"***, the fresh air percentage of the minimum fresh air rate is reduced (summer function).
A hysteresis of 1 K is taken into account for the summer function.

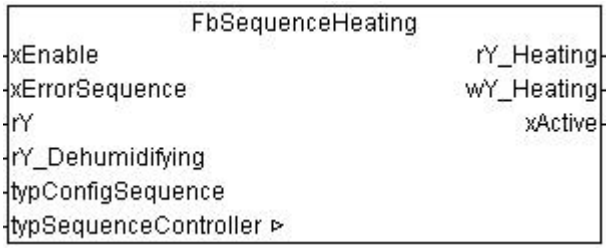
When the system is enabled, it is possible to force the fresh air percentage down to the minimum fresh air rate via the ***"xMinFreshAir"*** input.

The ***"xActive"*** output indicates whether the set value for the mixed air damper is greater than zero.

When the *"xEnable"* input is not activated, or when a malfunction is signaled at the ***"xErrorSequence"*** input, the variable ***"typSequenceController"*** notifies the sequence controller that this sequence must be skipped.

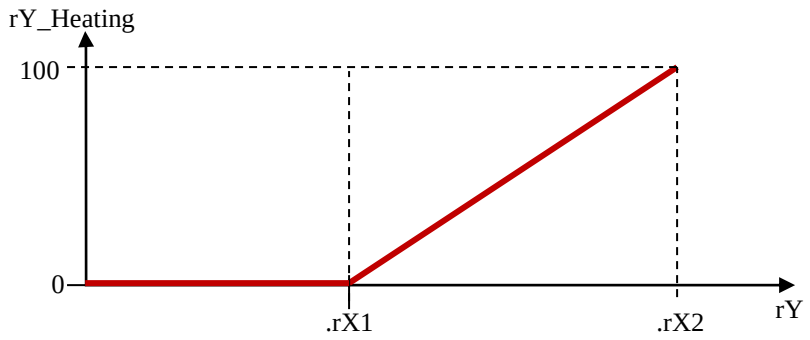
Heating Sequence (FbSequenceHeating)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbSequenceHeating	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable sequence Default setting = TRUE
xErrorSequence	BOOL	Error check-back signal from heating register
rY	REAL	Set value for sequence controller [%]
rY_Dehumidifying	REAL	Set value for dehumidifier [%]
typConfigSequence	←	Configuration parameters:
.rX1	REAL	Min. setting value for heating sequence [%]
.rX2	REAL	Max. setting value for heating sequence [%]
.bSequenceNumber	BYTE	Sequence number (from cooling toward heating) Value range = 1 – 4
typSequenceController	←	Data structure for data exchange between the individual sequences and the sequence controller
Return value:	**Data type:**	**Comment:**
rY_Heating	REAL	Valve position heating register
wY_Heating	WORD	Valve position heating register Value range = 0 – 32767
xActive	BOOL	Active indicator for sequence
Graphical illustration:		



Visualization objects:	
ConfigSequence	Sequence number %d X1 sequence %2.1f [%] X2 sequence %2.1f [%]

Diagram:



Function description:

The **FbSequenceHeating** converts the set value from the sequence controller into a setting value for the heating element.

Configuration parameters:

The configuration structure ***"typConfigSequence"*** contains the following parameters:

- ***".rX1"*** defines the minimum set value for the sequence controller for the heating sequence.
- ***".rX2"*** defines the maximum set value for the sequence controller for the heating sequence.
- ***".bSequenceNumber"*** defines the number of the sequence. Sequence numbers are assigned in the HVAC system in order from cooling toward heating.

The sequence is enabled via the ***"xEnable"*** input. The output set value ***"rY_Heating"*** is then calculated from the set value from the sequence controller ***"rY"***.

The output value ***"wY_Heating"*** has the same meaning as the *"rY_Heating"* output, except that the output has standardized values between 0 – 32767.

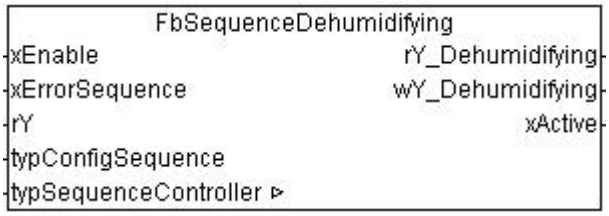
When the set value ***"rY_Dehumidifying"*** for dehumidifying is greater than zero, this sequence is inhibited for the heating register (preheater).

The ***"xActive"*** output indicates whether the set value for the heating register is greater than zero.

When the *"xEnable"* input is not activated, or when a malfunction is signaled at the ***"xErrorSequence"*** input, the variable ***"typSequenceController"*** notifies the sequence controller that this sequence must be skipped.

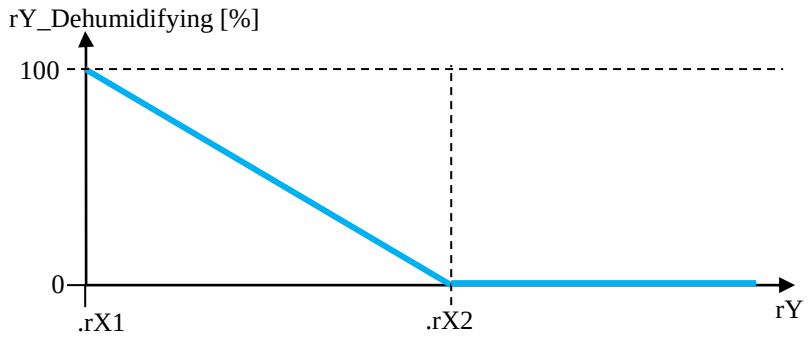
Dehumidifying Sequence (FbSequenceDehumidifying)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbSequenceDehumidifying	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable sequence Default setting = TRUE
xErrorSequence	BOOL	Error check-back signal from cooling register
rY	REAL	Set value for sequence controller [%]
typConfigSequence	←	Configuration parameters:
.rX1	REAL	Min. setting value for dehumidifying sequence [%]
.rX2	REAL	Max. setting value for dehumidifying sequence [%]
.bSequenceNumber	BYTE	Sequence number Value range = 1 – 4
typSequenceController	←	Data structure for data exchange between the individual sequences and the sequence controller
Return value:	**Data type:**	**Comment:**
rY_Dehumidifying	REAL	Valve position cooling register
wY_Dehumidifying	WORD	Valve position cooling register Value range = 0 – 32767
xActive	BOOL	Active indicator for sequence
Graphical illustration:		



Visualization objects:	
ConfigSequence	

Diagram:



Function description:

The **FbSequenceDehumidifying** function block converts the set value for the sequence controller into a setting value for the cooling register. The cooling register is actuated via a MAX logic between the set value from the cooling sequence and the set value from the dehumidifying sequence.

Configuration parameters:

The configuration structure ***"typConfigSequence"*** contains the following parameters:

- ***".rX1"*** defines the minimum set point for the sequence controller for the dehumidifying sequence.
- ***".rX2"*** defines the maximum set value for the sequence controller for the dehumidifying sequence.
- ***".bSequenceNumber"*** defines the number of the sequence.

The sequence is enabled via the ***"xEnable"*** input. The output set value ***"rY_Dehumidifying"*** is then calculated from the set value from the sequence controller ***"rY"***.

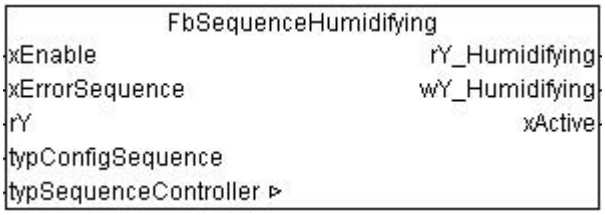
The output value ***"wY_Dehumidifying"*** has the same meaning as the "*rY_Dehumidifying"* output, except that the output has standardized values between 0 – 32767.

The ***"xActive"*** output indicates whether the set value for dehumidifying is greater than zero.

When the *"xEnable"* input is not activated, or when a malfunction is signaled at the ***"xErrorSequence"*** input, the variable ***"typSequenceController"*** notifies the sequence controller that this sequence must be skipped.

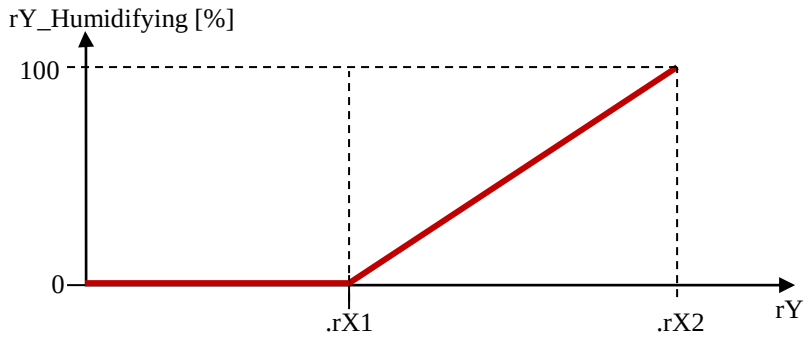
Humidifying Sequence (FbSequenceHumidifying)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbSequenceHumidifying	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable sequence Default setting = TRUE
xErrorSequence	BOOL	Error check-back signal from humidifier
rY	REAL	Set value for sequence controller [%]
typConfigSequence	←	Configuration parameters:
.rX1	REAL	Min. setting value for humidifying sequence [%]
.rX2	REAL	Max. setting value for humidifying sequence [%]
.bSequenceNumber	BYTE	Sequence number Value range = 1 – 4
typSequenceController	←	Data structure for data exchange between the individual sequences and the sequence controller
Return value:	**Data type:**	**Comment:**
rY_Humidifying	REAL	Valve setting humidifier [%]
wY_Humidifying	WORD	Valve position for the humidifier Value range = 0 – 32767
xActive	BOOL	Active indicator for sequence
Graphical illustration:		



Visualization objects:	
ConfigSequence	Sequence number %d X1 sequence %2.1f [%] X2 sequence %2.1f [%]

Diagram:



Function description:

The **FbSequenceHumidifying** function block converts the set value from the sequence controller into a setting value for the humidifier.

Configuration parameters:

The configuration structure ***"typConfigSequence"*** contains the following parameters:

- ***".rX1"*** defines the minimum set value for the sequence controller for the humidifying sequence.
- ***".rX2"*** defines the maximum set value for the sequence controller for the humidifying sequence.
- ***".bSequenceNumber"*** defines the number of the sequence.

The sequence is enabled via the ***"xEnable"*** input. The output set value ***"rY_Humidifying"*** is then calculated from the set value from the sequence controller ***"rY"***.

The output value ***"wY_Humidifying"*** has the same meaning as the *"rY_Humidifying"* output, except that the output has standardized values between 0 – 32767.

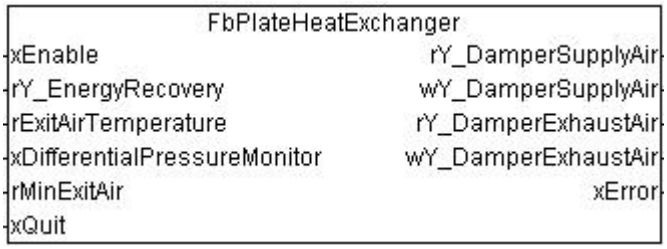
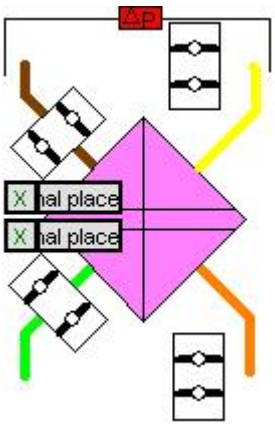
The ***"xActive"*** output indicates whether the set value for the humidifier is greater than zero.

When the *"xEnable"* input is not activated, or when a malfunction is signaled at the ***"xErrorSequence"*** input, the variable ***"typSequenceController"*** notifies the sequence controller that this sequence must be skipped.

07 Heat Exchangers

Plate-Type Heat Exchanger (FbPlateHeatExchanger)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbPlateHeatExchanger	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable plate-type heat exchanger Default setting = TRUE
rY_EnergyRecovery	REAL	Set value energy recovery sequence [%]
rExitAirTemperature	REAL	Actual value exhaust air temperature [°C]
xDifferentialPressure Monitor	BOOL	Differential pressure sensor plate-type heat exchanger Default setting = TRUE
rMinExitAir	REAL	Minimum exhaust air temperature for regulation of the plate-type heat exchanger in the incoming air duct [°C] Default setting = 6 °C
xQuit	BOOL	Error message acknowledgement
Return value:	**Data type:**	**Comment:**
rY_DamperSupplyAir	REAL	Set value for plate-type heat exchanger in incoming air duct [%] Value range = 0 – 100
wY_DamperSupplyAir	WORD	Set value for plate-type heat exchanger in the incoming air duct Value range = 0 – 32767
rY_DamperExhaustAir	REAL	Set value for plate-type heat exchanger in exhaust air duct [%] Value range = 0 – 100
wY_DamperExhaustAir	WORD	Set value for plate-type heat exchanger in the exhaust air duct Value range = 0 – 32767
xError	BOOL	Fault plate-type heat exchanger

Graphical illustration:	
	
Visualization objects:	
ConfigPlateHeat Exchanger	
PlateHeatExchanger	

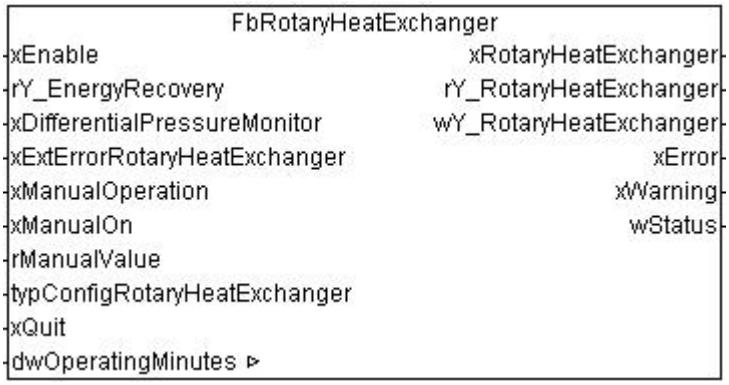
Function description:
The **FbPlateHeatExchanger** function block controls the plate-type heat exchanger. The two dampers for the exhaust air and the two dampers for the supply air of the heat exchanger are controlled separately, since it is important to prevent frost during the winter. This is accomplished by routing only a part of the supply air to the plate-type heat exchanger, while the other part is routed past the heat exchanger (bypass). Control of the plate-type heat exchanger is enabled via the ***"xEnable"*** input. When enabled, the set value for energy recovery ***"rY_EnergyRecovery"*** is passed on to the ***"rY_DamperExhaustAir"*** output. During normal operation, the entire outside air is routed to the plate-type heat exchanger. There is a risk of freezing if the exhaust air temperature ***"rExitAirTemperature"*** falls below the minimum exhaust air temperature ***"rMinExitAir"***. In this case, an internal controller ensures that the supply air dampers ***"rY_DamperSupplyAir"*** route a portion of the outside air around the plate-type heat exchanger via the bypass. The bypass dampers are always open when the unit is switched off. Fouling of the plate-type heat exchanger is detected by a differential pressure monitor ***"xDifferentialPressureMonitor"***. In order that the fouling warning message is indicated even if the system is switched off, it is saved and indicated at the ***"xError"*** output. The warning message can be acknowledged via a flank at the ***"xQuit"*** input. The output values ***"wY_DamperSupplyAir"*** and ***"wY_DamperExhaustAir"*** have the same meaning as the *"rY_DamperSupplyAir"* and *"rY_DamperExhaustAir"* outputs, except that the outputs have standardized values between 0 – 32767.

Rotary Heat Exchanger (FbRotaryHeatExchanger)

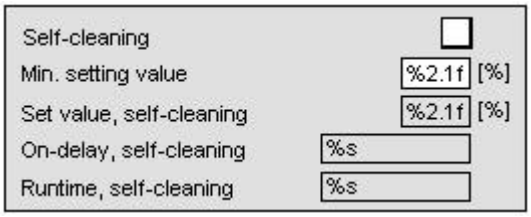
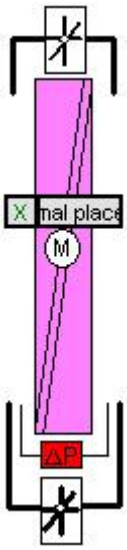
WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbRotaryHeatExchanger	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable rotary heat exchanger Default setting = TRUE
rY_EnergyRecovery	REAL	Set value energy recovery sequence [%]
xDifferentialPressure Monitor	BOOL	Differential pressure monitor for rotary heat exchanger Default setting = TRUE
xExtErrorRotaryHeat Exchanger	BOOL	External error check-back signal from the rotary heat exchanger
xManualOperation	BOOL	Enable manual operation
.xManualOn	BOOL	Switch on manually
rManualValue	REAL	Set value manual operation [%] Value range = 0 – 100
typConfigRotaryHeat Exchanger	←	Configuration parameters:
.rY_Min	REAL	Minimum set value for controlling the rotary heat exchanger [%] Default setting = 3
.rY_SelfCleaning	REAL	Set value for rotary heat exchanger during self-cleaning [%] Default setting = 0
.tMaxOff	TIME	Maximum turn-off time up to activation of self-cleaning Default setting = t#24h
.tSwitchOn	TIME	Time period for self-cleaning Default setting = t#10 m
.xSelfCleaning	BOOL	Enable self-cleaning Default setting = FALSE
xQuit	BOOL	Acknowledgement of warning and malfunction message
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating time of the rotary heat exchanger in minutes
Return value:	**Data type:**	**Comment:**
xRotaryHeatExchanger	BOOL	Switch on rotary heat exchanger and close bypass dampers.
rY_RotaryHeatExchanger	REAL	Set value rotary heat exchanger [%] Value range = 0 – 100

wY_RotaryHeatExchanger	WORD	Set value for the rotary heat exchanger Value range = 0 – 32767
xError	BOOL	Fault rotary heat exchanger
xWarning	BOOL	Differential pressure monitor tripped
wStatus	WORD	Display current status 0 = OK 1 = On 2 = Off 43 = Error RHE 44 = Differential pressure monitor

Graphical illustration:



Visualization objects:

ConfigRotaryHeat Exchanger	
RotaryHeatExchanger	

Function description:

The **FbRotaryHeatExchanger** function block controls a rotary heat exchanger. It also provides for a self-cleaning function and control of the bypass dampers.

Configuration parameters:

The configuration structure ***"typConfigRotaryHeatExchanger"*** contains the following parameters:

- ***".rY_Min"*** defines the minimum set value for the rotary heat exchanger before it is switched on.
- ***".rY_SelfCleaning"*** defines the set value for the rotary heat exchanger during self-cleaning.
- ***".xSelfCleaning"*** enables the self-cleaning function.
- ***".tMaxOff"*** defines the maximum turn-off time until self-cleaning is started.
- ***".tSwitchOn"*** defines the time period for self-cleaning.

Control of the rotary heat exchanger is enabled via the ***"xEnable"*** input. When enabled, the set value for energy recovery ***"rY_EnergyRecovery"*** is passed on to the ***"rY_RotaryHeatExchanger"*** output. At the same time, the rotary heat exchanger is switched on via the ***"xRotaryHeatExchanger"*** output and the bypass dampers closed.

The output value ***"wY_RotaryHeatExchanger"*** has the same meaning as the *"rY_RotaryHeatExchanger"* output, except that the output has standardized values between 0 – 32767.

The rotary heat exchanger can be started up at least one time within a defined time period in order to avoid fouling of the heat exchanger over extended outage periods. Self-cleaning must be activated for this.

Fouling of the rotary heat exchanger is detected by a differential pressure monitor ***"xDifferentialPressureMonitor"***. In order that the fouling warning message is indicated even if the system is switched off, it is saved and indicated at the ***"xWarning"*** output.

Using the ***"xExtErrorRotaryHeatExchanger"*** input, it is possible to monitor for an external error message from the rotary heat exchanger. If an external error occurs, the ***"xError"*** output is set. At the same time, the rotary heat exchanger is switched off and the bypass dampers opened.

The error and warning messages can be acknowledged via a flank at the ***"xQuit"*** input.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the rotary heat exchanger is switched on via the ***"xManualOn"*** input and controlled via the ***"rManualValue"*** input.

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

2.) The bypass dampers should be opened when they are de-energized.

3.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

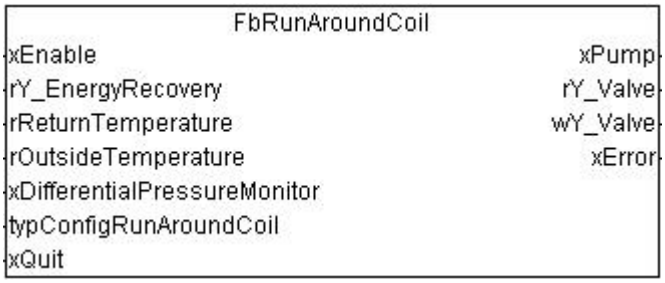
Run-Around Coil (FbRunAroundCoil)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbRunAroundCoil	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	

Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable run-around coil Default setting = TRUE
rY_EnergyRecovery	REAL	Set value energy recovery sequence [%]
rReturnTemperature	REAL	Actual value return temperature [°C]
rOutsideTemperature	REAL	Actual value outside temperature [°C]
xDifferentialPressure Monitor	BOOL	Differential pressure monitor for run-around coil system Default setting = TRUE
typConfigRunAroundCoil	←	Configuration parameters:
.rY_Min	REAL	Minimum set value for operation of the run-around coil system [%] Default setting = 3
.rMinTempFrost Protection	REAL	Minimum reference value for antifreeze controller [°C] Default setting = 10
.rMaxTempFrost Protection	REAL	Maximum reference value for the antifreeze controller [°C] Default setting = 15
.rXpFrostProtection	REAL	Proportional band for antifreeze controller Default setting = 10
xQuit	BOOL	Error acknowledgement

Return value:	**Data type:**	**Comment:**
xPump	BOOL	Pump enable
rY_Valve	REAL	Set value for 3-way valve [%] Value range = 0 – 100
wY_Valve	WORD	Set value for the 3-way valve Value range = 0 – 32767
xError	BOOL	Fault run-around coil

Graphical illustration:



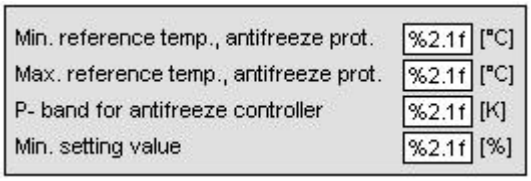
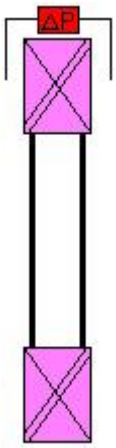
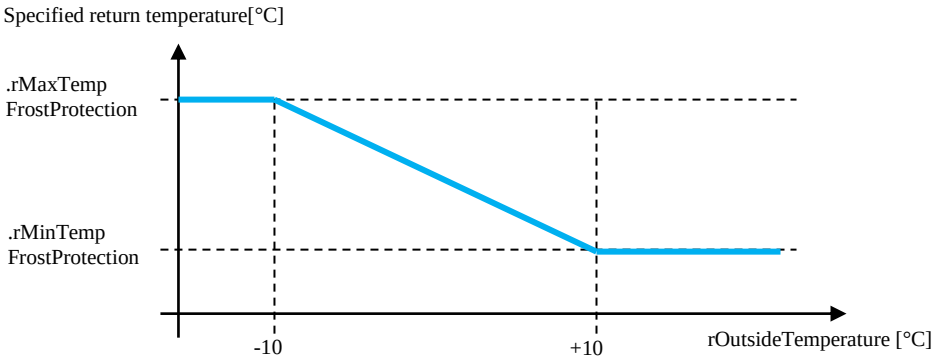
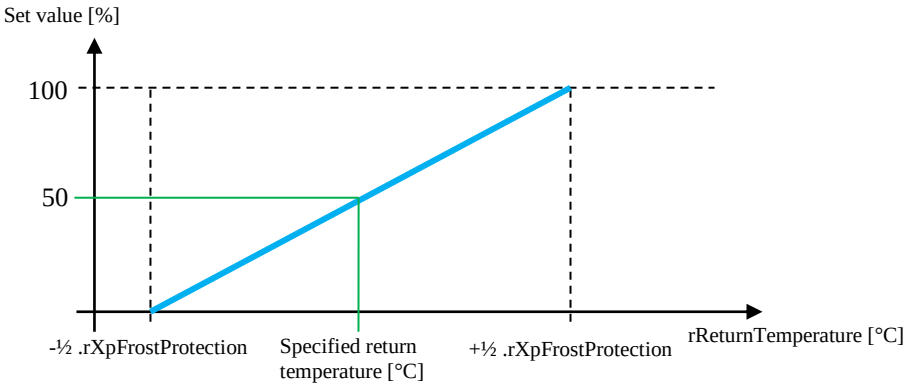
Visualization objects:	
ConfigRunAroundCoil	
RunAroundCoil	

Diagram:

Shifting of specified return temperature as a function of the outside temperature:



Control response of the return temperature controller:



Function description:

The **FbRunAroundCoil** function block controls a run-around coil system filled with glycol (air – glycol – air).

Configuration parameters:

The configuration structure ***"typConfigRunAroundCoil"*** contains the following parameters:

- ***".rY_Min"*** defines the minimum set value for the run-around coil system before it is switched on.
- ***".rMinTempFrostProtection"*** defines the minimum reference value for the return temperature at an outside temperature of 10°C.
- ***".rMaxTempFrostProtection"*** defines the maximum reference value for the return temperature at an outside temperature of –10°C.
- ***".rXpFrostProtection"*** defines the proportional band for the return temperature controller (antifreeze controller).

Control of the run-around coil system is enabled via the ***"xEnable"*** input. When enabled, the set value for energy recovery ***"rY_EnergyRecovery"*** is passed on to the ***"rY_Valve"*** output.

The return temperature ***"rReturnTemperature"*** is monitored for a minimum value to prevent any damage due to frost. The reference value for the return temperature is shifted as a function of the outside temperature ***"rOutsideTemperature"*** over a 4-point characteristic curve.

The return temperature controller is configured via a proportional band, with the set value for the valve *"rY_Valve"* determined via a MIN logic between the set value for energy recovery and the set value for the return temperature controller.

The output value ***"wY_Valve"*** has the same meaning as the *"rY_Valve"* output, except that the output has standardized values between 0 – 32767.

Icing (freezing) of the run-around coil system is detected by a differential pressure monitor ***"xDifferentialPressureMonitor"***. In order that the icing (freezing) message is indicated even if the system is switched off, the warning message is saved and indicated at the ***"xError"*** output.

The warning message can be acknowledged via a flank at the ***"xQuit"*** input.

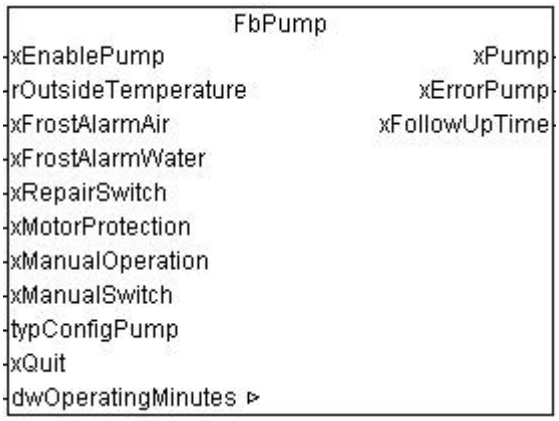
08 Pumps and Valves

Pump (FbPump)

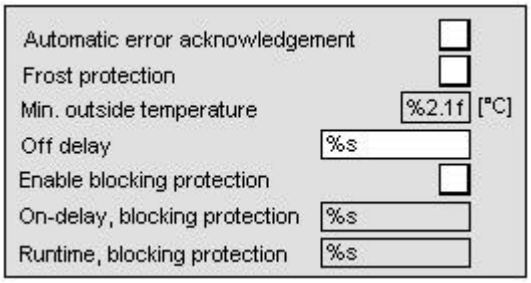
WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbPump	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnablePump	BOOL	Enable pump control
rOutsideTemperature	REAL	Actual value outside temperature [°C]
xFrostAlarmAir	BOOL	Frost alarm from FbAntifreezeAir
xFrostAlarmWater	BOOL	Frost alarm from FbAntifreezeWater
xRepairSwitch	BOOL	Repair switch pump Default setting = TRUE
xMotorProtection	BOOL	Motor protection switch pump Default setting = TRUE
xManualOperation	BOOL	Enable manual operation
xManualSwitch	BOOL	Switch on pump manually
typConfigPump	←	Configuration parameters:
.rMinOutside Temperature	REAL	Minimum outside temperature for freeze protection [°C] Default setting = 2 °C
.tOffDelay	TIME	Turn-off delay of the pump Default setting = t#15 m
.tMaxOff	TIME	Maximum turn-off time of the pump when blocking protection is enabled Default setting = t#48h
.tSwitchOn	TIME	Turn-on time of the pump when blocking protection is enabled Default setting: t#60s
.xPumpWinter	BOOL	Enable freeze protection
.xBlockingProtection	BOOL	Enable blocking protection Default setting = TRUE
.xAutoQuit	BOOL	Automatic acknowledgement of fault messages Default setting = FALSE
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating time of the pump in minutes

Return value:	Data type:	Comment:
xPump	BOOL	Switching signal for the pump
xErrorPump	BOOL	Error message pump
xFollowUpTime	BOOL	Pump is in the follow-up phase

Graphical illustration:



Visualization objects:

ConfigPump	
Pump	

Function description:

The **FbPump** function block serves to switch on pumps depending on the demand.

<u>Configuration parameters:</u>

The configuration structure ***"typConfigPump"*** contains the following parameters:

- ***".rMinOutsideTemperature"*** defines the limit for automatic switching on of the pump.
- ***".xPumpWinter"*** ensures that the pump is switched on when it is deactivated when the temperature falls below the limit *"rMinOutsideTemperature"*.
- ***".tOffDelay"*** defines the follow-up time for the pump.
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".tMaxOff"*** defines the maximum turn-off time until the blocking protection function is started.
- ***".tSwitchOn"*** defines the runtime for blocking protection.
- ***".xAutoQuit"*** acknowledges the error message as soon as the malfunction has been rectified.

Control of the pump is enabled via the ***"xEnablePump"*** input. When enabled, the pump is switched on via the ***"xPump"*** output.

When the pump enable is canceled, the pump follow-up for a defined time before it switches off. During this follow-up time, the ***"xFollowUpTime"*** output is activated.

In the Winter mode, the pump can also be switched on while deactivated when the outside temperature ***"rOutsideTemperature"*** falls below a defined limit.

The pump is also switched on even if the system is switched off for ***"xFrostAlarmAir"*** or ***"xFrostAlarmWater"***.

In order to avoid pump blocking during extended downtimes, the pump can be put into operation at least once within a defined time period. The blocking protection function must be activated for this.

If there is a pump error message at the input ***"xMotorProtection"*** or ***"xRepairSwitch"***, the pump is switched off and the ***"xErrorPump"*** output activated.

The error can be acknowledged via a positive edge at the ***"xQuit"*** input, or by automatic acknowledgement.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the pump is controlled via the ***"xManualSwitch"*** input.

Note:

1.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

2.) Blocking protection can also be activated by a timer program, so that a potential pump error message is issued only during a defined time period.

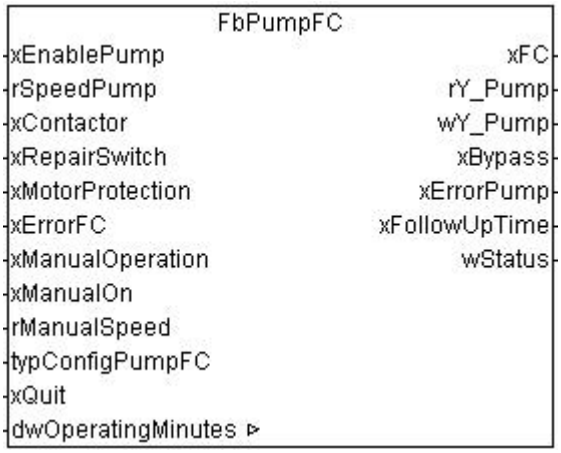
Pump with Frequency Converter (FbPumpFC)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbPumpFC	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnablePump	BOOL	Enable pump control
rSpeedPump	REAL	Pump speed in Automatic mode [%] Value range = 0 – 100
xContactor	BOOL	Contactor monitoring via auxiliary contact
xRepairSwitch	BOOL	Repair switch pump Default setting = TRUE
xMotorProtection	BOOL	Motor protection switch pump Default setting = TRUE
xErrorFC	BOOL	Fault check-back from frequency converter
.xManualOperation	BOOL	Enable manual operation
.xManualOn	BOOL	Switch on frequency converter manually
rManualSpeed	REAL	Pump speed in Manual mode [%] Value range = 0 – 100 Default setting = 50
typConfigPumpFC	←	Configuration parameters:
.tOnDelay	TIME	Runup delay for pump Default setting: t#0s
.tOffDelay	TIME	Turn-off delay of the pump Default setting = t#15 m
.tBypassDelay	TIME	Bypass protection delay time on frequency converter error Default setting: t#5s
.rY_BlockingProtection	REAL	Speed during blocking protection [%] Value range = 0 – 100 Default setting = 100
.tMaxOff	TIME	Maximum turn-off time of the pump when blocking protection is enabled Default setting = t#48h
.tSwitchOn	TIME	Turn-on time of the pump when blocking protection is enabled Default setting: t#60s
.xBlockingProtection	BOOL	Enable blocking protection Default setting = TRUE
.xAuxiliaryContact	BOOL	Auxiliary contact for contactor monitoring present Default setting = FALSE
xQuit	BOOL	Error message acknowledgement

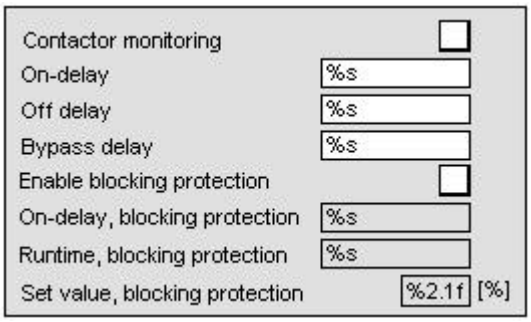
Input/output parameters:	Data type:	Comment:
dwOperatingMinutes	DWORD	Operating time of the pump in minutes

Return value:	Data type:	Comment:
xFC	BOOL	Switch on frequency converter
rY_Pump	REAL	Set value for frequency converter [%] Value range = 0 – 100
wY_Pump	WORD	Set value for frequency converter Value range = 0 – 32767
xBypass	BOOL	Bypass protection switching signal
xErrorPump	BOOL	Error message pump
xFollowUpTime	BOOL	Pump is in the follow-up phase
wStatus	WORD	Display current status 0 = OK 16 = Repair switch 17 = Motor protection switch 26 = Error Contactor contact 34 = Follow up time 38 = Error FC

Graphical illustration:



Visualization objects:

ConfigPumpFC	
PumpFC	

Function description:

The **FbPumpFC** is used for controlling and monitoring a pump, with actuation via a frequency converter.

Configuration parameters:

The configuration structure ***"typConfigPumpFC"*** contains the following parameters:

- ***".tOnDelay"*** defines the On-delay for the pump.
- ***".tOffDelay"*** defines the follow-up time for the pump.
- ***".tBypassDelay"*** defines the delay period for bypass switchover.
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".tMaxOff"*** defines the maximum turn-off time until the blocking protection function is started.
- ***".tSwitchOn"*** defines the runtime for blocking protection.
- ***".rY_BlockingProtection"*** defines the pump speed during the blocking protection function.
- ***".xAuxiliaryContact"*** indicates whether a check-back signal from the power contactor auxiliary contact is present.

Control of the pump is enabled via the ***"xEnablePump"*** input. When enabled, the frequency converter is activated via the ***"xFC"*** output.

In the Automatic mode, the required speed from the ***"rSpeedPump"*** input is output directly at the ***"rY_Pump"*** output.

The output value ***"wY_Pump"*** has the same meaning as the *"rY_Pump"* output, except that the output has standardized values between 0 – 32767.

When the pump enable is canceled, the pump follow-up for a defined time before it switches off. During this follow-up time, the ***"xFollowUpTime"*** output is activated.

A bypass contactor can be used in the event of a frequency converter malfunction.

If the frequency converter reports a malfunction via the ***"xErrorFC"*** input, the frequency converter contactor is disconnected from the pump. When contactor monitoring reports the open (disconnected) status, the bypass contactor is activated with a time delay via the ***"xBypass"*** output.

When the frequency converter malfunction is rectified, the bypass contactor is first opened and the contactor for the frequency converter re-activated with a time delay.

When contactor monitoring is activated, the input ***"xContactor"*** monitors for proper functioning of the power contactor. The switch output is compared with the check-back signal from the contactor for this. If the switch status of the contactor differs from the respective output for more than one second, there is a contactor malfunction.

In the event of a defective contactor, or an error message at the ***"xMotorProtection"*** and ***"xRepairSwitch"*** inputs, the pump is switched off and the ***"xErrorPump"*** output activated. A more detailed description of the malfunction is provided by the ***"wStatus"*** output.

The malfunction can be acknowledged via a positive flank at the ***"xQuit"*** input.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the frequency converter is activated via the ***"xManualOn"*** input and the pump controlled via the ***"rManualSpeed"*** input.

In order to avoid pump blocking during extended downtimes, the pump can be put into operation at least once within a defined time period. The blocking protection function must be activated for this. During blocking protection, the pump is controlled with a defined speed.

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

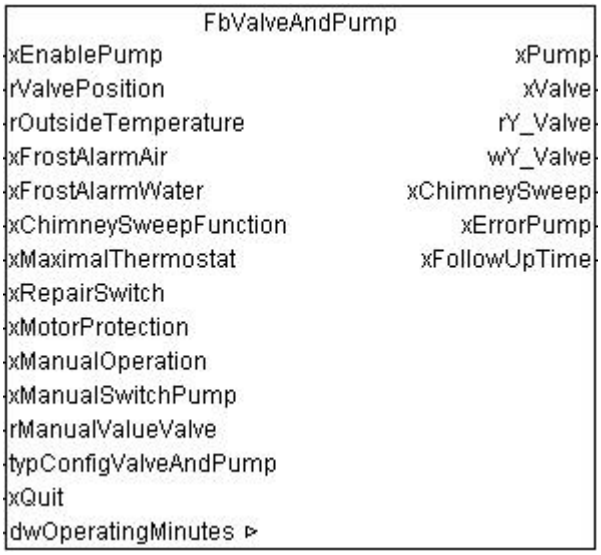
2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

3.) Blocking protection can also be activated by a timer program, so that a potential pump error message is issued only during a defined time period.

Valve and Pump (FbValveAndPump)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbValveAndPump	
Type:	Function ☐	Function block ☒ Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnablePump	BOOL	Enable pump and valve control
rValvePosition	REAL	Specified position of valve [%]
rOutsideTemperature	REAL	Actual value outside temperature [°C]
xFrostAlarmAir	BOOL	Frost alarm from FbAntifreezeAir
xFrostAlarmWater	BOOL	Frost alarm from FbAntifreezeWater
xChimneySweepFunction	REAL	Activate chimney sweep function
xMaximalThermostat	BOOL	Maximum thermostat for heating circuit Default setting = TRUE
xRepairSwitch	BOOL	Repair switch pump Default setting = TRUE
xMotorProtection	BOOL	Motor protection switch pump Default setting = TRUE
xManualOperation	BOOL	Enable manual operation
xManualSwitchPump	BOOL	Switch on pump manually
rManualValueValve	BOOL	Manually open the valve
typConfigValveAndPump	←	Configuration parameters:
.rY_Min	REAL	Minimum set value for valve [%] Default setting = 3
.rMinOutside Temperature	REAL	Minimum outside temperature for freeze protection [°C] Default setting = 2 °C
.tOffDelay	TIME	Turn-off delay of the pump Default setting = t#15 m
.rValveChimneySweep	REAL	Chimney sweep function valve position [%] Default setting = 25
.tMaxChimneySweep Function	REAL	Maximum duration of chimney sweep function Default setting = t#30 m
.tMaxOff	TIME	Maximum turn-off time of the pump when blocking protection is enabled Default setting = t#48h
.tSwitchOn	TIME	Turn-on time of the pump when blocking protection is enabled Default setting: t#60s
.xPumpWinter	BOOL	Enable freeze protection
.xThreeWayValve	BOOL	3-way valve present in supply line Default setting = TRUE

.xAutoQuit	BOOL	Automatic acknowledgement of fault messages Default setting = FALSE
.xBlockingProtection	BOOL	Enable blocking protection Default setting = TRUE
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating time of the pump in minutes
Return value:	**Data type:**	**Comment:**
xPump	BOOL	Switching signal for the pump
xValve	BOOL	Switching signal for 2-way valves
rY_Valve	REAL	Set value for 3-way valve [%] Value range = 0 – 100%
wY_Valve	WORD	Setting value for the 3-way valve Value range = 0 – 32767
xChimneySweep	BOOL	Indication of Chimney Sweep function
xErrorPump	BOOL	Error message pump
xFollowUpTime	BOOL	Pump is in the follo-up phase
Graphical illustration:		



Visualization objects:	
ConfigValveAndPump	Frost protection ☐ Automatic error acknowledgemen ☐ 3-way valve ☐ Set value, chimney sweep function %2.1f [%] Max. time, chimney sweep %s Min. outside temperature %2.1f [°C] Min. valve position for valve ON %2.1f [%] Off delay %s Enable blocking protection ☐ On-delay, blocking protection %s Runtime, blocking protection %s
ConfigValveAndPump Heating	Frost protection ☐ 3-way valve ☐ Min. outside temperature %2.1f [°C] Min. valve position for valve ON %2.1f [%] Off delay %s Enable blocking protection ☐ On-delay, blocking protection %s Runtime, blocking protection %s
ConfigValveAndPump Cooling	Continious valve ☐ Min. valve position for valve ON %2.1f [%] Off delay %s Enable blocking protection ☐ On-delay, blocking protection %s Runtime, blocking protection %s
Pump	
ThreeWayValve	
TwoWayValve	

Function description:

The **FbValveAndPump** function block serves to switch on pumps and valves depending on the demand.

Configuration parameters:

The configuration structure ***"typConfigValveAndPump"*** contains the following parameters:

- ***".rY_Min"*** defines the set value that must at least be reached to switch on the valve. Below this value rY_Valve will be 0 and xValve=FALSE.
- ***".rMinOutsideTemperature"*** defines the limit for automatic switching on of the pump.
- ***".xPumpWinter"*** ensures that the pump is switched on when it is deactivated when the temperature falls below the limit *"rMinOutsideTemperature"*.
- ***".tOffDelay"*** defines the follow-up time for the pump.
- ***".rValveChimneySweep"“*** specifies the valve position during the chimney sweep function.
- ***".tMaxChimneySweepFunction"*** specifies the maximum runtime for the chimney sweep function. The module returns to normal operation when this time period expires.
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".tMaxOff"*** defines the maximum turn-off time until the blocking protection function is started.
- ***".tSwitchOn"*** defines the runtime for blocking protection.
- ***".xThreeWayValve"*** defines whether a 3-way valve is used.
- ***".xAutoQuit"*** acknowledges the error message as soon as the malfunction has been rectified.

The pump is switched via the ***"xPump"*** output when it is either enabled via the ***"xEnablePump"*** input, or when ***"rValvePosition"*** is greater than the minimum set value.

When the switch-on conditions for the pump are no longer fulfilled and the a 3-way value has been selected in the configuration, the pump does not shut down until the defined follow-up time has elapsed. During this follow-up time, the ***"xFollowUpTime"*** output is activated.

When *"rValvePosition"* is greater than the minimum set value, the valve is opened via the ***"xValve"*** output, or the valve position is specified from the input *"rValvePosition"* to the ***"rY_Valve"*** output.

The output value ***"wY_Valve"*** has the same meaning as the *"rY_Valve"* output, except that the output has standardized values between 0 – 32767.

In the Winter mode, the pump can also be switched on while deactivated when the outside temperature ***"rOutsideTemperature"*** falls below a defined limit.

The pump is also switched on even if the system is switched off for ***"xFrostAlarmAir"*** or ***"xFrostAlarmWater"***. The valve position is unchanged and will be changed e.g. by the **FbAntifreezeAir** function block.

On initiation of the ***"xMaximalThermostat"*** maximum thermostat function = FALSE, the pump is switched off immediately and the valve closed.

If there is a pump error message at the input ***"xMotorProtection"*** or ***"xRepairSwitch"***, the pump is switched off and the ***"xErrorPump"*** output activated.

The error can be acknowledged via a positive edge at the ***"xQuit"*** input, or by automatic acknowledgement.

In order to avoid pump blocking during extended downtimes, the pump can be put into operation at least once within a defined time period. The blocking protection function must be activated for this.

When the chimney sweep function ***"xChimneySweepFunction"*** is activated, *"xPump"* switches the pump on and the valve *"rY_Valve"* is set to a configurable value. At the same time, the ***"xChimneySweep"*** indicates that the chimney sweep function is activated.

The chimney sweep function is canceled when either the *"xChimneySweepFunction"* input is reset, or when the maximum runtime for the function has elapsed.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the pump is activated via the ***"xManualSwitchPump"*** input and the valve controlled via the ***"rManualValueValve"*** input.

Note:

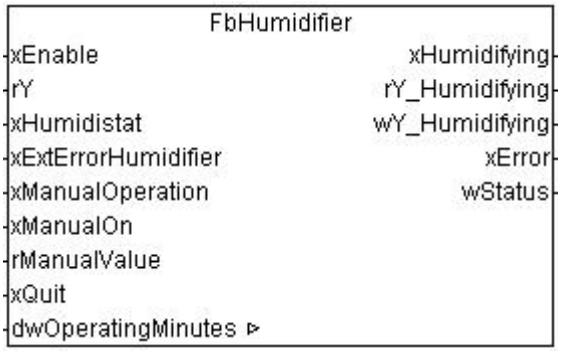
1.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

2.) Blocking protection can also be activated by a timer program, so that a potential pump error message is issued only during a defined time period.

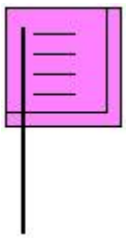
Humidifier (FbHumidifier)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbHumidifier	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable Humidifier Default setting = TRUE
rY	REAL	Set value for humidifying sequence [%]
xHumidistat	BOOL	Check-back signal from the humidistat Default setting = TRUE
xExtErrorHumidifier	BOOL	External error check-back signal from humidifier
xManualOperation	BOOL	Enable manual operation
.xManualOn	BOOL	Switch humidifier to Manual mode
rManualValue	BOOL	Manually open the valve
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating time of the pump in minutes
Return value:	**Data type:**	**Comment:**
xHumidifying	BOOL	Enable Humidifier
rY_Humidifying	REAL	Set value humidifier [%] Value range = 0 – 100%
wY_Humidifying	WORD	Set value for the humidifier Value range = 0 – 32767
xError	BOOL	Error message humidifier
wStatus	WORD	Display current status 0 = OK 1 = On 2 = Off 45 = Error Humidifier

Graphical illustration:



Visualization objects:

Humidifier	

Function description:

The **FbHumidifier** function block controls a humidifier.

Control of the pump is enabled via the ***"xEnablePump"*** input. When enabled, the pump is switched on via the ***"xPump"*** output.

Control of the humidifier is enabled via the ***"xEnable"*** input. When enabled, the set value for the humidifying sequence ***"rY"*** is passed on to the ***"rY_Humidifying"*** output. At the same time, the ***"xHumidifying"*** output is activated.

The output value ***"wY_Humidifying"*** has the same meaning as the *"rY_Humidifying"* output, except that the output has standardized values between 0 – 32767.

In the event of an error message at the ***"xHumidistat"*** and ***"xExtErrorHumidifier"*** inputs, the humidifier is switched off and the ***"xError"*** output activated. A more detailed description of the malfunction is provided by the ***"wStatus"*** output.

The malfunction can be acknowledged via a positive flank at the ***"xQuit"*** input.

Manual override is activated via the ***"xManualOperation"*** input. During manual override, the humidifier is switched on via the ***"xManualOn"*** input and the valve controlled via the ***"rManualValue"*** input.

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

09 Analog Signals

Scaling of Input Values 0 to 32767 (FuAI)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FuAI	
Type:	Function ☒	Function block ☐ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
wAI	WORD	Measured value of the analog input module Value range = 0 – 32767
rMin	REAL	Minimum output value for scaling
rMax	REAL	Maximum output value for scaling
Return value:	**Data type:**	**Comment:**
FuAI	REAL	Scaled output value
Graphical illustration:		

Function description:

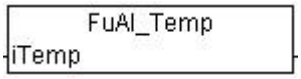
The function **FuAI** scales the measured value of the analog input modules (0 – 32767) and converts it to REAL.

The scale value range is defined via the inputs ***"rMin"*** and ***"rMax"***.

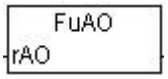
<u>Example: navigation of level 3.</u>
Active temperature sensor 0 – 10 V, measuring range –-20 °C to 60 °C
Measured temperature = 10 °C

Measured value of the input module: 16384 (5 V) ,rMin = -20; rMax = 60
Scaled measured value (REAL) = 10

Scaling of Temperature Values in °C (FuAI_Temp)

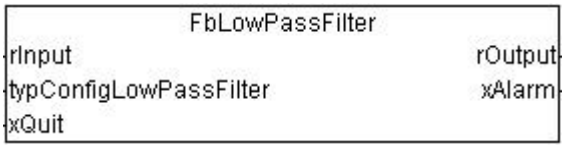
WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FuAI_Temp	
Type:	Function [X]	Function block [ ] Program [ ]
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
iTemp	INT	Temperature value is tenths of °C
Return value:	**Data type:**	**Comment:**
AI_Temp	REAL	Scaled temperature [°C]
Graphical illustration:		
		
Function description:		
The function **FuAI_Temp** scales the measured value of the resistance modules (tenth of °C) in degrees Celsius (°C) and converts it to REAL. <u>Example: navigation of level 3.</u> Measured temperature: 25.5 °C Input value of the resistance module: 255 Scaled measured value (REAL) = 25.5		

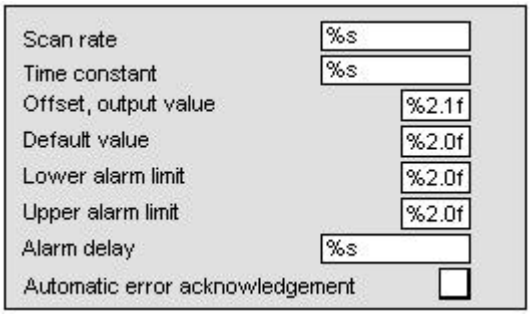
Scaling of Output Values 0 to 32767 (FuAO)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FuAO		
Type:	Function ☒	Function block ☐	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
rAO	REAL	Set value Value range = 0 – 100	
Return value:	**Data type:**	**Comment:**	
FuAO	WORD	Scaled output value Value range = 0 – 32767	
Graphical illustration:			
			
Function description:			
The function **FuAO** scales the set point as a percentage to a set value for the analog output modules (0 – 32767) . <u>Example for a 0 – 10 V signal:</u> Set value from controller (REAL): 50% Output set value (WORD): 16383 Output voltage: 5 V			

1st Order Low-Pass Filter (FbLowPassFilter)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbLowPassFilter	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
rInput	REAL	Input value
typConfigLowPassFilter	←	Configuration parameters:
.tCycleTime	TIME	Cycle time for the PT1 circuit Default = t#100ms
.tT1	TIME	Time constant for the PT1 circuit Default setting: t#2s
.rOffset	REAL	Measured value compensation for the input Default setting = 0
.rLowLimitAlarm	REAL	Lower limit for alarm Default setting = -32767
.rHighLimitAlarm	REAL	Upper limit for alarm Default setting = 32768
.tAlarm	TIME	Minimum time on limit violation until an alarm is issued. Default setting: t#10s
.rDefaultValue	REAL	Defined output value as long as the *xAlarm* output is set Default setting = 0
.xAutoQuit	BOOL	Automatic acknowledgement of the alarm Default setting = TRUE
xQuit	BOOL	Error acknowledgement
Return value:	**Data type:**	**Comment:**
rOutput	REAL	Filtered output value
xAlarm	BOOL	Input signal error
Graphical illustration:		



Visualization objects:	
ConfigLowPassFilter	

Function description:

The **FbLowPassFilter** function block is used to smoothen noisy input signals. It can also be used to define the upper and lower alarm limits.

Configuration parameters:

The configuration structure ***"typConfigLowPassFilter"*** contains the following parameters:

- ***".tCycleTime"*** defines the cycle time for the PT1 circuit (low pass).
- ***".tT1"*** defines the time constant for the PT1 circuit.
- ***".rOffset"*** enables measured value compensation for the input signal.
- ***".rLowLimitAlarm"*** defines the lower limit for issuing an alarm.
- ***".rHighLimitAlarm"*** defines the upper limit for issuing an alarm.
- ***".tAlarm"*** defines the time period for which the input value must have violated the lower or upper limit before an alarm is issued.
- ***".rDefaultValue"*** defines the output value active while the alarm is being issued.
- ***".xAutoQuit"*** acknowledges the error message as soon as the input value is again situated within the defined alarm limits.

The ***"rInput"*** input signal is smoothed via a PT1 circuit and output at the ***"rOutput"*** output.

If the input signal violates the defined limits for a defined time, an alarm message is output at the ***"xAlarm"*** output.

In this case, the *"rOutput"* output assumes the defined default setting.

The alarm can be acknowledged after elimination of the error via a positive edge at the ***"xQuit"*** input, or by automatic acknowledgement.

1st Order Low-Pass Filter for AI 0 - 32767 (FbLowPassFilterAI)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FbLowPassFilterAI		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		

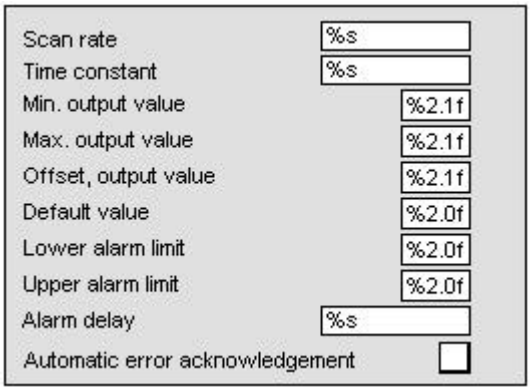
Input parameters:	**Data type:**	**Comment:**
wInput	WORD	Measured value of the analog input module Value range = 0 – 32767
typConfigLowPassFilterAI	←	Configuration parameters:
.tCycleTime	TIME	Cycle time for the PT1 circuit Default = t#100ms
.tT1	TIME	Time constant for the PT1 circuit Default setting: t#2s
.rOffset	REAL	Measured value compensation for the input Default setting = 0
.rMin	REAL	Minimum output value for scaling Default setting = 0
.rMax	REAL	Maximum output value for scaling Default setting = 32767
.rLowLimitAlarm	REAL	Lower limit for alarm Default setting = -32767
.rHighLimitAlarm	REAL	Upper limit for alarm Default setting = 32768
.tAlarm	TIME	Minimum time on limit violation until an alarm is issued. Default setting: t#10s
.rDefaultValue	REAL	Defined output value as long as the *xAlarm* output is set Default setting = 20
.xAutoQuit	BOOL	Automatic acknowledgement of the alarm Default setting = TRUE
xQuit	BOOL	Error acknowledgement

Return value:	**Data type:**	**Comment:**
rOutput	REAL	Scaled and filtered output value
xAlarm	BOOL	Analog input signal error

Graphical illustration:

Visualization objects:

ConfigLowPassFilterAI



Function description:

The **FbLowPassFilterAI** function block scales the input value and smoothens noisy input signals. It can also be used to define the upper and lower alarm limits.

Configuration parameters:

The configuration structure ***"typConfigLowPassFilterAI"*** contains the following parameters:

- ***".tCycleTime"*** defines the cycle time for the PT1 circuit (low pass).
- ***".tT1"*** defines the time constant for the PT1 circuit.
- ***".rOffset"*** enables measured value compensation for the input signal.
- ***".rMin"*** defines the minimum output value for scaling.
- ***".rMax"*** defines the maximum output value for scaling.
- ***".rLowLimitAlarm"*** defines the lower limit for issuing an alarm.
- ***".rHighLimitAlarm"*** defines the upper limit for issuing an alarm.
- ***".tAlarm"*** defines the time period for which the input value must have violated the lower or upper limit before an alarm is issued.
- ***".rDefaultValue"*** defines the output value active while the alarm is being issued.
- ***".xAutoQuit"*** acknowledges the error message as soon as the input value is again situated within the defined alarm limits.

The ***"wInput"*** input signal is scaled using a 4-point characteristic curve and smoothed via a PT1 circuit. The scaled and smoothed value is output at the ***"rOutput"*** output.

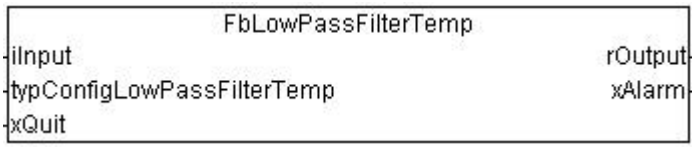
If the input signal violates the defined limits for a defined time, an alarm message is output at the ***"xAlarm"*** output.

In this case, the *"rOutput"* output assumes the defined default setting.

The alarm can be acknowledged after elimination of the error via a positive edge at the ***"xQuit"*** input, or by automatic acknowledgement.

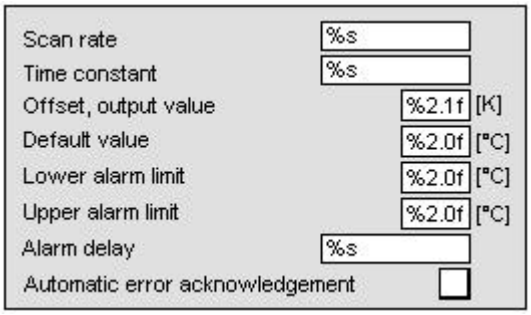
1st Order Low-Pass Filter for Temperatures (FbLowPassFilterTemp)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbLowPassFilterTemp	
Type:	Function ☐	Function block ☒ Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
iInput	INT	Temperature value is tenths of °C
typConfigLowPassFilter Temp	←	Configuration parameters:
.tCycleTime	TIME	Cycle time for the PT1 circuit Default = t#100ms
.tT1	TIME	Time constant for the PT1 circuit Default setting: t#2s
.rOffset	REAL	Measured value compensation for the input Default setting = 0
.rLowLimitAlarm	REAL	Lower limit for alarm Default setting = -32767
.rHighLimitAlarm	REAL	Upper limit for alarm Default setting = 32768
.tAlarm	TIME	Minimum time on limit violation until an alarm is issued. Default setting: t#10s
.rDefaultValue	REAL	Defined output value as long as the *xAlarm* output is set Default setting = 20
.xAutoQuit	BOOL	Automatic acknowledgement of the alarm Default setting = TRUE
xQuit	BOOL	Error acknowledgement
Return value:	**Data type:**	**Comment:**
rOutput	REAL	Scaled and filtered output value
xAlarm	BOOL	Analog input signal error
Graphical illustration:		



Visualization objects:

ConfigLowPassFilter Temp



Function description:

The **FbLowPassFilterTemp** function block scales the input value and smoothens noisy input signals. It can also be used to define the upper and lower alarm limits.

Configuration parameters:

The configuration structure ***"typConfigLowPassFilterTemp"*** contains the following parameters:

- ***".tCycleTime"*** defines the cycle time for the PT1 circuit (low pass).
- ***".tT1"*** defines the time constant for the PT1 circuit.
- ***".rOffset"*** enables measured value compensation for the input signal.
- ***".rLowLimitAlarm"*** defines the lower limit for issuing an alarm.
- ***".rHighLimitAlarm"*** defines the upper limit for issuing an alarm.
- ***".tAlarm"*** defines the time period for which the input value must have violated the lower or upper limit before an alarm is issued.
- ***".rDefaultValue"*** defines the output value active while the alarm is being issued.
- ***".xAutoQuit"*** acknowledges the error message as soon as the input value is again situated within the defined alarm limits.

The ***"iInput"*** input signal is divided by ten (°C) and smoothed via a PT1 circuit. The scaled and smoothed value is output at the ***"rOutput"*** output.

If the input signal violates the defined limits for a defined time, an alarm message is output at the ***"xAlarm"*** output.

In this case, the *"rOutput"* output assumes the defined default setting.

The alarm can be acknowledged after elimination of the error via a positive edge at the ***"xQuit"*** input, or by automatic acknowledgement.

1st Order Low-Pass Filter for Bus Signals (FbLowPassFilterBus)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbLowPassFilterBus	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	

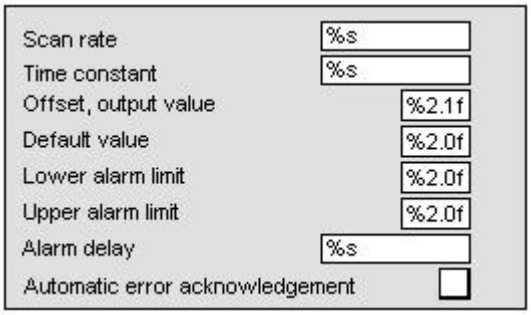
Input parameters:	**Data type:**	**Comment:**
rInput	REAL	Input value
xUpdate	BOOL	A positive edge marks a new measured value from the bus system
typConfigLowPassFilter Bus	←	Configuration parameters:
.tCycleTime	TIME	Cycle time for the PT1 circuit Default = t#100ms
.tT1	TIME	Time constant for the PT1 circuit Default setting: t#2s
.rOffset	REAL	Measured value compensation for the input Default setting = 0
.rLowLimitAlarm	REAL	Lower limit for alarm Default setting = -32767
.rHighLimitAlarm	REAL	Upper limit for alarm Default setting = 32768
.tAlarm	TIME	Minimum time on limit violation until an alarm is issued. Default setting: t#10s
.rDefaultValue	REAL	Defined output value as long as the *xAlarm* output is set Default setting = 20
.xAutoQuit	BOOL	Automatic acknowledgement of the alarm Default setting = TRUE
xQuit	BOOL	Error acknowledgement

Return value:	**Data type:**	**Comment:**
xReady	BOOL	Indicates that at least one new measured value has been received after a restart.
rOutput	REAL	Scaled and filtered output value
xAlarm	BOOL	Analog input signal error

Graphical illustration:

Visualization objects:

ConfigLowPassFilterBus



Function description:

The **FbLowPassFilterBus** function block monitors the input signal from the bus and smoothens noisy input signals. It can also be used to define the upper and lower alarm limits.

Configuration parameters:

The configuration structure ***"typConfigLowPassFilterBus"*** contains the following parameters:

- ***".tCycleTime"*** defines the cycle time for the PT1 circuit (low pass).
- ***".tT1"*** defines the time constant for the PT1 circuit.
- ***".rOffset"*** enables measured value compensation for the input signal.
- ***".rLowLimitAlarm"*** defines the lower limit for issuing an alarm.
- ***".rHighLimitAlarm"*** defines the upper limit for issuing an alarm.
- ***".tAlarm"*** defines the time period for which the input value must have violated the lower or upper limit before an alarm is issued. At the same time, this period is also used for monitoring the maximum "spacing" between two bus telegrams.
- ***".rDefaultValue"*** defines the output value after a restart and while the alarm is being issued.
- ***".xAutoQuit"*** acknowledges the error message as soon as the input value is again situated within the defined alarm limits.

The ***"rInput"*** input signal is smoothed via a PT1 circuit and output at the ***"rOutput"*** output.

The default value is given at the *"rOutput"* output as long as the sensor receives no value after a restart.

As soon as a new measured value is detected at the *"rInput"* input via a positive edge at the ***"xUpdate"*** input, the ***"xReady"*** output is set to TRUE.

If the input signal violates the defined limits for a defined time, an alarm message is output at the ***"xAlarm"*** output.

Besides the limits, the time between two measured values is also monitored. If the *"xUpdate"* input does not receive a new measured value within the configured time, an alarm is issued at the *"xAlarm"* output.

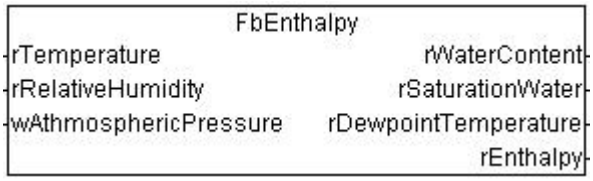
In this case, the *"rOutput"* output assumes the defined default setting.

The alarm can be acknowledged after elimination of the error via a positive edge at the ***"xQuit"*** input, or by automatic acknowledgement.

10 Temperature Evaluation

Enthalpy (FbEnthalpy)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FbEnthalpy		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
rTemperature	REAL	Current temperature [°C] Value range = -50 °C – 50 °C	
rRelativeHumidity	REAL	Relative humidity [%] Value range = 0 – 100%	
wAthmosphericPressure	WORD	Atmospheric pressure [hPa] Value range 0 hPa – 1050 hPa Default value = 1013 hPa	
Return value:	**Data type:**	**Comment:**	
rWaterContent	REAL	Water content [g/kg] Value range: 0 g/kg – 100 g/kg	
rSaturationWater	REAL	Saturated water content [g/kg] Value range 0 g/kg – 100 g/kg	
rDewpointTemperature	REAL	Dew point temperature [°C] Value range = -50 °C – 50 °C	
rEnthalpy	REAL	Enthalpy [kJ/kg] Value range: -500 – 500 kJ/kg	
Graphical illustration:			



Function description:

The function block calculates the water content ***"rWaterContent",*** the saturated water content ***"rSaturationWater"***, the dew point temperature ***"rDewpointTemperature"*** and the enthalpy ***"rEnthalpy"*** of air.

In order to calculate these values it is necessary to know the temperature **"rTemperature"** and the relative humidity ***"rRelativeHumidity"***.

Another input value is the relative pressure ***"wAthmosphericPressure".*** If the atmospheric pressure is not measured, a constant value can be chosen from the table below.

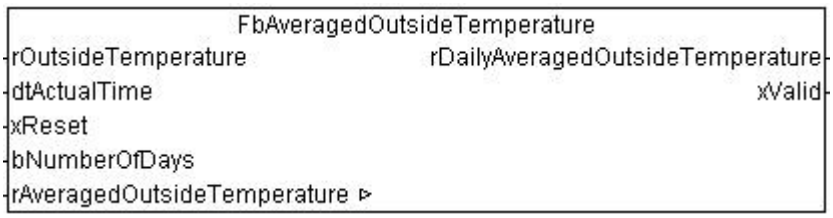
With temperatures below -15 °C the saturated water content is set to 1g/kg, with temperatures above 45 °C, the value is set to 65.4 g/kg.

With a water content of less than 1 g/kg the dew point temperature is set to -15 °C, with a water content of more than 55.6 g/kg, the value is set to 42 °C.

Altitude (above sea level)	Press
0 m	1013 hPa
300 m	980 hPa
400 m	966 hPa
600 m	943 hPa
800 m	921 hPa
1000 m	899 hPa
1500 m	842 hPa
2000 m	795 hPa

Averaged Outside Temperature (FbAveragedOutsideTemperature)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbAveragedOutsideTemperature	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
rOutsideTemperature	REAL	Actual value outside temperature [°C]
dtActualTime	DT	Actual time
xReset	BOOL	Deleting of all measured values
bNumberOfDays	BYTE	Number of days over which the average is to be taken Default setting = 3
Input/output parameters:	**Data type:**	**Comment:**
rAveragedOutside Temperature	REAL	Averaged outside temperature [°C]
Return value:	**Data type:**	**Comment:**
rDailyAveragedOutside Temperature	REAL	Outside tempeature averaged over one day
xValid	BOOL	The value for the averaged outside temperature is valid
Graphical illustration:		



Visualization objects:	
ConfigAveragedOutside Temperature	Time period for averaged temperature %d [d]

Function description:

The **FbAveragedOutsideTemperature** function block measures the outside temperature at 7:00 a.m., at 2:00 p.m. and at 7:00 p.m. The average outside temperature is calculated applying different weighting to the measured temperatures.

The current time is detected via the ***"dtActualTime"*** input. The measured outside temperature is accepted by the ***"rOutsideTemperature"*** input for calculation of the average outside temperature when the defined time of day is reached.

The number of days over which the outside temperature is to be averaged can be defined at the ***"bNumberOfDays"*** input. The input/output variable ***"rAveragedOutsideTemperature"*** indicates the outside temperature averaged over the set number of days. The ***"rDailyAveragedOutside Temperature"*** output indicates the average outside temperature for the previous day only.

The ***"xValid"*** output is TRUE when measured values for at least one day are available.

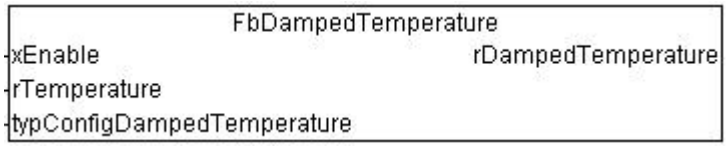
The measured values can be deleted via the ***"xReset"*** input.

Note:

The *"rAveragedOutsideTemperature"* variable should be declared as RETAIN PERSISTENT.

Damped Temperature (FbDampedTemperature)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbDampedTemperature	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Activation of averaging
rTemperature	REAL	Actual temperature [°C]
typConfigDamped Temperature	←	Configuration parameters:
.tTimeSlot	TIME	Time frame for averaging Default setting = t#60 m
.bBuffersize	BYTE	Number of values to be used for averaging Default setting = 60
Return value:	**Data type:**	**Comment:**
rDampedTemperature	REAL	Damped temperature [°C]
Graphical illustration:		



Visualization objects:	
ConfigDamped Temperature	Measuring interval %s Number of measured values %d

Function description:

The **FbDampedTemperature** function block calculates the damped temperature by averaging the temperature values measured up to a defined point (e.g., outside tmperature).

Configuration parameters:

The configuration structure ***"typConfigDampedTemperature"*** contains the following parameters:

- ***".tTimeSlot"*** defines the time period over which averaging is to be performed.
- ***".bBuffersize"*** defines the number of measured values to be saved within the defined time period.

Averaging of the temperature values is enabled via the ***"xEnable"*** input.

When this function block is enabled, the measured values from the ***"rTemperature"*** input are saved to the buffer and an average calculated from the values contained in the buffer. This average value is output at the ***"rDampedTemperature"*** output.

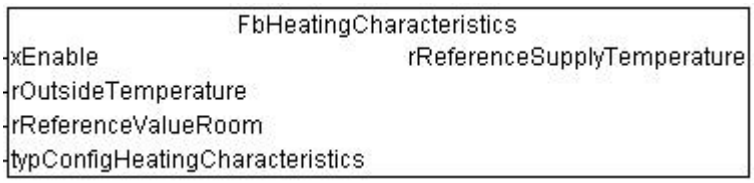
The scanning interval for the damped outside temperature is calculated as follows:

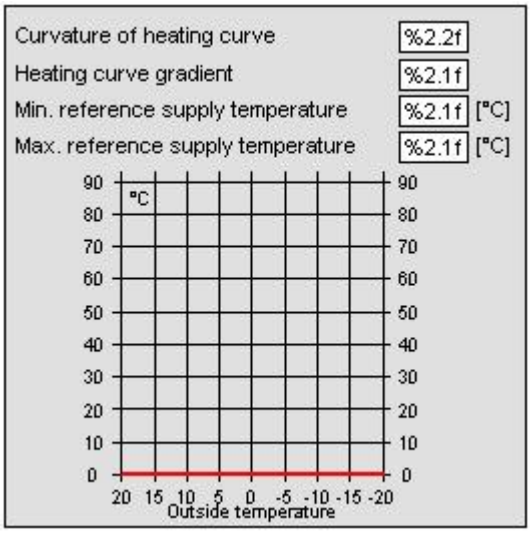
Scanning interval *=".tTimeSlot" / ".bBuffersize" = 60 min / 60 = 1min*

11 Set Value Adjustment

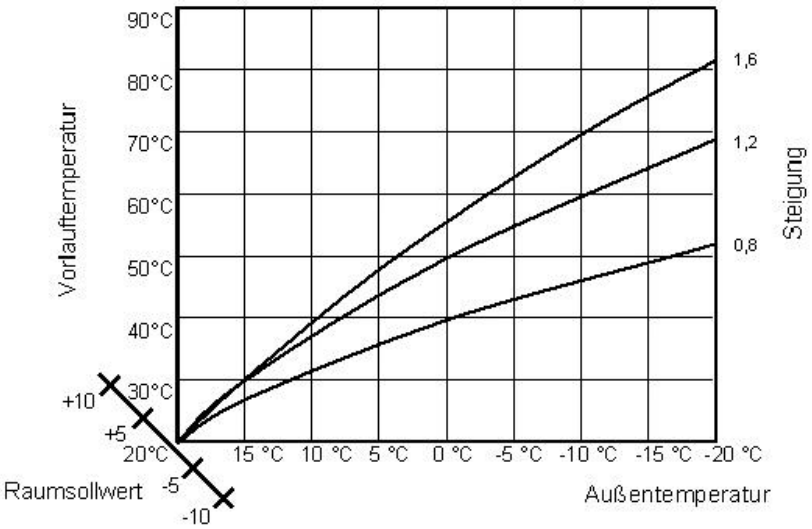
Heating Characteristics (FbHeatingCharacteristics)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbHeatingCharacteristics	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable calculation of heating curve
rOutsideTemperature	REAL	Actual value outside temperature [°C]
rReferenceValueRoom	REAL	Room temperature reference value [°C] Default setting = 20
typConfigHeating Characteristics	←	Configuration parameters:
.rCurve	REAL	Heat curve gradient curvature (heating unit exponent) Default setting = 1.33
.rGradient	REAL	Heating curve gradient Default setting = 1.6
.rMinSupply Temperature	REAL	Minimum specified supply temperature [°C] Default setting = 30
.rMaxSupply Temperature	REAL	Maximum specified supply temperature [°C] Default setting = 90
Return value:	**Data type:**	**Comment:**
rReferenceSupply Temperature	REAL	Reference value for supply temperature [°C]
Graphical illustration:		



Visualization objects:	
ConfigHeating Characteristics	

Characteristic:



Function description:

The **FbHeatingCharacteristics** heating characteristic function block calculates the reference value for the supply temperature as a function of the outside temperature. The heating characteristic is defined by slope and curvature.

Configuration parameters:

The configuration structure ***"typConfigHeatingCharacteristics"*** contains the following parameters:

- ***".rGradient"*** defines the slope of the curve for the relationship between the outside and supply temperature.
- ***".rCurve"*** defines the curvature of the characteristic so as to also account for non-linear thermal output of heating surfaces.
- ***".rMinSupplyTemperature"*** defines the lower limit for the specified supply temperature
- ***".rMaxSupplyTemperature"*** defines the upper limit for the specified supply temperature

Calculation of the supply temperature is enabled via the ***"xEnable"*** input.

The ***"rReferenceValueRoom"*** can also be used for parallel shifting of the heating characteristic.

The specified supply temperature ***"rReferenceSupplyTemperature"*** is calculated using the heating characteristic as a function of the outside temperature ***"rOutsideTemperature"***.

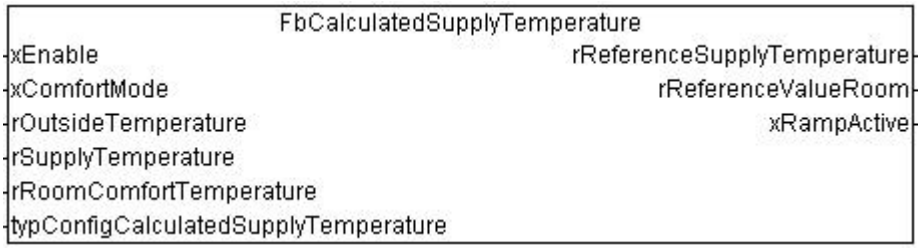
Typical heating characteristic values:

	Gradient	Curve
Radiators	1.6	1.33
In-floor heating	0.8	1.1

Calculation of Supply Temperature (FbCalculatedSupplyTemperature)

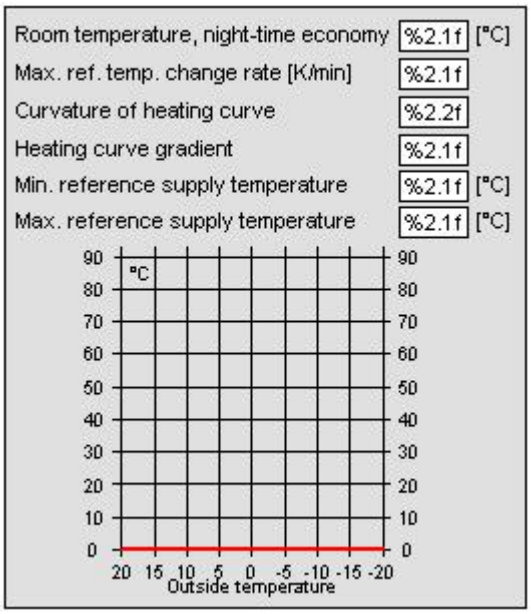
WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbCalculatedSupplyTemperature	
Type:	Function ☐	Function block ☒ Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	When the module is enable, the ramp for the specified supply temperature begins with the actual value for the supply temperature
xComfortMode	BOOL	Room reference value for heating curve TRUE = Room reference value for Comfort mode FALSE = Room reference value for night-time economy mode Default setting = FALSE
rOutsideTemperature	REAL	Actual value outside temperature [°C]
rSupplyTemperature	REAL	Actual value supply temperature [°C]
rRoomComfort Temperature	REAL	Room reference value value for Comfort mode [°C] Default setting = 20
typConfigCalculatedSupply Temperature	←	Configuration parameters:
.rRoomEconomy Temperature	REAL	Room reference value value for night-time economy mode [°C] Default setting = 14
.rCurve	REAL	Heat curve gradient curvature (heating unit exponent) Default setting = 1.33
.rGradient	REAL	Heating curve gradient Default setting = 1.6
.rMinSupply Temperature	REAL	Minimum specified supply temperature [°C] Default setting = 30
.rMaxSupply Temperature	REAL	Maximum specified supply temperature [°C] Default setting = 90
.rStepRangeRamp	REAL	Maximum value change rate per minute [K] Default setting = 1
Return value:	**Data type:**	**Comment:**
rReferenceSupply Temperature	REAL	Reference value for supply temperature [°C]
rReferenceValueRoom	REAL	Current room reference value for heating curve
xRampActive	BOOL	The ramp is active

Graphical illustration:

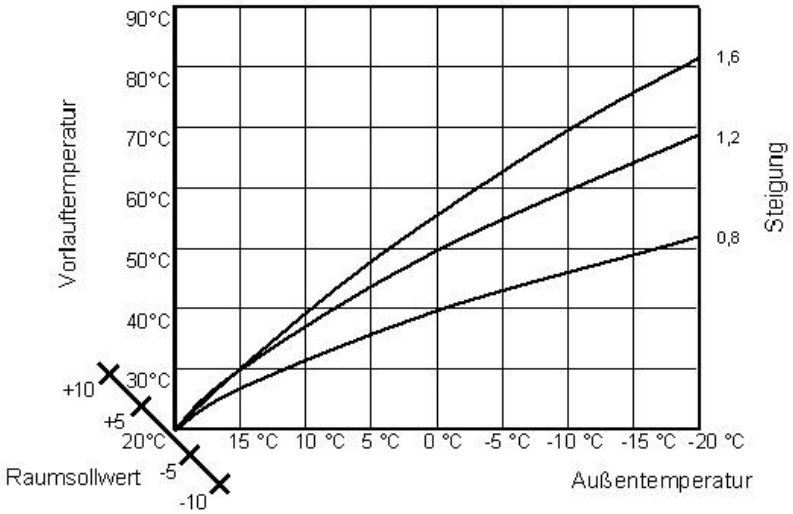


Visualization objects:

ConfigCalculatedSupply Temperature



Characteristic:



Function description:

The **FbCalculatedSupplyTemperature** function block calculates the reference value for the supply temperature as a function of the outside temperature. An additional ramp function is integrated to prevent overly rapid heating of the piping and the noises associated with this.

Configuration parameters:

The configuration structure ***"typConfigCalculatedSupplyTemperature"*** contains the following parameters:

- ***".rRoomEconomyTemperature"*** defines the specified room temperature for the night-time economy mode.
- ***".rGradient"*** defines the slope of the curve for the relationship between the outside and supply temperature.
- ***".rCurve"*** defines the curvature of the characteristic so as to also account for non-linear thermal output of heating surfaces.
- ***".rMinSupplyTemperature"*** defines the lower limit for the specified supply temperature
- ***".rMaxSupplyTemperature"*** defines the upper limit for the specified supply temperature
- ***".rStepRangeRamp"*** defines the maximum rate change for the specified supply temperature.

Calculation of the supply temperature is enabled via the ***"xEnable"*** input. When first enabled, the current supply temperature ***"rSupplyTemperature"*** is set as the starting value for the ramp function.

The specified supply temperature ***"rReferenceSupplyTemperature"*** is calculated using the heating characteristic as a function of the outside temperature ***"rOutsideTemperature"***.

The output for the reference supply temperature *"rReferenceSupplyTemperature"* keeps pace with this as long as the calculated reference temperature, and the change rate, is less than the maximum change rate that has been defined.

If the calculated reference supply temperature changes more rapidly than the defined change rate, the *"rReferenceSupplyTemperature"* output will lag behind the calculated reference supply temperature. During this time, the ***"xRampActive"*** output is set.

The reference room temperature is used for parallel shifting of the heating characteristic.

When the ***"xComfortMode"*** input is activated (Comfort mode), the value for the ***"rReferenceValueRoom"*** input is used as the reference room temperature.

When the *"xComfortMode"* input is deactivated (night-time economy mode), the temperature defined for the night-time economy mode is used as the reference room temperature.

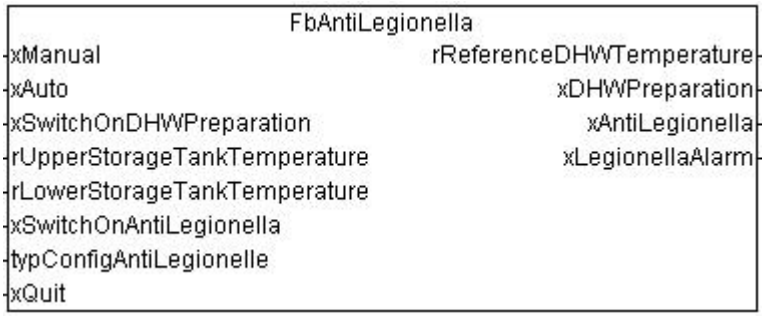
Typical values for the heating curve are:

	Gradient	Curve
Radiators	1.6	1.33
In-floor heating	0.8	1.1

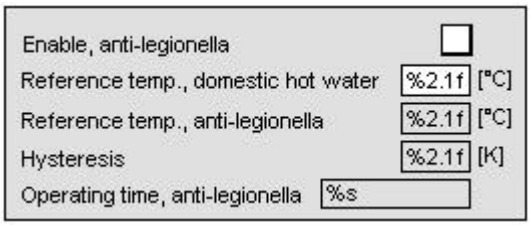
Anti-Legionnaire's Disease Function (FbAntiLegionella)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbAntiLegionella	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xManual	BOOL	Manual mode, Domestic Hot Water preparation (DHW)
xAuto	BOOL	Automatic mode for DHW preparation
xSwitchOnDHW Preparation	BOOL	Enable for DHW conditioning by timer program
rUpperStorageTank Temperature	REAL	Upper storage tank temperature sensor [°C]
rLowerStorageTank Temperature	REAL	Lower storage tank temperature sensor [°C]
xEnableAntiLegionella	BOOL	Enable for anti-Legionnaires' disease function
xSwitchOnAntiLegionella	BOOL	Activation of anti-Legionnaires' disease function by timer program
typConfigAntiLegionella	←	Configuration parameters:
.rReferenceTemperature DHW	REAL	DHW reference temperature value [°C] Default setting = 50
.rReferenceTemperature AntiLegionella	REAL	Reference value for DHW while anti-Legionnaires' disease function is active [°C] Default setting = 70
.tDurationAntiLegionella	TIME	Duration of anti-Legionnaires' disease function Default setting = t#10 m
.rHysteresis	REAL	Hysteresis for the limits [K] Default setting = 2.5
.xEnableAntiLegionella	BOOL	Enable for the Anti-Legionnaire's Disease function Default setting = FALSE
xQuit	BOOL	Acknowledgement of the anti-Legionnaires' disease alarm
Return value:	**Data type:**	**Comment:**
rReferenceDHW Temperature	REAL	DHW reference value for the DHW controller [°C]
xDHWPreparation	BOOL	Enable for DHW controller
xAntiLegionella	BOOL	Anti-Legionnaires' disease function is active
xLegionellaAlarm	BOOL	Anti-Legionnaires' disease alarm

Graphical illustration:



Visualization objects:

ConfigAntiLegionella	

Function description:

The **FbAntiLegionella** function block safeguards hot water conditioning against Legionnaire's Disease bacteria by regularly increasing the temperature of the hot water. The hot water is heated further for a set time period to a defined anti-Legionnaires's disease reference value to achieve this.

Configuration parameters:

The configuration structure ***"typConfigAntiLegionella"*** contains the following parameters:

- ***".rReferenceTemperatureDHW"*** defines the specified hot water temperature for normal operation.
- ***".rReferenceTemperatureAntiLegionella"*** defines the specified hot water temperature for the anti-Legionnaire's Disease function.
- ***".tDurationAntiLegionella"*** defines the time period over which the hot water storage tank must at least maintain the specified hot water temperature for the anti-Legionnaire's Disease function.
- ***".rHysteresis"*** defines the hysteresis for the lower and upper storage tank temperature.
- ***".xEnableAntiLegionella"*** enables the anti-Legionnaire's Disease function.

During normal operation, hot water conditioning is activated either via the ***"xManual"*** input (Manual mode), or via the ***"xAuto"*** and ***"xSwitchOnDHWPreparation"*** inputs (switching signal from the time program).

When domestic hot water preparation (DHW) is activated, domestic hot water preparation is enabled via the ***"xDHWPreparation"*** output and the reference value defined for DHW preparation output at the ***"rReferenceDHWPreparation"*** output.

When the anti-Legionnaire's Disease function is enabled, the function is started via a positive edge at the ***"xSwitchOnAntiLegionella"*** (switching signal from the timer program). The status of this function is indicated at the ***"xAntiLegionella"*** output.

When the anti-Legionnaire's Disease function is started, DHW preparation is enabled via the ***"xDHWPreparation"*** output and the reference value defined for the function output at the *"rReferenceDHWPreparation"* output.

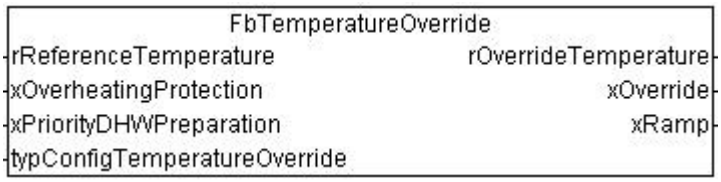
The anti-Legionnaire's Disease function is terminated after a defined time when the upper storage tank temperature ***"rUpperStorageTankTemperature"*** and the lower storage tank temperature ***"rLowerStorageTankTemperature"*** reaches the reference value for the anti-Legionnaire's Disease function, minus the hysteresis.

If the anti-Legionnaire's Disease function is terminated via the *"xSwitchOnAntiLegionella"* input without the conditions for ending the function being fulfilled, an alarm is issued at the ***"xAntiLegionellaAlarm"*** output.

The alarm can be canceled either by a flank at the ***"xQuit"*** input, or by restarting the anti-Legionnaire's Disease function via the *"xSwitchOnAntiLegionella"* input.

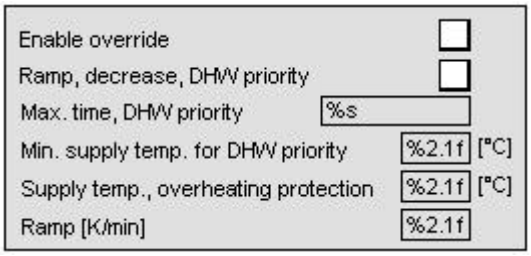
Overheating and Condensation Protection (FbTemperatureOverride)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbTemperatureOverride	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
rReferenceTemperature	REAL	Specified supply temperature [°C]
xOverheatingProtection	BOOL	Enable for overheating protection
xPriorityDHWPreparation	BOOL	Enables the domestic hot water (DHW) priority function (condensation protection)
typConfigTemperature Override	←	Configuration parameters:
.rTemperature Overheating	REAL	Specified supply temperature for overheating protection [°C] Default setting = 70
.rMinSupply Temperature	REAL	Minimum specified supply temperature [°C] Default setting = 30
.tMaxDHWPreparation	TIME	Maximum time period for DHW priority Default setting = t#60 m
.xRampPriorityDHW	BOOL	Ramp for reducing the specified supply temperature during the DHW priority function
.rStepRangeRamp	REAL	Maximum value change rate per minute [K] Default setting = 3
.xEnableOverride	BOOL	Enable for the heating override function
Return value:	**Data type:**	**Comment:**
rOverrideTemperature	REAL	Overridden specified supply temperature [°C]
xOverride	BOOL	Override status
xRamp	BOOL	Override ramp is active
Graphical illustration:		

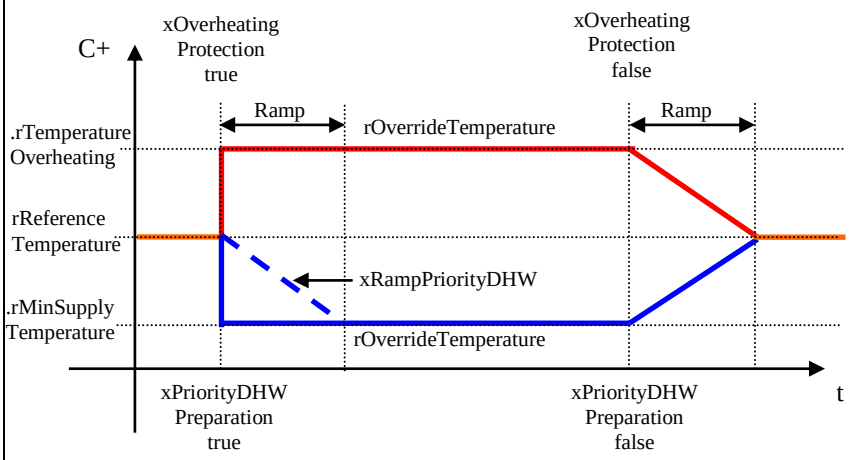


Visualization objects:

ConfigTemperature Override



Time referenced behavior:



Function description:

The **FbTemperatureOverride** function block is used for overriding the specified temperature.
If the heating unit temperature is too high, this function block can be used for forced dissipation of the heat to the downstream heating circuit. If, on the other hand, insufficient thermal output is available for domestic hot water preparation, forced reduction of the specified temperature for the heating circuit can be induced.

Configuration parameters:

The configuration structure ***"typConfigTemperatureOverride"*** contains the following parameters:

- ***".rTemperatureOverheating"*** defines the reference supply temperature for overheating protection.
- ***".rMinSupplyTemperature"*** defines the reference supply temperature for the domestic hot water priority function.
- ***".tMaxDHWPreparation"*** defines the maximum time period for the domestic hot water priority function so that the room conditions are not permanently affected by this function.
- ***".xRampPriorityDHW"*** specifies that switchover to the reference value for the domestic hot water priority function is to take place via a ramp. Otherwise, the reference value will be switched directly, without a ramp.
- ***".rStepRangeRamp"*** defines the maximum rate change for the specified supply temperature.
- ***"xEnableOverride"*** enables the override function.

The reference supply temperature from the ***"rReferenceTemperature"*** input is output directly at the ***"rOverrideTemperature"*** output as long as no override function is active.

Overheating protection is activated via the ***"xOverheatingProtection"*** input. The reference temperature for overheating protection is signaled at the output *"rOverrideTemperature"* when the overheating protection function is activated.

The DHW priority function is activated via the ***"xPriorityDHWPreparation"*** input. The reference temperature for condensation protection is signaled at the *"rOverrideTemperature"* output when the DHW priority function is activated.

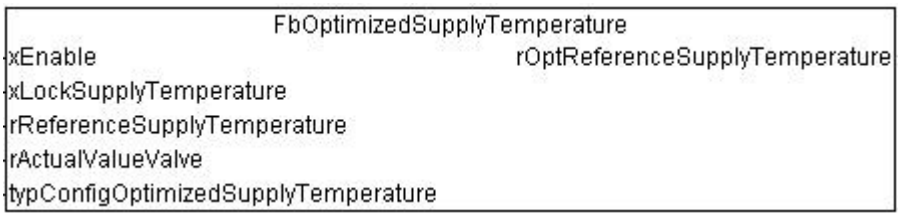
The ***"xOverride"*** output is activated as long as overheating protection of the DHW priority function is activated.

At the conclusion of the overheating protection or DHW priority function the reference supply temperature *"rOverrideTemperature"* is re-adjusted to the normal value via a ramp function.

As long as it active, the ramp function is indicated at the ***"xRamp"*** output.

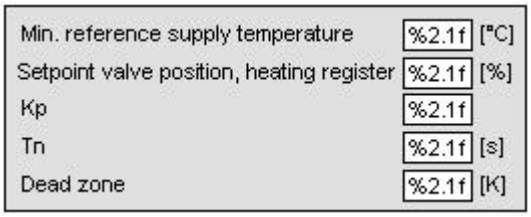
Optimized Supply Temperature (FbOptimizedSupplyTemperature)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbOptimizedSupplyTemperature	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable for optimization function Default setting = TRUE
xLockSupplyTemperature	BOOL	"Freezing" of the last reference supply temperature that was output
rReferenceSupply Temperature	REAL	Calculated supply temperature of the heating characteristic [°C] Default setting = 90 °C
rActualValueValve	REAL	Actual valve position for the heating register
typConfigOptimizedSupply Temperature	←	Configuration parameters:
.rMinSupply Temperature	REAL	Minimum specified supply temperature [°C] Default setting = 60
.rReferenceValueValve	REAL	Specified valve position for the heating register [%] Default setting = 90
.rKp	REAL	Proportional gain (P portion) Default setting = 1
.rTn	REAL	Reset time (I part) [s] Default setting = 0
.rDeadZone	REAL	Dead zone +/- [K] Default setting = 0
Return value:	**Data type:**	**Comment:**
rOptReferenceSupply Temperature	REAL	Optimized specified supply temperature
Graphical illustration:		

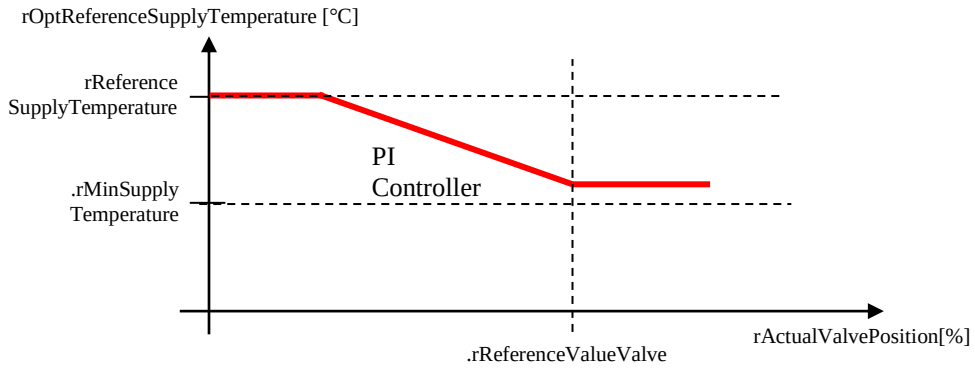


Visualization objects:

ConfigOptimizedSupply Temperature



Characteristic:



Function description:

The **FbOptimizedSupplyTemperature** function ensures that the specified supply temperature for a heating register is optimized as a function of the valve position.

<u>Configuration parameters:</u>

The configuration structure ***"typConfigOptimizedSupplyTemperature"*** contains the following parameters:

- ***".rMinSupplyTemperature"*** defines the minimum specified supply temperature at the output *"rOptReferenceSupplyTemperature".*
- ***".rReferenceValueValve"*** defines the specified valve position for optimization.
- ***".rKp"*** defines the proportional gain for the controller.
- ***".rTn"*** defines the reset time of the controller.
- ***".rDeadZone"*** defines the range around the reference value in which the set value may not be changed (dead zone).

Supply temperature optimization is enabled via the ***"xEnable"*** input.

After being enabled, the optimization module ensures that the reference supply temperature ***"rOptReferenceSupplyTemperature"*** is optimized between the ***"rReferenceSupplyTemperature"*** and the minimum reference supply temperature as a function of the valve position.

A PI controller, which determines the necessary reference supply temperature as a function of the current valve position ***"rActualValueValve"*** and the specified valve position, is used for optimization.

The last reference supply temperature that has been established ***"rOptReferenceSupplyTemperature"*** is "frozen" when the *"xLockSupplyTemperature"* input is activated.

If *"xEnable"* is not activated, *"rReferenceSupplyTemperature"* is output directly at the *"rOptReferenceSupplyTemperature"* output.

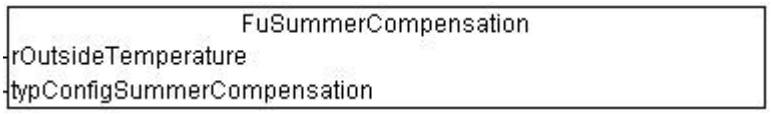
Summer Compensation (FuSummerCompensation)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FuSummerCompensation	
Type:	Function ☒	Function block ☐ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	

Input parameters:	**Data type:**	**Comment:**
rOutsideTemperature	REAL	Actual value outside temperature [°C]
typConfigSummer Compensation	←	Configuration parameters:
.rMinOutside Temperature	REAL	Lower outside temperature limit [°C] Default setting = 22
.rMaxOutside Temperature	REAL	Upper outside temperature limit [°C] Default setting = 32
.rMinReferenceValue	REAL	Minimum specified room temperature [°C] Default setting = 22
.rMaxReferenceValue	REAL	Maximum specified room temperature [°C] Default setting = 26

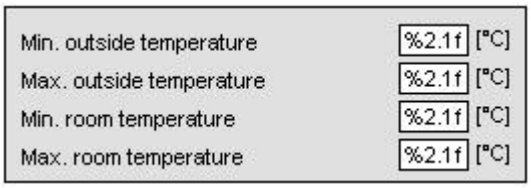
Return value:	**Data type:**	**Comment:**
FuSummerCompensation	REAL	Reference value room temperature [°C]

Graphical illustration:

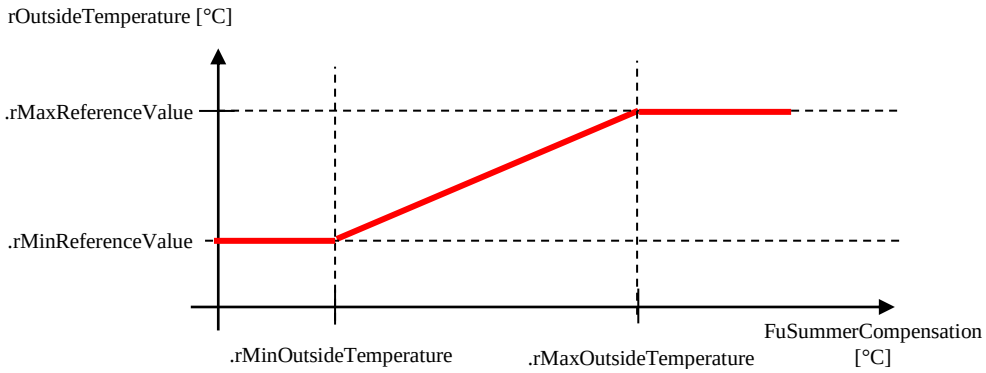


Visualization objects:

ConfigSummer Compensation



Characteristic:



Function description:

The **FuSummerCompensation** function enables the specified room temperature to be adjusted dynamically as a function of the outside temperature. This function is implemented using a linear equation with an upper and lower limit.

Configuration parameters:

The configuration structure ***"typConfigSummerCompensation"*** contains the following parameters:

- ***".rMinOutsideTemperature"*** defines the minimum outside temperature for summer compensation.
- ***".rMaxOutsideTemperature"*** defines the maximum outside temperature for summer compensation.
- ***".rMinReferenceValue"*** defines the minimum room temperature for summer compensation.
- ***".rMaxReferenceValue"*** defines the maximum room temperature for summer compensation.

The reference value for the room temperature ***"FuSummerCompensation"*** is changed as a function of the outside temperature ***"rOutsideTemperature"***.

The room temperature reference value changes according to a linear equation between the minimum and maximum outside temperature.

Note:

The default values of the temperature comply with summer compensation in accordance with VDI 1946.

Reduced Minimum Fresh Air (FbMinFreshAir)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FbMinFreshAir		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		

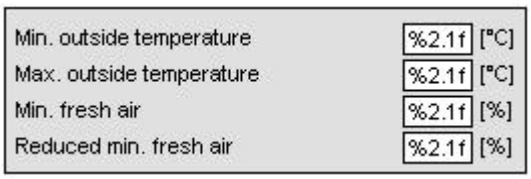
Input parameters:	**Data type:**	**Comment:**
rOutsideTemperature	REAL	Actual value of outside temperature [°C]
typConfigMinFreshAir	←	Configuration parameters:
.rMinOutside Temperature	REAL	Minimum outside temperature [°C] Default setting = 0 °C
.rMaxOutside Temperature	REAL	Maximum outside temperature [°C] Default setting = 26 °C
.rMinFreshAir	REAL	Set value for minimum fresh air rate [%] Default setting = 30
.rReducedMinFreshAir	REAL	Set value for winter fresh air rate [%] Default setting = 15

Return value:	**Data type:**	**Comment:**
rY_MinFreshAir	REAL	Set value minimum fresh air rate [%] Value range = 0 – 100

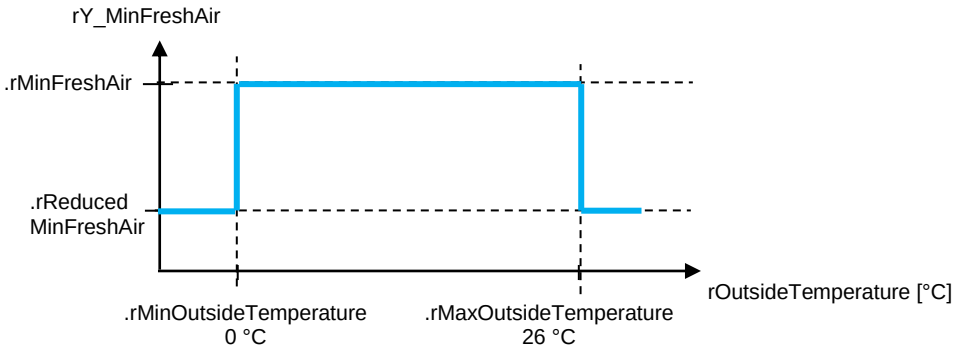
Graphical illustration:

Visualization objects:

ConfigMinFreshAir



Time referenced behavior:



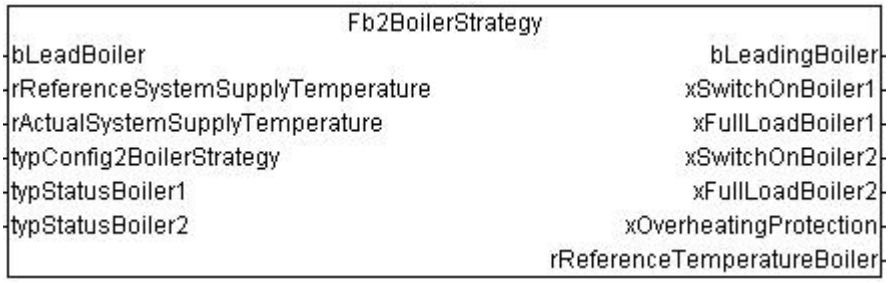
Function description:
Using the **FbMinFreshAir** function block, the minimum fresh air rate can be reduced to 50% at temperatures below 0°C or above 26°C in accordance with DIN 1946 Part 2. **Configuration parameters:** The configuration structure ***"typConfigMinFreshAir"*** contains the following parameters: • ***".rMinOutsideTemperature"*** defines the minimum outside temperature for adjusting the minimum fresh air rate. • ***".rMaxOutsideTemperature"*** defines the maximum outside temperature for adjusting the minimum fresh air rate. • ***".rMinFreshAir"*** defines the minimum fresh air rate. • ***".rReducedMinFreshAir"*** defines the reduced minimum fresh air rate. When the outside temperature ***"rOutsideTemperature"*** is within the defined limits, the set minimum fresh air rate is output at the ***"rY_MinFreshAir"*** output. Of the outside temperature *"rOutsideTemperature"* above/below the defined limits, the reduced minimum fresh air rate is output at the *"rY_MinFreshAir"* output.

12 Boiler Control

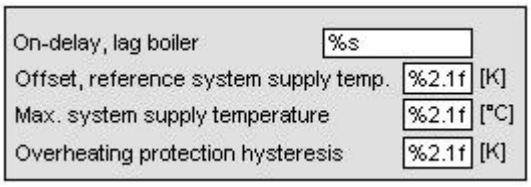
Boiler Strategy for Two Boilers (Fb2BoilerStrategy)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	Fb2BoilerStategy	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
bLeadBoiler	BYTE	Defining of lead boiler Value range = 0 – 2 (0 = both boilers in parallel) Default setting = 1
rReferenceSystemSupply Temperature	REAL	Specified system supply temperature [°C] Default setting = 60
rActualSystemSupply Temperature	REAL	Actual system supply temperature [°C]
typConfig2BoilerStategy	←	Configuration parameters:
.tSwitchOnDelayNext Boiler	TIME	Switch-on delay for second boiler Default setting = t#20 m
.rOffsetReference Temperature	REAL	Offset for the specified system supply temperature [K] Default setting = 3 K
.rMaxSystemSupply Temperature	REAL	Maximum system supply temperature for overheating protection [°C] Default setting = 85 °C
.rHysteresisOverheating Protection	REAL	Overheating protection hysteresis [K] Default setting = 5 K
typStatusBoiler1	typStatus Boiler	Status check-back signal from 1st boiler module
typStatusBoiler2	typStatus Boiler	Status check-back signal from 2nd boiler module
Return value:	**Data type:**	**Comment:**
bLeadingBoiler	BYTE	Indication of lead boiler
xSwitchOnBoiler1	BOOL	Enable Boiler 1
xFullLoadBoiler1	BOOL	Boiler 1 should run at full load
xSwitchOnBoiler2	BOOL	Enable Boiler 2
xFullLoadBoiler2	BOOL	Boiler 2 should run at full load
xOverheatingProtection	BOOL	Overheating protection active
rReferenceTemperature Boiler	REAL	Display of specified system supply temperature [°C]

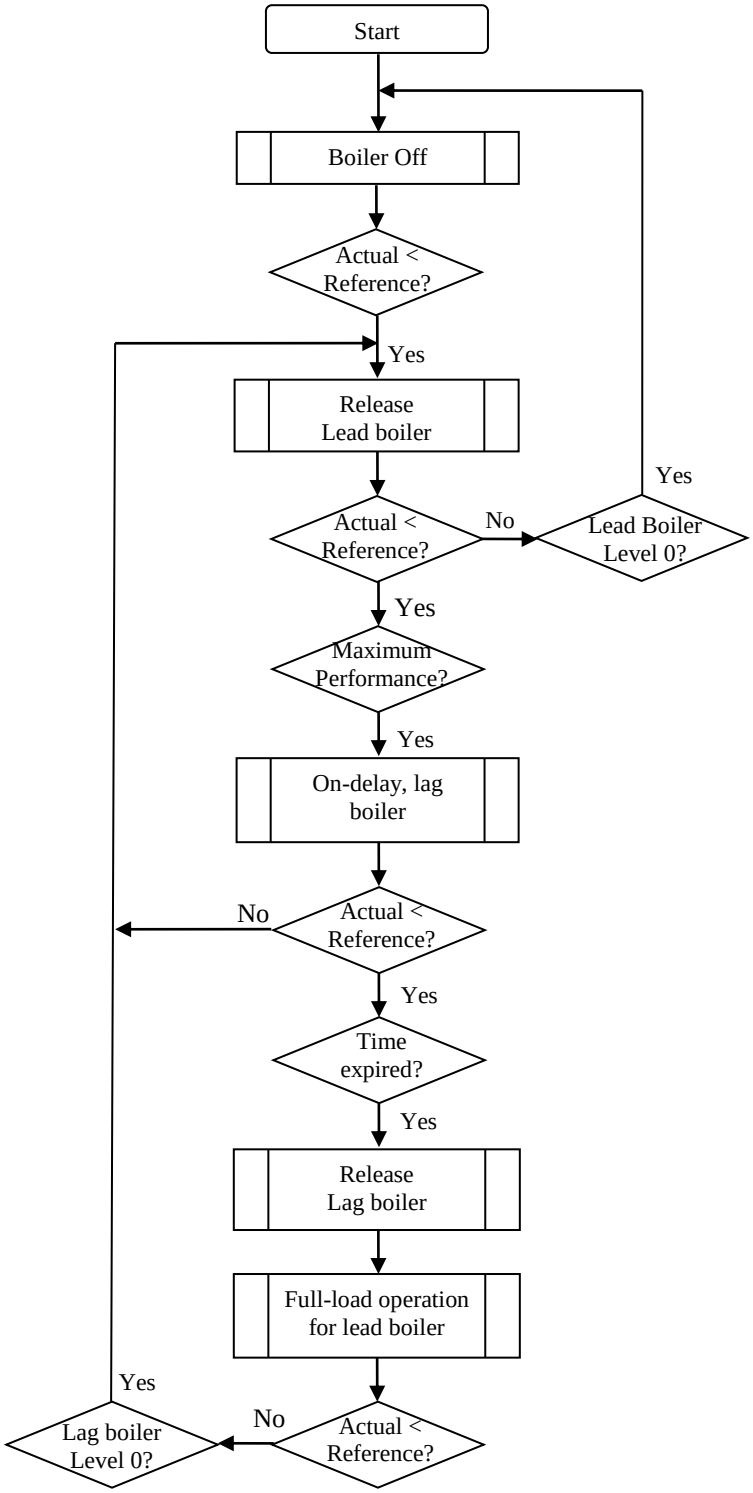
Graphical illustration:



Visualization objects:

Config2BoilerStategy	

Process diagram



Function description:

The **Fb2BoilerStrategy** function block enables a boiler sequence control be enabling the two boilers in line with current demand.

Configuration parameters:

The configuration structure ***"typConfig2BoilerStategy"*** contains the following parameters:

- ***".tSwitchOnDelayNextBoiler"*** defines the On-delay for the second boiler when the first boiler is running at full load.
- ***".rOffsetReferenceTemperature"*** defines the offset to the reference system supply temperature for the specified boiler temperature.
- ***".rMaxSystemSupplyTemperature"*** defines the maximum system supply temperature for overheating protection.
- ***".rHysteresisOverheatingProtection"*** defines the hysteresis for overheating protection.

The lead boiler can be defined for boiler sequence control via the ***"bLeadBoiler"*** input. Both boilers are controlled simultaneously if a zero is present at the *"bLeadBoiler"* input.

In the event of a boiler malfunction, the lead boiler is changed.The current lead boiler is indicated at the ***"bLeadingBoiler"*** output.

The reference system supply temperature is specified at the ***"rReferenceSystemSupplyTemperature"*** input. This can be determined, for example, via a MAX logic circuit for the requisite supply temperatures for the HVAC circuits linked to the system.

The specified boiler temperature is indicated at the ***"rReferenceTemperatureBoiler"*** and is yielded from the specified system supply temperature, plus the defined offset.

If the system supply temperature ***"rActualSystemSupplyTemperature"*** falls below the specified boiler temperature *"rReferenceTemperatureBoiler",* the lead boiler is enabled via the ***"xSwitchOnBoilerX"*** output.

When the lead boiler reaches it maximum output and the specified system supply temperature is still not achieved, the lag boiler is then enabled *"xSwitchOnBoilerX"* with a defined delay time. At the same time, the lead boiler is put into full load via the ***"xFullLoadBoilerX"*** output.

When operating at full load, the lead boiler is limited by the maximum boiler temperature. The enable function for the lag boiler remains active until the system supply temperature is achieved and the lag boiler is switched off.

As soon as the enable function for the lag boiler *"xSwitchOnBoilerX"* is canceled, the full load signal *"xFullLoadBoilerX"* from the lead boiler is also canceled.

The enable signal for the lead boiler is canceled when the system supply temperature is reached and the lead boiler is switched off.

The boiler sequence control is deactivated when one of the two boilers is in the Manual mode.

If the system supply temperature *"rActualSystemSupplyTemperature"* exceeds the defined maximum system supply temperature, the enable signal for both boilers is canceled and the ***"xOverheatingProtection"*** output activated. Overheating protection is deactivated when the system supply temperature falls below the maximum system supply temperature, minus the hysteresis.

The status check-back signal from the boiler modules is given by ***"typStatusBoilerX"***.

Boiler Module for 2-Level Boiler (Fb2LevelBoiler)

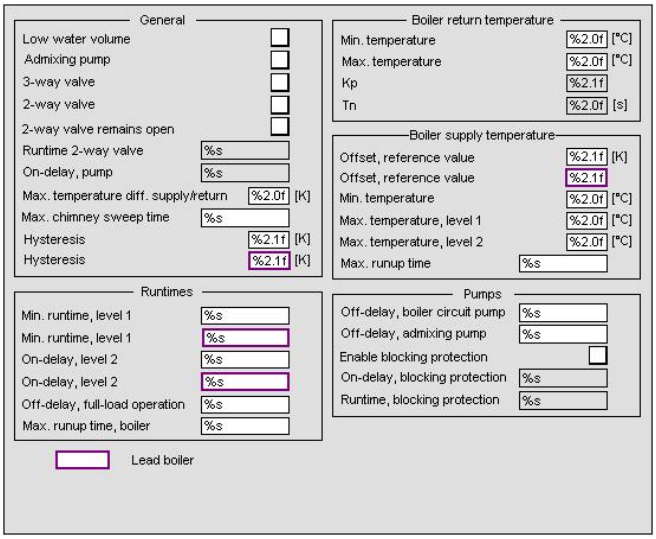
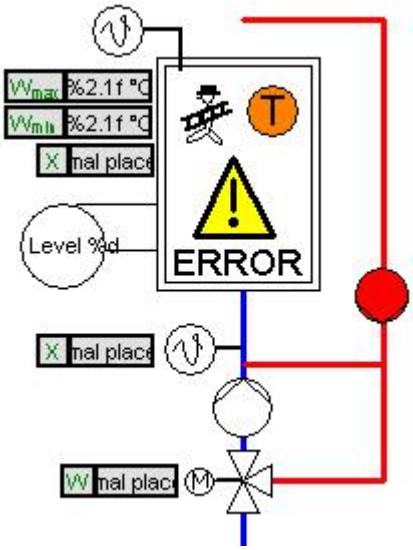
WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	Fb2LevelBoiler		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release note		
Input parameters:	**Data type:**	**Comment:**	
xManual	BOOL	Manual mode	
xAuto	BOOL	Automatic mode	
xSwitchOnBoiler	BOOL	Switch-on signal in Automatic mode (e.g., from boiler strategy)	
xFullLoad	BOOL	Boiler is to run at full load	
bLeadBoiler	BYTE	Defining of lead boiler Value range = 0 – 2 (0 = both boilers in parallel) Default setting = 1	
bBoilerNumber	BYTE	Boiler number Value range = 1 – 2 Default setting = 1	
xSafetyChain	BOOL	Check-back signal from boiler safety chain Default setting = TRUE	
rReferenceBoiler Temperature	REAL	Specified boiler temperature [°C]	
rActualBoilerTemperature	REAL	Actual boiler temperature [°C]	
rActualReturnTemperature	REAL	Actual boiler return temperature [°C]	
xChimneySweepFunction	BOOL	Activate chimney sweep function	
xLimitSwitchValve	BOOL	Limit switch, 2-way valve Default setting = TRUE	
xRepairSwitchPump	BOOL	Repair switch, Boiler circuit pump Default setting = TRUE	
xMotorProtectionPump	BOOL	Motor protection switch, Boiler circuit pump Default setting = TRUE	
xRepairSwitchAdmixing Pump	BOOL	Repair switch, Admixing pump Default setting = TRUE	
xMotorProtectionAdmixing Pump	BOOL	Motor protection switch, Admixing pump Default setting = TRUE	
xManualOperation	BOOL	Enable manual operation	
bManualLevelBoiler	BYTE	Boiler level in Manual mode Value range = 0 – 2	
xManualOnBoilerPump	BOOL	Switch on boiler circuit pump manually	
xManualOnAdmixingPump	BOOL	Switch on admixing pump manually	
rManualValueValve	REAL	Valve position in Manual mode [%]	
xFeedbackManual Operation	BOOL	External check-back signal, Manual mode	

typConfig2LevelBoiler	←	Configuration parameters:
.rOffsetReference Temperature	REAL	Offset to the specified boiler temperature [K] Default setting = 3
.rMinBoilerTemperature	REAL	Minimum boiler temperature [°C] Default setting = 50
.rMaxBoilerTemperature Level1	REAL	Maximum boiler temperature at Level 1 [°C] Default setting = 90
.rMaxBoilerTemperature Level2	REAL	Maximum boiler temperature at Level 2 [°C] Default setting = 85
.rMinReturnTemperature	REAL	Minimum return temperature [°C] Default setting = 50
.rMaxReturn Temperature	REAL	Maximum return temperature [°C] Default setting = 75
.rKpMinReturn Temperature	REAL	Proportional gain (P portion) Default setting = 12
.rTnMinReturn Temperature	REAL	Reset time of the controller [s] Default setting = 0
.tMaxChimneySweep Function	TIME	Maximum duration of chimney sweep function Default setting = t#15 m
.tMaxStartupMinBoiler Temperature	TIME	Maximum time until minimum boiler temperature is achieved during the startup process Default setting = t#10 m
.tMaxFlushPeriod	TIME	Maximum time for the startup process Default setting = t#15 m
.tMinRuntimeLevel1	TIME	Minimum runtime for lag boiler at Level 1 Default setting = t#3 m
.tSwitchOnDelayLevel2	TIME	Switch-on delay for lag boiler at Level 2 Default setting = t#20 m
.tOffDelayFullLoad	TIME	Switch-off delay for full-load operation Default setting: t#30s
.tOnDelayPump	TIME	Switch-on delay for boiler circuit pump with one 2-way valve Default setting = t#2 m
.tOffDelayPump	TIME	Switch-off delay for boiler circuit pump Default setting = t#10 m
.tOffDelayAdmixing Pump	TIME	Switch-off delay for admixing pump Default setting = t#1 m
.rDiffBoilerAndReturn Temperature	REAL	Max. temperature difference between boiler and return temperature for de-activation of the boiler circuit pump [K] Default setting = 5
.rHysteresis	REAL	Hysteresis for the individual limit values [K] Default setting = 5
.xThreeWayValve	BOOL	3-way valve present in boiler circuit Default setting = TRUE

.xTwoWayValve	BOOL	2-way valve (boiler valve) present Default setting = FALSE
.xSmallWaterVolume	BOOL	Low volume of water in boiler Default setting = FALSE
.xBlockingProtection	BOOL	Enable blocking protection Default setting = TRUE
.xAdmixingPump	BOOL	Admixing pump present Default setting = FALSE
.xTwoWayValve PermanentOpen	BOOL	2-way valve closed in the event of a fault Default setting = FALSE
.tMaxRuntimeValve	TIME	Maximum runtime for 2-way valve Default setting: t#30s
.tPumpValveMaxOFF	TIME	Maximum turn-off time of the pump and valve up to activation of blocking protection Default setting = t#24h
.tPumpValveON	TIME	Maintenance run time for pump and valve Default setting: t#30s
.tLeadMinRuntime Level1	TIME	Minimum runtime for lead boiler at Level 1 Default setting = t#5 m
.tLeadSwitchOnDelay Level2	TIME	Switch-on delay for lead boiler at Level 2 Default setting = t#15 m
.rLeadHysteresis	REAL	Hysteresis as the lead boiler for the individual limit values Default setting = 5
.rLeadOffsetReference Temperature	REAL	Offset to the specified boiler temperature as lead boiler [K] Default setting = 5
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Number of operating minutes of the boiler
Return value:	**Data type:**	**Comment:**
xLevel1	BOOL	Boiler Level 1
xLevel2	BOOL	Boiler Level 2
xBoilerPump	BOOL	Boiler circuit pump
xValve	BOOL	2-way valve
rY_Valve	REAL	Valve position, 3-way valve [%] Value range = 0 – 100
wY_Valve	WORD	Valve position, 3-way valve Value range = 0 – 32767
xAdmixingPump	BOOL	Switching signal for the admixing pump
xChimneySweep	BOOL	Chimney sweep function active
xCondensationProtection	BOOL	Condensation protection
rMinBoilerTemperature	REAL	Minimum boiler temperature

xErrorBoilerPump	BOOL	Boiler circuit pump error
xErrorAdmixingPump	BOOL	Admixing pump error
xErrorStartUp	BOOL	Start-up system fault
typStatusBoiler	typStatus Boiler	Boiler status for the strategy module
wStatus	WORD	Display current status 0 = OK 2 = Off 10 = Error Pump 29 = Error Safety chain 30 = Condensation protection 31 = Chimney sweep function 32 = Overheating protection 33 = Error Valve 34 = Follow up time 35 = Manual operation 39 = Reference value exceeded 40 = Startup behaviour

Graphical illustration:

Visualization objects:	
Config2LevelBoiler	
TwoLevelBoiler	

Function description:

The **Fb2LevelBoiler** function block contains various startup processes based on the valves and pumps used in the specific configuration and also regulates a 2-level boiler.

Configuration parameters:

The configuration structure ***"typConfig2LevelBoiler"*** contains the following parameters:

- ***"rOffsetReferenceTemperature"*** and ***"rLeadOffsetReference Temperature"*** (parameter for lead boiler) defines the offset to the specified boiler temperature.
- ***".rMinBoilerTemperature"*** defines the minimum boiler temperature at which the boiler may be operated.
- ***".rMaxBoilerTemperatureLevel1"*** defines the maximum boiler temperature that results in the boiler being shut down.
- ***".rMaxBoilerTemperatureLevel2"*** defines the maximum boiler temperature that results in switching back to Level 1.
- ***".rMaxReturnTemperature"*** defines the maximum return temperature that results in the boiler being shut down.
- ***".rMinReturnTemperature"*** defines the minimum return temperature at which the boiler should be operated.
- ***".rKpMinReturnTemperature"*** defines the proportional gain for minimum return temperature control.
- ***".rTnMinReturnTemperature"*** defines the reset time for minimum return temperature control.
- ***".tMaxChimneySweepFunction"*** defines the maximum time period for the chimney sweep function.
- ***".tMaxStartUpMinBoilerTemperature"*** defines the maximum startup time for reaching the minimum boiler temperature.
- ***".tMaxFlushPeriod"*** defines the maximum startup time for reaching the minimum return temperature.
- ***".tMinRuntimeLevel1"*** and ***".tLeadMinRuntimeLevel1"*** (parameter for lead boiler) define the minimum runtime for the boiler at Level 1.
- ***".tSwitchOnDelayLevel2"*** and ***".tLeadSwitchOnDelayLevel2"*** define the On-delay for the second boiler level.
- ***".tOffDelayFullLoad"*** defines the follow-up time for the full-load mode.
- ***".tOnDelayPump"*** defines the On-delay for the pump with a series-connected boiler valve.
- ***".tOffDelayPump"*** defines the minimum follow-up time for the boiler circuit pump.
- ***".tOffDelayAdmixingPump"*** defines the follow-up time for the admixing pump.
- ***".rDiffBoilerAndReturnTemperature"*** defines the maximum temperature difference between the boiler temperature and the return temperature for shutting down the boiler pump.

- ***".rHysteresis"*** and ***".rLeadHysteresis"*** (parameter for the lead boiler) define the hysteresis for the respective limits.
- ***".xThreeWayValve"*** indicates for the startup circuit whether a 3-way valve is available for the admixing function.
- ***".xTwoWayValve"*** indicates whether a boiler valve is available.
- ***".xSmallWaterVolume"*** indicates for the startup circuit whether only a small volume of water is being routed through the boiler.
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".xAdmixingPump"*** indicates for the startup circuit whether an admixing pump is available for increasing the return temperature.
- ***".xTwoWayValvePermanentOpen"*** enables the boiler valve to be kept permanently open.
- ***".tMaxRuntimeValve"*** defines the maximum runtime for the boiler valve.
- ***".tPumpValveMaxOff"*** defines the maximum turn-off time for the blocking protection function.
- ***".tPumpValveOn"*** defines the runtime for the blocking protection function.

The boiler is activated either via the ***"xManual"*** input (Manual mode), or via the two inputs ***"xAuto"*** and ***"xSwitchOnBoiler"***.

When activated, the minimum boiler supply temperature is output for evaluation of the system supply temperature at the ***"rMinBoilerTemperature"*** output.

The specific boiler number ***"bBoilerNumber"*** and the number of the lead boiler ***"bLeadBoiler"*** determine whether the boiler is the lead or lag boiler. If both of these numbers are the same, the parameters for the lead boiler will be used.

Different startup procedures can apply, depending on the valve being used and the water volume:

2-way valve with large volume of water:

1.) Switch on the admixing pump ***"xAdmixingPump"***.

2.) Level 1 ***"xLevel1"*** is activated when the boiler temperature ***"rActualBoilerTemperature"*** is less than the specified boiler temperature ***"rReferenceBoilerTemperature"***, plus the defined offset.

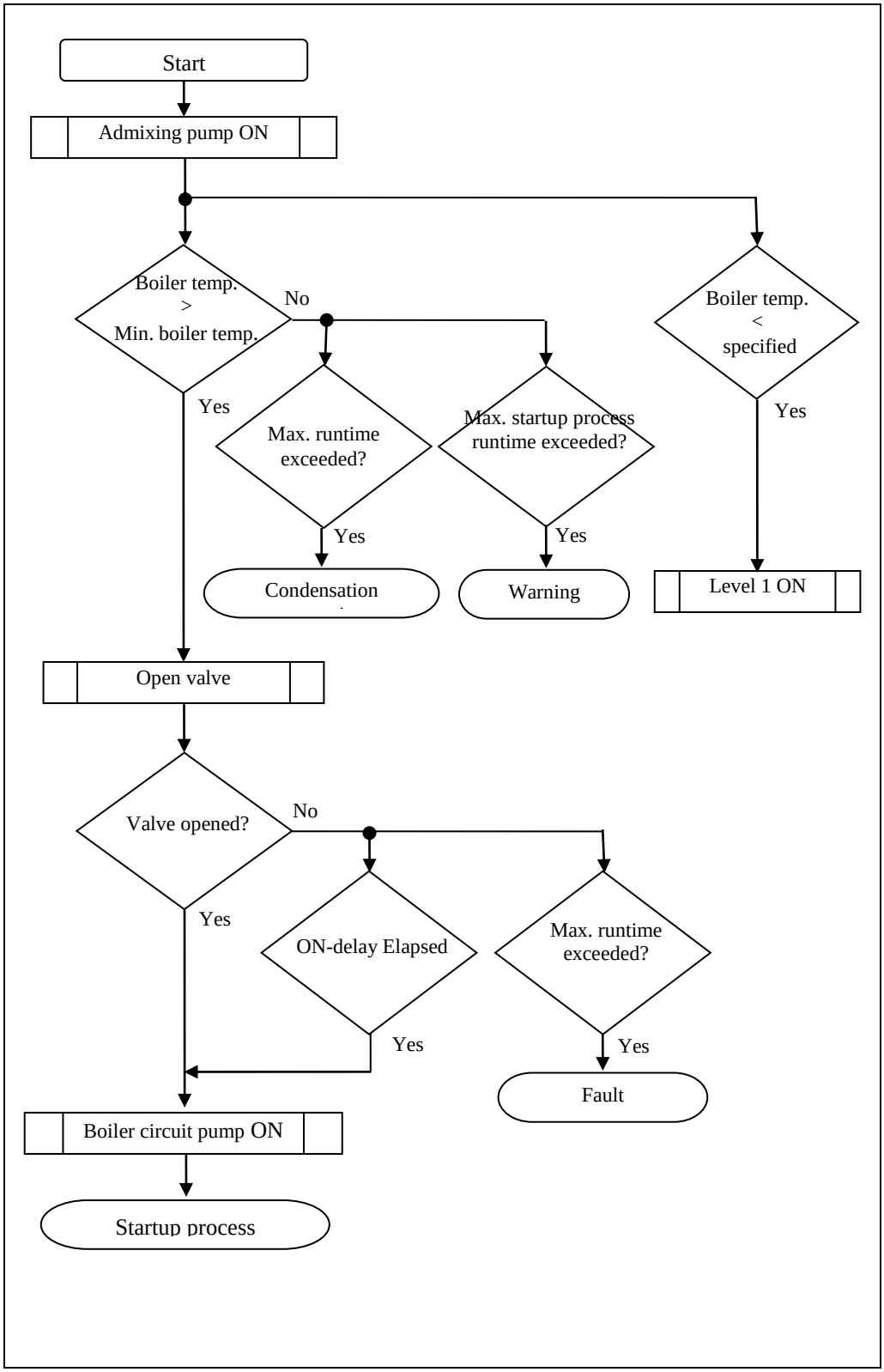
3.) The 2-way valve ***"xValve"*** is opened when the minimum boiler temperature is exceeded.

4.) If the boiler temperature fails to reach the minimum boiler temperature within a defined time, condensation protection "***xCondensationProtection"*** is activated and this indicated at the "wStatus" output.

5.) The boiler circuit pump ***"xBoilerPump"*** is switched on when the defined delay period has elapsed, or when a positive edge at the ***"xLimitSwitchValve"*** input reports the open status of the valve.

6.) If the 2-way valve fails to reach its final position within the defined runtime, the boiler is switched off and an error message output at the *"wStatus"* output.

7.) The startup procedure is terminated once the boiler circuit pump has been switched on. If the startup procedure exceeds the maximum defined time, the ***"xErrorStartUp"*** output is set and a warning issued at the *"wStatus"* output.


Start
Admixing pump ON
Boiler temp.
>
Min. boiler temp.
No
Yes
Max. runtime exceeded?
Max. startup process runtime exceeded?
Boiler temp.
<
specified
Yes
Yes
Yes
Condensation
Warning
Level 1 ON
Open valve
Valve opened?
No
Yes
ON-delay Elapsed
Max. runtime exceeded?
Yes
Yes
Fault
Boiler circuit pump ON
Startup process

2-Way Valve with Low Water Volume

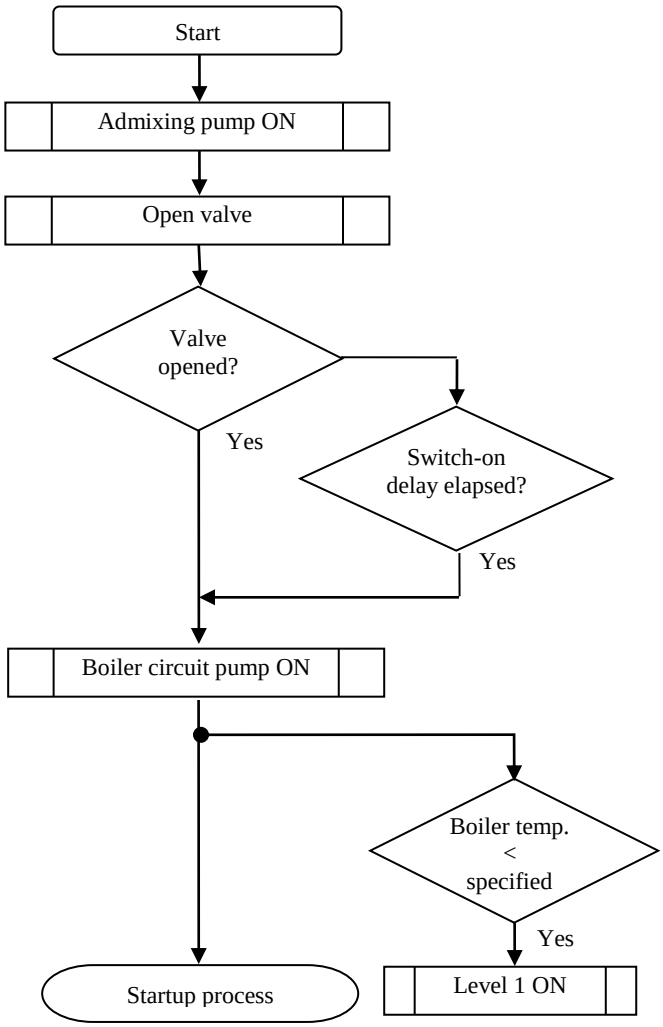
1.) Switch on the admixing pump *"xAdmixingPump"*.

2.) Open the 2-way valve *"xValve"*

3.) The boiler circuit pump *"xBoilerPump"* is switched on when the On-delay for the pump has elapsed, or when a positive edge at the *"xLimitSwitchValve"* input reports the open status of the valve. The startup process is also terminated at the same time.

4.) Level 1 *"xLevel1"* is activated when the boiler temperature *"rActualBoilerTemperature"* is less than the specified boiler temperature *"rReferenceBoilerTemperature"*.

5.) If the 2-way valve fails to reach its final position within the defined runtime, the boiler is switched off and an error message output at the *"wStatus"* output.



3-way valve

1.) Switch on the admixing pump *"xAdmixingPump"*.

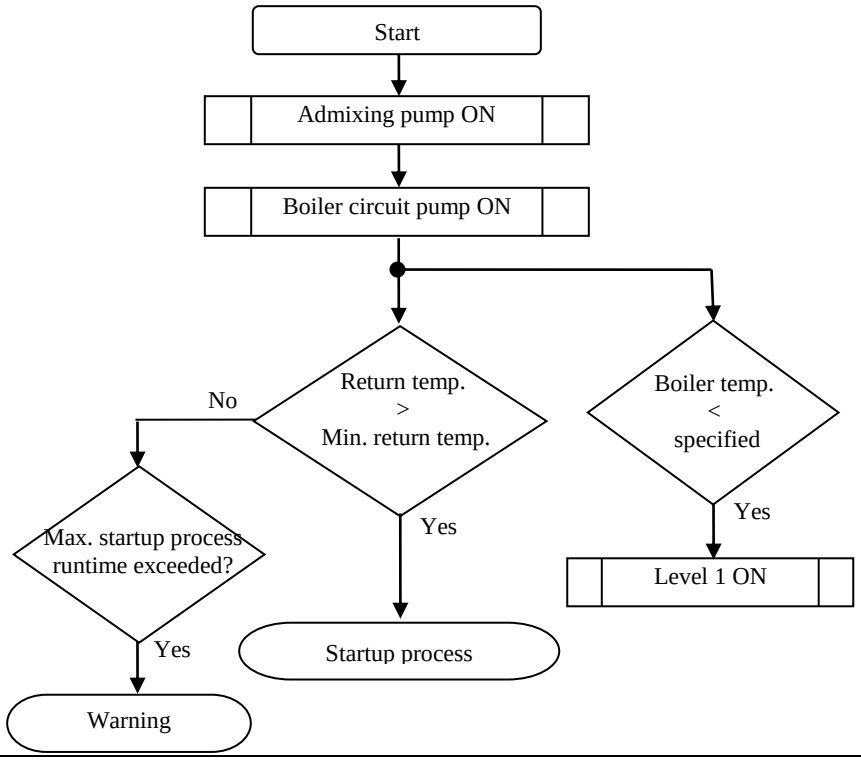
2.) Switch on the boiler circuit pump *"xBoilerPump"*

3.) Level 1 *"xLevel1"* is activated when the boiler temperature *"rActualBoilerTemperature"* is less than the specified boiler temperature *"rReferenceBoilerTemperature"*.

4.) 3-way valve ***"rY_Valve"*** is closed (boiler circuit)

5.) The startup procedure is terminated as soos as the return temperature rises above the minimum return temperature.

6.) If the minimum return temperature is not reached with the defined time, the ***"xErrorStartUp"*** output is activated and a warning issued via the *"wStatus"* output.



After the startup procedure the boiler remains at the first level for at least the minimum switch-on time.

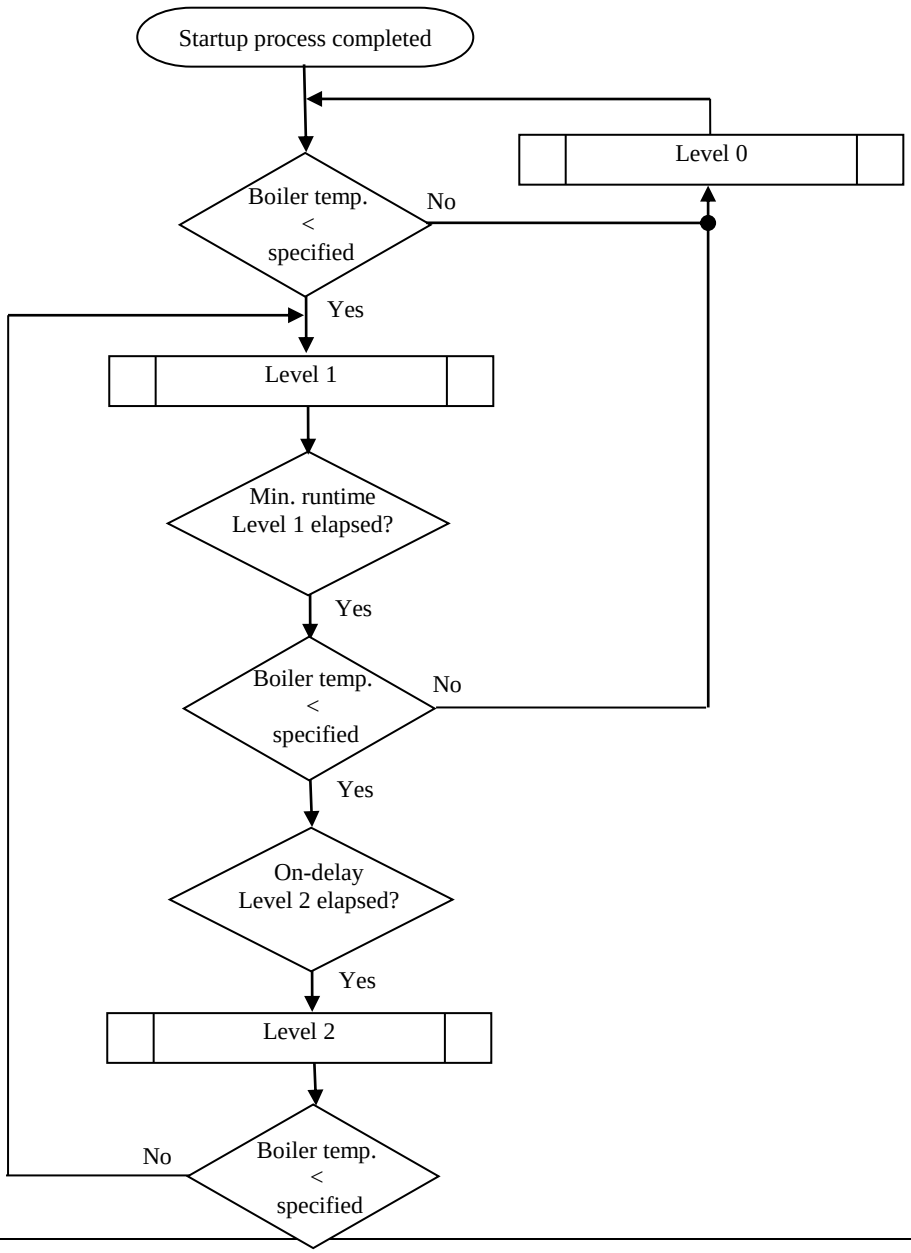
If the boiler fails to reach its specified boiler temperature within the defined time, the boiler is switched to Level 2 ***"xLevel2"***.

When the boiler then reaches its specified boiler temperature, it is switched back from Level 2 to Level 1.

The boiler is switched to Level 2 again when the temperature falls below the specified boiler temperature, minus hysteresis.

The boiler switches from Level 1 to Level 0 when the specified boiler temperature is maintained for the minimum switch-on time for the Level 1 time.

The boiler is switched back to Level 1 if its temperature falls below the specified boiler temperature at Level 0.



If there is a 3-way valve in the boiler return line, the minimum return temperature is permanently maintained during ongoing operation. A PI controller is used for the minimum return temperature.

The admixing pump *"xAdmixingPump"* is switched on during ongoing operation when the temperature falls below the minimum boiler temperature or the minimum return temperature when a 2-way valve is available. If the temperature drops below the minimum boiler temperature, the *"xCondensationProtection"* output is also set.

The boiler circuit pump continues to run when the boiler is switched off until the Switch-off delay time elapses and the difference between *"rActualBoilerTemperature"* and *"rActualReturnTemperature"* is less than the defined difference. The valve in the return line is not closed until the boiler circuit pump is switched off.

If the ***"xFullLoad"*** input is set using the strategy module, the boiler module no longer regulates the temperature in line with its specified boiler temperature, but is controlled only by maximum limiting.

The necessary information about the boiler is supplied to the strategy module through the ***"typStatusBoiler"*** structure.

The ***"xSafetyChain"*** input monitors the safety chain for the boiler. As soon as this input is switched to FALSE, the boiler is switched off and a corresponding error message indicated at the *"wStatus"* output.

In the event of a malfunction with the boiler circuit pump caused by the motor protection switch ***"xMotorProtectionPump"*** or the repair switch ***"xRepairSwitchPump"***, the boiler is switched off and the error indicated at the *"wStatus"* and ***"xErrorBoilerPump"*** output.

In the event of a malfunction of the admixing pump caused by the motor protection switch ***"xMotorProtectionAdmixingPump"*** or the repair switch ***"xRepairSwitchAdmixingPump"***, the admixing pump is switched off and the error indicated at the ***"xErrorAdmixingPump"*** output.

The error messages can be acknowledged via a flank at the ***"xQuit"*** input.

When the chimney sweep function ***"xChimneySweepFunction"*** is activated, the boiler switches on with an elevated reference value (maximum boiler temperature Level 2). The "***xChimneySweep"*** output is set as a check-back signal that the chimney sweep function has been activated. The chimney sweep function is canceled when the *"xChimneySweepFunction"* input is deactivated, or when the maximum runtime has elapsed.

In the Manual mode **"xManualOperation"** the boiler level is controlled via ***"bManualLevelBoiler",*** the boiler circuit pump via ***"xManualOnBoilerPump",*** the admixing pump via ***"xManualOnAdmixingPump"*** and the valve via ***"rManualValueValve"***.

If the boiler is switched to the Manual mode via an external circuit, a check-back signal should be transmitted to the boiler module via the ***"xFeedbackManualOperation"*** so that automatic control can be deactivated.

The pump or valve can be put through a maintenance run to prevent them from blocking during extended outage periods. Blocking protection must be activated for this.

The blocking protection function ensures that the pump and the valve do not remain switched off/closed longer than the specified monitoring period. On expiration of this time period, the pump and the valve are activated one after the other for the maintenance run for the defined time.

The output value ***"wY_Valve"*** has the same meaning as the *"rY_Valve"* output, except that the output has standardized values between 0 – 32767.

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

3.) The input *"xLimitSwitchValve"* must be set to TRUE when a 2-way valve without a limit switch is used.

Boiler Module for Modulating Boiler (FbModulatingBoiler)

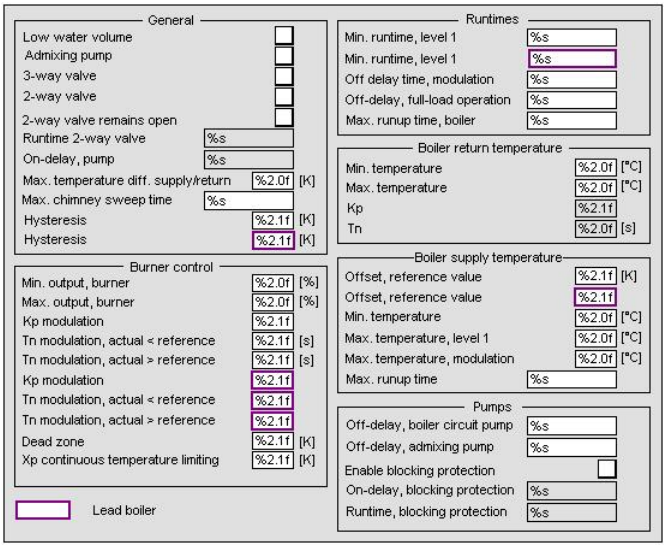
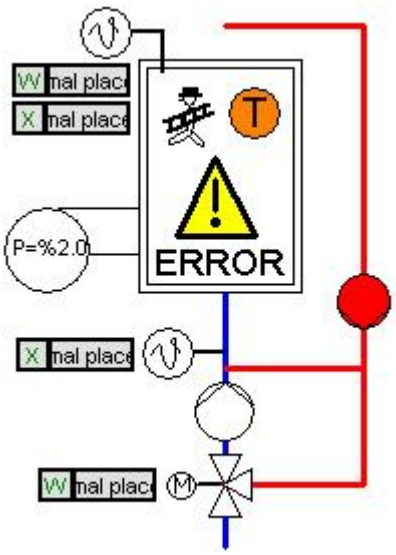
WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FbModulatingBoiler		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
xManual	BOOL	Manual mode	
xAuto	BOOL	Automatic mode	
xSwitchOnBoiler	BOOL	Switch-on signal in Automatic mode (e.g., from boiler strategy)	
xFullLoad	BOOL	Boiler is to run at full load	
bLeadBoiler	BYTE	Defining of lead boiler Value range = 0 – 2 (0 = both boilers in parallel) Default setting = 1	
bBoilerNumber	BYTE	Boiler number Value range = 1 – 2 Default setting = 1	
xSafetyChain	BOOL	Check-back signal from boiler safety chain Default setting = TRUE	
rReferenceBoiler Temperature	REAL	Specified boiler temperature [°C]	
rActualBoilerTemperature	REAL	Actual boiler temperature [°C]	
rActualReturnTemperature	REAL	Actual boiler return temperature [°C]	
xChimneySweepFunction	BOOL	Activate chimney sweep function	
xLimitSwitchValve	BOOL	Limit switch, 2-way valve Default setting = TRUE	
xRepairSwitchPump	BOOL	Repair switch, Boiler circuit pump Default setting = TRUE	
xMotorProtectionPump	BOOL	Motor protection switch, Boiler circuit pump Default setting = TRUE	
xRepairSwitchAdmixing Pump	BOOL	Repair switch, Admixing pump Default setting = TRUE	
xMotorProtectionAdmixing Pump	BOOL	Motor protection switch, Admixing pump Default setting = TRUE	
xManualOperation	BOOL	Enable manual operation	
xManualOnBurner	BOOL	Switch on burner manually	
rManualValueBoiler	REAL	Boiler output in Manual mode [%] Value range = 0 – 100	
xManualOnBoilerPump	BOOL	Switch on boiler circuit pump manually	
xManualOnAdmixingPump	BOOL	Switch on admixing pump manually	
rManualValueValve	REAL	Valve position in Manual mode [%]	

xFeedbackManual Operation	BOOL	External check-back signal, Manual mode
typConfigModulatingBoiler	←	Configuration parameters:
.rOffsetReference Temperature	REAL	Offset to the specified boiler temperature [K] Default setting = 3
.rMinBoilerTemperature	REAL	Minimum boiler temperature [°C] Default setting = 50
.rMaxBoilerTemperature Level1	REAL	Maximum boiler temperature at Level 1 [°C] Default setting = 90
.rMaxBoilerTemperature Modulating	REAL	Maximum boiler temperature during modulation [°C] Default setting = 85
.rMinReturnTemperature	REAL	Minimum return temperature [°C] Default setting = 50
.rMaxReturn Temperature	REAL	Maximum return temperature [°C] Default setting = 75
.rKpMinReturn Temperature	REAL	Proportional gain (P portion) Default setting = 12
.rTnMinReturn Temperature	REAL	Reset time of the controller [s] Default setting = 0
.tMaxChimneySweep Function	TIME	Maximum duration of chimney sweep function Default setting = t#15 m
.tMaxStartupMinBoiler Temperature	TIME	Maximum time until minimum boiler temperature is achieved during the startup process Default setting = t#10 m
.tMaxFlushPeriod	TIME	Maximum time for the startup process Default setting = t#15 m
.tMinRuntimeLevel1	TIME	Minimum runtime for lag boiler at Level 1 Default setting = t#3 m
.tOffDelayFullLoad	TIME	Switch-off delay for full-load operation Default setting: t#30s
.tOnDelayPump	TIME	Switch-on delay for boiler circuit pump with one 2-way valve Default setting = t#2 m
.tOffDelayPump	TIME	Switch-off delay for boiler circuit pump Default setting = t#10 m
.tOffDelayAdmixing Pump	TIME	Switch-off delay for admixing pump Default setting = t#1 m
.rDiffBoilerAndReturn Temperature	REAL	Max. temperature difference between boiler and return temperature for de-activation of the boiler circuit pump [K] Default setting = 5
.rHysteresis	REAL	Hysteresis for the individual limit values [K] Default setting = 5
.xThreeWayValve	BOOL	3-way valve present in boiler circuit Default setting = TRUE

.xTwoWayValve	BOOL	2-way valve (boiler valve) present Default setting = FALSE
.xSmallWaterVolume	BOOL	Low volume of water in boiler Default setting = FALSE
.xBlockingProtection	BOOL	Enable blocking protection Default setting = TRUE
.xAdmixingPump	BOOL	Admixing pump present Default setting = FALSE
.xTwoWayValve PermanentOpen	BOOL	2-way valve closed in the event of a fault Default setting = FALSE
.tMaxRuntimeValve	TIME	Maximum runtime for 2-way valve Default setting: t#30s
.tPumpValveMaxOFF	TIME	Maximum turn-off time of the pump and valve up to activation of blocking protection Default setting = t#24h
.tPumpValveON	TIME	Maintenance run time for pump and valve Default setting: t#30s
.rKpBurner	REAL	Proportional gain (P portion) Default setting = 3
.rTnBurnerUp	REAL	Reset time of the controller for Actual > Specified [s] Default setting = 220
.rTnBurnerDown	REAL	Reset time of the controller for Actual > Specified [s] Default setting = 20
.rDeadZoneBurner	REAL	Dead zone +/- [K] Default setting = 1
.rMinPowerBurner	REAL	Minimum output of burner [%] Default setting = 10
.rMaxPowerBurner	REAL	Maximum output of burner [%] Default setting = 100
.rXpMaxBoiler Temperature	REAL	Proportional band for continuous limiting of boiler temperature Default setting = 10
.tOffDelayModulation	TIME	Off-delay for modulation Default setting: t#30s
.rLeadKpBurner	REAL	Proportional gain (P portion) as lead boiler Default setting = 3
.rLeadTnBurnerUp	REAL	Reset time of the controller for actual value < specified value as the lead boiler [s] Default setting = 120
.rLeadTnBurnerDown	REAL	Reset time of the controller for actual value > specified value as the lead boiler [s] Default setting = 20
.tLeadMinRuntime Level1	TIME	Minimum runtime for lead boiler at Level 1 Default setting = t#5 m
.rLeadHysteresis	REAL	Hysteresis as the lead boiler for the individual limit values Default setting = 5
.rLeadOffsetReference Temperature	REAL	Offset to the specified boiler temperature as lead boiler [K] Default setting = 5

xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Number of operating minutes of the boiler
Return value:	**Data type:**	**Comment:**
xEnableBurner	BOOL	Enable Burner
rY_Burner	REAL	Burner output Value range = 0 – 100
wY_Burner	WORD	Burner output Value range = 0 – 32767
xBoilerPump	BOOL	Switching signal for boiler circuit pump
xValve	BOOL	Switching signal for 2-way valve
rY_Valve	REAL	Valve position, 3-way valve [%] Value range = 0 – 100
wY_Valve	WORD	Valve position, 3-way valve Value range = 0 – 32767
xAdmixingPump	BOOL	Switching signal for the admixing pump
xChimneySweep	BOOL	Chimney sweep function active
xCondensationProtection	BOOL	Condensation protection
rMinBoilerTemperature	REAL	Minimum boiler temperature
xErrorBoilerPump	BOOL	Boiler circuit pump error
xErrorAdmixingPump	BOOL	Admixing pump error
xErrorStartUp	BOOL	Start-up system fault
typStatusBoiler	typStatus Boiler	Boiler status for the strategy module
wStatus	WORD	Display current status 0 = OK 2 = Off 10 = Error Pump 29 = Error Safety chain 30 = Condensation protection 31 = Chimney sweep function 32 = Overheating protection 33 = Error Valve 34 = Follow up time 35 = Manual operation 39 = Reference value exceeded 40 = Startup behaviour

Graphical illustration:

Visualization objects:	
ConfigModulatingBoiler	
ModulatingBoiler	

Function description:

The **FbModulatingBoiler** function contains various startup processes based on the pumps and valves used in the specific configuration and also regulates a modulating boiler.

Configuration parameters:

The configuration structure ***"typConfigModulatingBoiler"*** contains the following parameters:

- ***"rOffsetReferenceTemperature"*** and ***"rLeadOffsetReference Temperature"*** (parameter for lead boiler) defines the offset to the specified boiler temperature.
- ***".rMinBoilerTemperature"*** defines the minimum boiler temperature at which the boiler may be operated.
- ***".rMaxBoilerTemperatureLevel1"*** defines the maximum boiler temperature at which the boiler will be de-activated.
- ***".rMaxBoilerTemperatureModulating"*** defines the maximum boiler temperature at which switching back to Level 1 is initiated.
- ***".rXpMaxBoilerTemperature"*** defines the proportional band for continuous limiting of the maximum boiler temperature.
- ***".rMaxReturnTemperature"*** defines the maximum return temperature that results in the boiler being shut down.
- ***".rMinReturnTemperature"*** defines the minimum return temperature at which the boiler should be operated.
- ***".rKpMinReturnTemperature"*** defines the proportional gain for minimum return temperature control.
- ***".rTnMinReturnTemperature"*** defines the reset time for minimum return temperature control.
- ***".tMaxChimneySweepFunction"*** defines the maximum time period for the chimney sweep function.
- ***".tMaxStartUpMinBoilerTemperature"*** defines the maximum startup time for reaching the minimum boiler temperature.
- ***".tMaxFlushPeriod"*** defines the maximum startup time for reaching the minimum return temperature.
- ***".tMinRuntimeLevel1"*** and ***".tLeadMinRuntimeLevel1"*** (parameter for lead boiler) define the minimum runtime for the boiler at Level 1.
- ***".tOffDelayFullLoad"*** defines the follow-up time for the full-load mode.
- ***".tOnDelayPump"*** defines the On-delay for the pump with a series-connected boiler valve.
- ***".tOffDelayPump"*** defines the minimum follow-up time for the boiler circuit pump.
- ***".tOffDelayAdmixingPump"*** defines the follow-up time for the admixing pump.
- ***".rDiffBoilerAndReturnTemperature"*** defines the maximum temperature difference between the boiler temperature and the return temperature for shutting down the boiler pump.

- ***".rHysteresis"*** and ***".rLeadHysteresis"*** (parameter for the lead boiler) define the hysteresis for the respective limits.
- ***".xThreeWayValve"*** indicates for the startup circuit whether a 3-way valve is available for the admixing function.
- ***".xTwoWayValve"*** indicates whether a boiler valve is available.
- ***".xSmallWaterVolume"*** indicates for the startup circuit whether only a small volume of water is being routed through the boiler.
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".xAdmixingPump"*** indicates for the startup circuit whether an admixing pump is available for increasing the return temperature.
- ***".xTwoWayValvePermanentOpen"*** enables the boiler valve to be kept permanently open.
- ***".tMaxRuntimeValve"*** defines the maximum runtime for the boiler valve.
- ***".tPumpValveMaxOff"*** defines the maximum turn-off time for the blocking protection function.
- ***".tPumpValveOn"*** defines the runtime for the blocking protection function.
- ***".rKpBurner"*** and ***".rLeadKpBurner"*** define the proportional gain for burner control
- ***".rTnBurnerUp"*** and ***"rLeadTnBurnerUp"*** define the reset time for "runup" of burner output.
- ***".rTnBurnerDown"*** and ***"rLeadTnBurnerDown"*** define the reset time for "running down" burner output.
- ***".rDeadZoneBurner"*** defines the dead zone for burner control. If the actual value is located within the dead zone, the set value is not modified.
- ***".rMinPowerBurner"*** defines the minimum burner output
- ***".rMaxPowerBurner"*** defines the maximum burner output
- ***".tOffDelayModulation"*** defines the turn-off delay for the modulation stage

The boiler is activated either via the ***"xManual"*** input, or via the two inputs ***"xAuto"*** and ***"xSwitchOnBoiler"***.

When activated, the minimum boiler supply temperature is output for evaluation of the system supply temperature at the ***"rMinBoilerTemperature"*** output.

The specific boiler number ***"bBoilerNumber"*** and the number of the lead boiler ***"bLeadBoiler"*** determine whether the boiler is the lead or lag boiler. If both of these numbers are the same, the parameters for the lead boiler will be used.

Different startup procedures can apply, depending on the valve being used and the water volume:

2-way valve with large volume of water:

1.) Switch on the admixing pump ***"xAdmixingPump"***.

2.) Switch on the burner ***"xEnableBurner"*** (***"rY_Burner"*** = 0)

3.) The 2-way valve ***"xValve"*** is opened when the minimum boiler temperature is exceeded.

4.) If the boiler temperature fails to reach the minimum boiler temperature within a defined time, condensation protection "***xCondensationProtection"*** is activated and this indicated at the "wStatus" output.

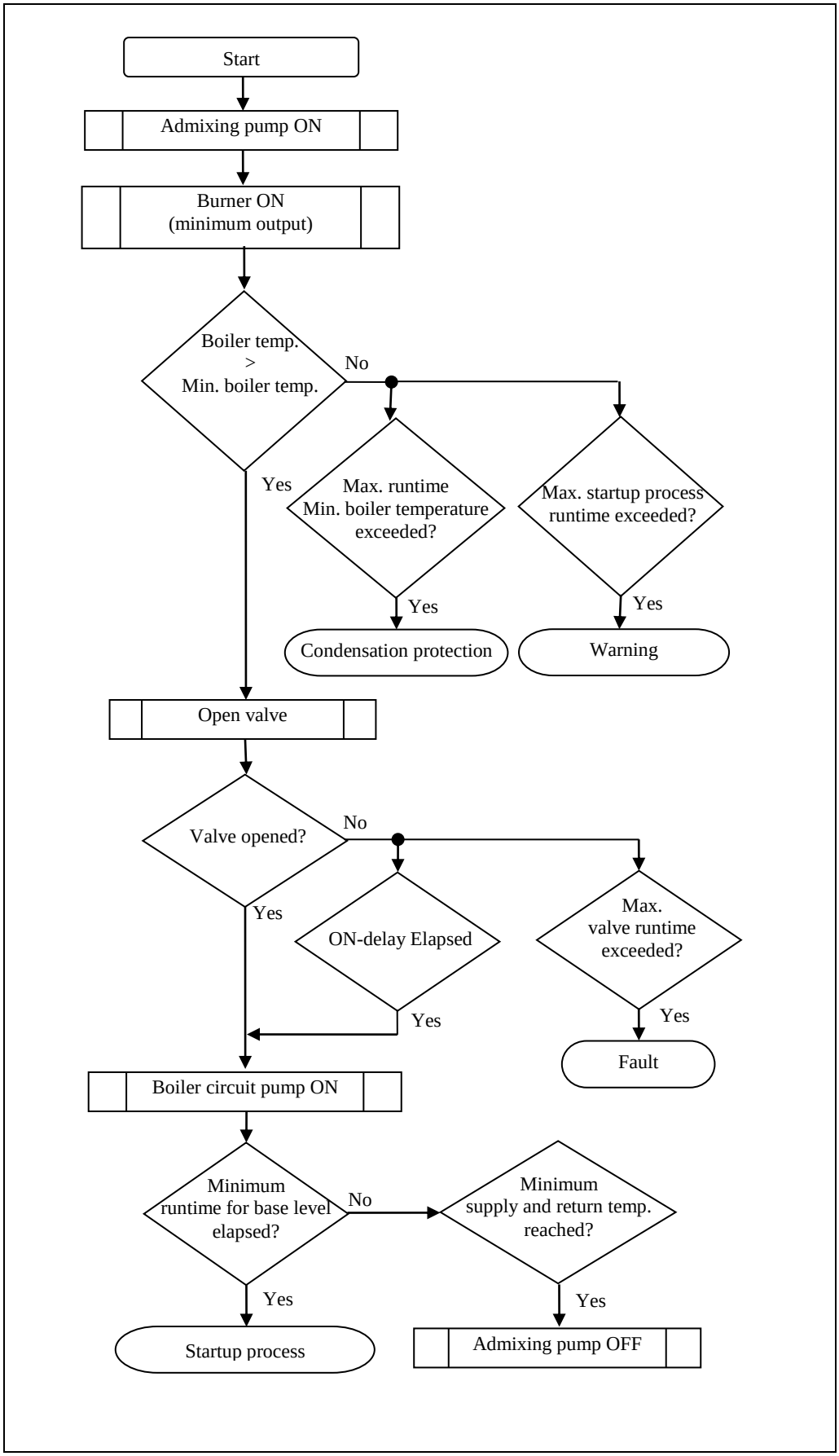
5.) The boiler circuit pump ***"xBoilerPump"*** is switched on when the open status of the valve is signaled via a positive edge at the ***"xLimitSwitchValve"*** input, or when the On-delay for the boiler circuit pump has elapsed.

6.) The minimum runtime at Level 1 begins when the boiler circuit pump is switched on. The startup procedure is terminated when the minimum runtime for Level 1 elapses.

7.) If the 2-way valve fails to reach its final position within the defined runtime, the boiler is switched off and an error message output at the *"wStatus"* output.

8.) If the startup procedure is not terminated within the configured time, the ***"xErrorStartUp"*** output is set and a warning issued via the *"wStatus"* output.

9.) The admixing pump is switched off as soon as the minimum return temperature and the minimum supply temperature are exceeded.


Start
Admixing pump ON
Burner ON
(minimum output)
Boiler temp.
>
Min. boiler temp.
No
Yes
Max. runtime
Min. boiler temperature
exceeded?
Max. startup process
runtime exceeded?
Yes
Yes
Condensation protection
Warning
Open valve
Valve opened?
No
Yes
ON-delay Elapsed
Max.
valve runtime
exceeded?
Yes
Yes
Fault
Boiler circuit pump ON
Minimum
runtime for base level
elapsed?
No
Minimum
supply and return temp.
reached?
Yes
Yes
Startup process
Admixing pump OFF

2-Way Valve with Low Water Volume

1.) Switch on the admixing pump *"xAdmixingPump"*.

2.) Open the 2-way valve *"xValve"*

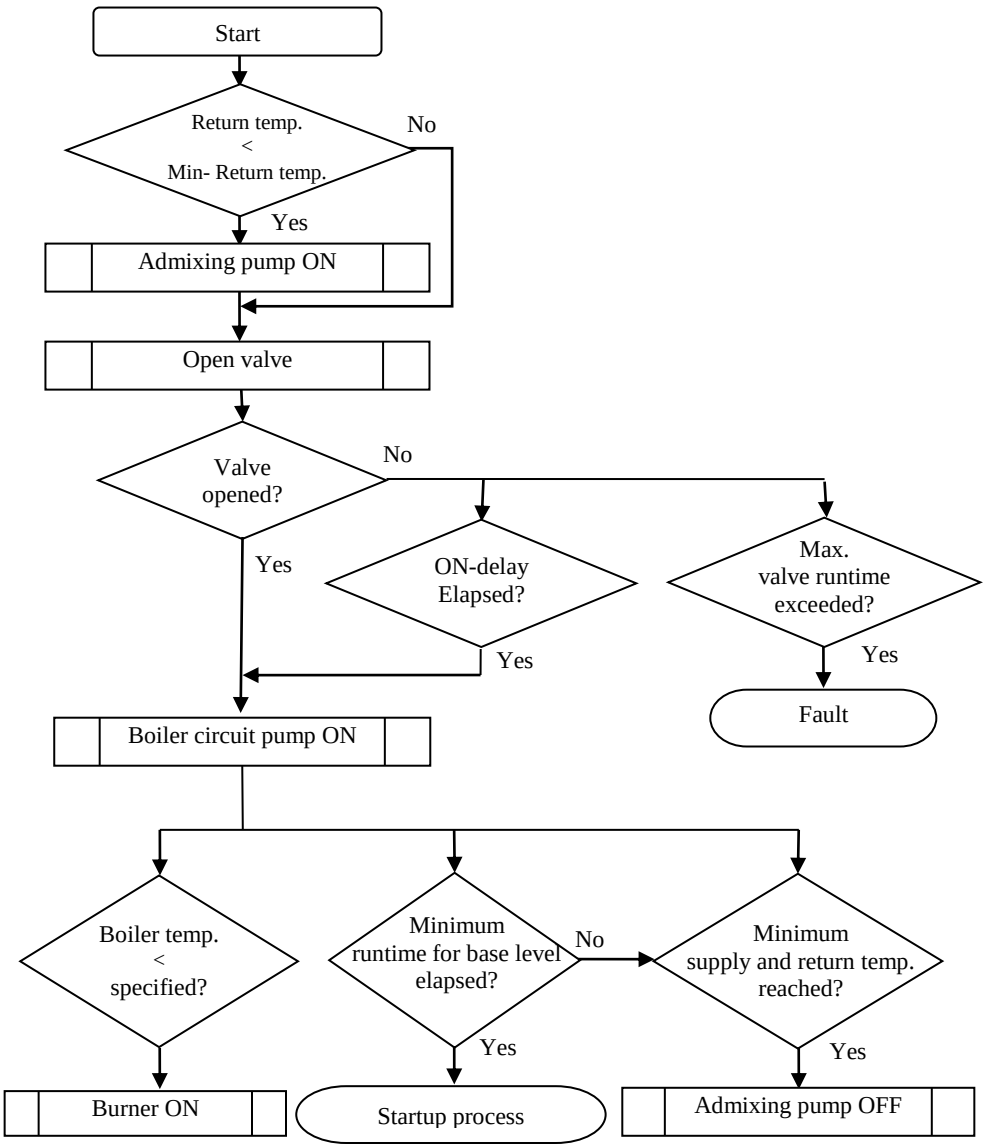
3.) The boiler circuit pump *"xBoilerPump"* is switched on when the On-delay for the pump has elapsed, or when a positive edge at the *"xLimitSwitchValve"* input reports the open status of the valve.

4.) The burner *"xEnableBurner"* is activated when the boiler temperature "rActualBoilerTemperature" is less than the specified boiler temperature "rReferenceBoilerTemperature".

5.) The startup procedure is terminated when the minimum runtime for Level 1 elapses.

6.) If the 2-way valve fails to reach its final position within the defined runtime, the boiler is switched off and an error message output at the *"wStatus"* output.

7.) The admixing pump is switched off as soon as the minimum return temperature and the minimum supply temperature are exceeded.



3-way valve

1.) Switch on the admixing pump *"xAdmixingPump"*.

2.) Switch on the boiler circuit pump *"xBoilerPump"*

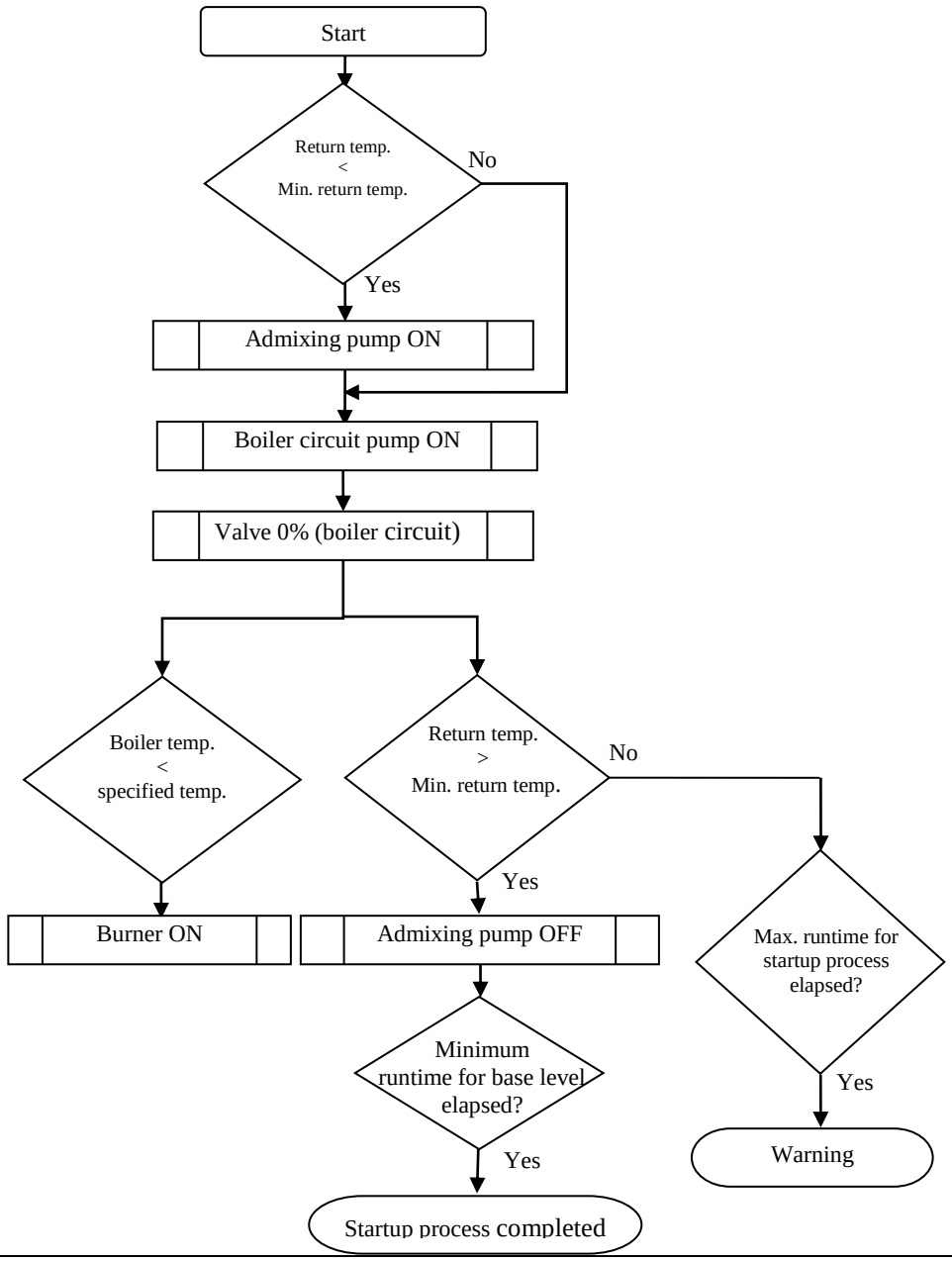
3.) Switch on the burner *"xEnableBurner"* (*"rY_Burner"* = 0)

4.) 3-way valve ***"rY_Valve"*** is closed (boiler circuit)

5.) The minimum runtime for Level 1 is started and the admixing pump switched off as soon as the return temperature exceeds the minimum return temperature.

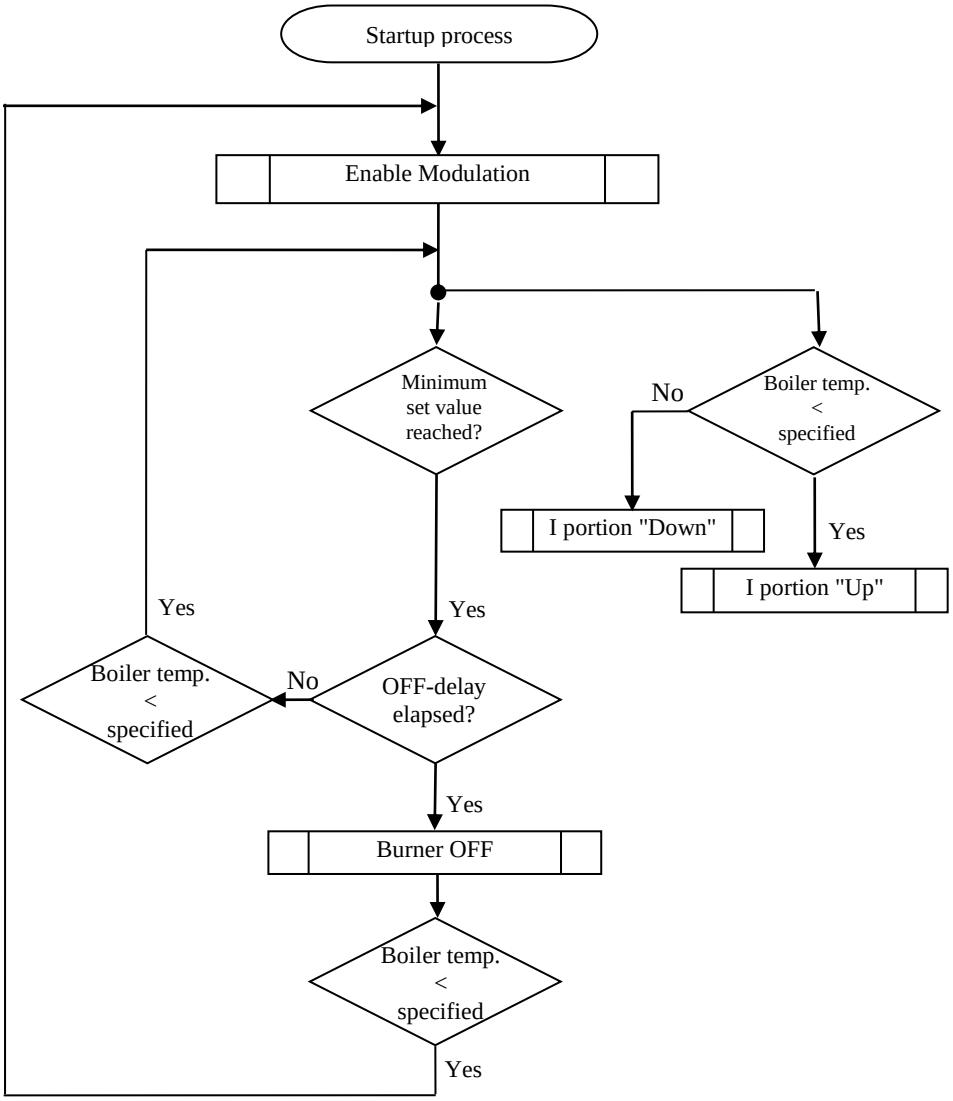
6.) The startup procedure is terminated when the minimum runtime for Level 1 elapses.

7.) If the minimum return temperature is not reached with the defined time, the ***"xErrorStartUp"*** output is set and a warning issued via the *"wStatus"* output.

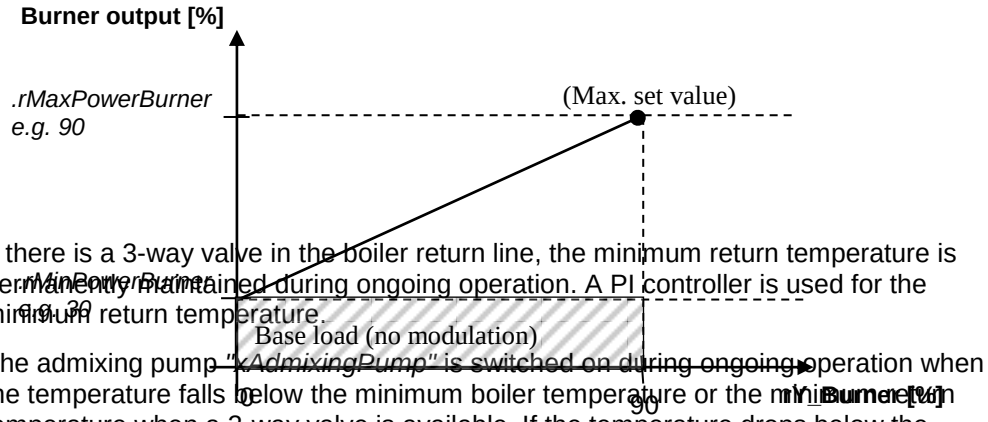


After the startup procedure, the burner is controlled via a PI controller. In this process, the reset time is switched as a function of the specified/actual value.
If the actual value is less than the specified value, the reset time for reducing burner output is applied. If the actual value is greater than the specified value, the reset time for increasing burner output is applied.

When the controller reaches its minimum set value, the boiler is switched off after a defined delay period. Modulation is started as soon as the boiler temperature falls below the specified value.



During modulation, boiler output is regulated to within the set minimum and maximum output levels via output *"rY_Burner"*, with the minimum output corresponding to the set value of 0%.



If there is a 3-way valve in the boiler return line, the minimum return temperature is permanently maintained during ongoing operation. A PI controller is used for the minimum return temperature.

The admixing pump *"xAdmixingPump"* is switched on during ongoing operation when the temperature falls below the minimum boiler temperature or the minimum return temperature when a 2-way valve is available. If the temperature drops below the minimum boiler temperature, the *"xCondensationProtection"* output is also set.

The boiler circuit pump continues to run when the boiler is switched off until the Switch-off delay time elapses and the difference between *"rActualBoilerTemperature"* and *"rActualReturnTemperature"* is less than the defined difference. The valve in the return line is not closed until the boiler circuit pump is switched off.

If the ***"xFullLoad"*** input is set using the strategy module, the boiler module no longer regulates the temperature in line with its specified boiler temperature, but is regulated only by maximum limiting. When the strategy module resets the *"xFullLoad"* input, the boiler remains in full-load operation for a defined time.

The necessary information about the boiler is supplied to the strategy module through the ***"typStatusBoiler"*** structure.

In addition to maximum limiting, the boiler module also provides continuous limiting of the boiler temperature.

The ***"xSafetyChain"*** input monitors the safety chain for the boiler. As soon as this input is switched to FALSE, the boiler is switched off and a corresponding error message indicated at the *"wStatus"* output.

In the event of a malfunction with the boiler circuit pump caused by the motor protection switch ***"xMotorProtectionPump"*** or the repair switch ***"xRepairSwitchPump"***, the boiler is switched off and the error indicated at the *"wStatus"* and ***"xErrorBoilerPump"*** output.

In the event of a malfunction of the admixing pump caused by the motor protection switch ***"xMotorProtectionAdmixingPump"*** or the repair switch ***"xRepairSwitchAdmixingPump"***, the admixing pump is switched off and the error indicated at the ***"xErrorAdmixingPump"*** output.

The error messages can be acknowledged via a flank at the ***"xQuit"*** input.

When the chimney sweep function *"xChimneySweepFunction"* is activated, the boiler switches on with an elevated reference value (maximum boiler temperature Modulation). The "***xChimneySweep"*** output is set as a check-back signal that the chimney sweep function has been activated. The chimney sweep function is canceled when the *"xChimneySweepFunction"* input is deactivated, or when the maximum runtime has elapsed.

If the boiler is switched to the Manual mode via an external circuit, a check-back signal should be transmitted to the boiler module via the ***"xFeedbackManualOperation"*** so that automatic control can be deactivated.

In the Manual mode ***"xManualOperation"*** the burner is switched on via ***"xManualOnBurner"*** and boiler output controlled via ***"rManualValueBoiler".*** The boiler circuit pump is switched on via ***"xManualOnBoilerPump"*** and the admixing pump via ***"xManualOnAdmixingPump"***. The valve is actuated via ***"rManualValueValve"***.

The pump or valve can be put through a maintenance run to prevent them from blocking during extended outage periods. Blocking protection must be activated for this.

The blocking protection function ensures that the pump and the valve do not remain switched off/closed longer than the specified monitoring period. On expiration of this time period, the pump and the valve are activated one after the other for the maintenance run for the defined time.

The output value ***"wY_Burner"*** has the same meaning as the *"rY_Burner"* output, except that the output has standardized values between 0 – 32767.

The output value ***"wY_Valve"*** has the same meaning as the *"rY_Valve"* output, except that the output has standardized values between 0 – 32767.

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

3.) The input *"xLimitSwitchValve"* must be set to TRUE when a 2-way valve without a limit switch is used.

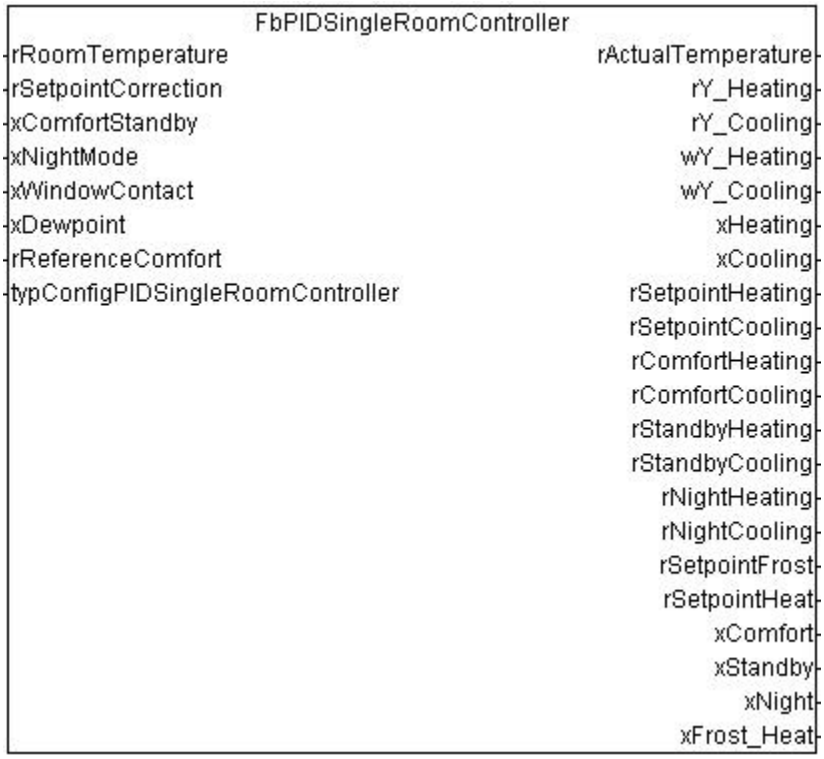
13 Single Room Control

PID Single Room Controller (FbPIDSingleRoomController)

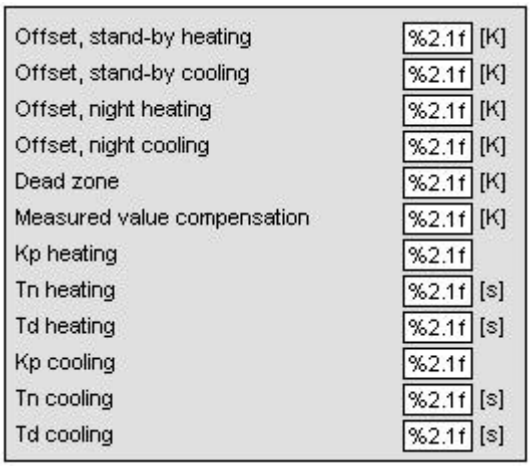
WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbPIDSingleRoomController	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
rRoomTemperature	REAL	Actual room temperature [°C]
rSetpointCorrection	REAL	Set value correction room temperature [K]
xComfortStandby	BOOL	Operating mode Comfort/Stand-by (1 / 0)
xNightMode	BOOL	Polling of operating mode "Night"
xWindowContact	BOOL	Window contact check-back signal for operating mode "Frost/Overheating protection“
xDewpoint	BOOL	Polling of operating mode “dew point alarm“
rReferenceComfort	REAL	Basic set value comfort mode [°C] Default setting = 21 °C
typConfigPIDSingleRoom Controller	←	Configuration parameters:
.rOffsetStandbyHeating	REAL	Stand-by temperature decrease [K] Default setting = 2 K
.rOffsetStandbyCooling	REAL	Stand-by temperature increase [K] Default setting = 2 K
.rOffsetNightHeating	REAL	Temperature decrease night [K] Default setting = 4 K
.rOffsetNightCooling	REAL	Temperature increase night [K] Default setting = 4 K
.rDeadZone	REAL	Dead zone between heating and cooling [K] Default setting = 2 K
.rOffset	REAL	Measured value compensation for room temperature input [K] Default setting = 0 K
.rKpHeating	REAL	Proportional gain, heating Default setting = 10
.rTnHeating	REAL	Reset time Tn heating [s] Default setting = 120 s
.rTdHeating	REAL	Derivative time Td heating [s] Default setting = 0 s

.rKpCooling	REAL	Proportional gain, cooling Default setting = 10
.rTnCooling	REAL	Reset time Tn cooling [s] Default setting = 120 s
.rTdCooling	REAL	Derivative time Td cooling [s] Default setting = 0 s
Return value:	**Data type:**	**Comment:**
rActualTemperature	REAL	Output of room temperature [°C]
rY_Heating	REAL	Set value heating valve Value range = 0 – 100
rY_Cooling	REAL	Set value cooling valve Value range = 0 – 100
wY_Heating	WORD	Set value heating valve Value range = 0 – 32767
wY_Cooling	WORD	Set value cooling valve Value range = 0 – 32767
xHeating	BOOL	Mode heating activated
xCooling	BOOL	Mode cooling activated
rSetpointHeating	REAL	Current set value for heating [°C]
rSetpointCooling	REAL	Current set value for cooling [°C]
rComfortHeating	REAL	Current set value for “comfort heating“ [°C]
rComfortCooling	REAL	Current set value for “comfort cooling“ [°C]
rStandbyHeating	REAL	Current set value for “standby heating“ [°C]
rStandbyCooling	REAL	Current set value for “standby cooling“ [°C]
rNightHeating	REAL	Current set value for “night heating“ [°C]
rNightCooling	REAL	Current set value for “night cooling“ [°C]
rSetpointFrost	REAL	Output of set value for frost protection [°C]
rSetpointHeat	REAL	Output of set value for heat protection [°C]
xComfort	BOOL	Display of operating mode "Comfort"
xStandby	BOOL	Display of operating mode "Stand-by"
xNight	BOOL	Display of operating mode "Night"
xFrost_Heat	BOOL	Display of operating mode “frost“

Graphical illustration:



Visualization objects:

ConfigPIDSingleRoom Controller	

Function description:

The **FbPIDSingleRoomController** function block allows individual room reference temperature control while taking local influences into account.

Configuration parameters:

The configuration structure ***"typConfigPIDSingleRoomController"*** contains the following parameters:

- ***".rOffsetStandbyHeating"*** defines the offset to the base reference value (heating) in the Stand-by mode.
- ***".rOffsetStandbyCooling"*** defines the offset to the base reference value (cooling) in the Stand-by mode.
- ***".rOffsetNightHeating"*** defines the offset to the reference value (heating) in the Stand-by mode during night-time temperature reduction.
- ***".rOffsetNightCooling"*** defines the offset to the reference value (cooling) in the Stand-by mode during night-time temperature reduction.
- ***".rDeadZone"*** defines the dead zone between heating and cooling. The selected size of this dead zone must not be too small in order to avoid a permanent changing back and forth between heating and cooling.
- ***".rOffset"*** enables measured value compensation for the room temperature sensor.
- ***".rKpHeating"*** defines the proportional gain of the controller for heating
- ***".rTnHeating"*** defines the reset time of the controller for heating
- ***".rTdHeating"*** defines the derivative time of the controller for heating
- ***".rKpCooling"*** defines the proportional gain of the controller for cooling
- ***".rTnCooling"*** defines the reset time of the controller for cooling
- ***".rTdCooling"*** defines the derivative time of the controller for cooling

The room temperature ***"rActualTemperature"*** is yielded from the measured room temperature ***"rRoomTemperature"*** and the variable measured value compensation.

The PID controller regulates the room temperature *"rActualTemperature"* to the defined reference value. Depending on the operating mode, the set value is given either at the ***"rY_Heating"*** or ***"rY_Cooling"*** output.

The output value ***"wY_Heating"*** has the same meaning as the *"rY_Heating"* output, except that the output has standardized values between 0 – 32767.

The output value ***"wY_Cooling"*** has the same meaning as the *"rY_Cooling"* output, except that the output has standardized values between 0 – 32767.

The controller detects four operating modes to each of which is assigned its own set value. The ***"rReferenceComfort"*** set value is used as a basic set value. All other set values refer to the basic set value and provoke each a set value increase or set value decrease by a parameterized value.

The reference value in the Comfort mode can be infinitely shifted via the ***"rSetpointCorrection"*** input.

The active operating mode (Comfort, Stand-by, Night, Antifreeze protection) is determined via the ***"xComfortStandby", "xNightMode"*** and ***"xWindowContact"*** inputs.

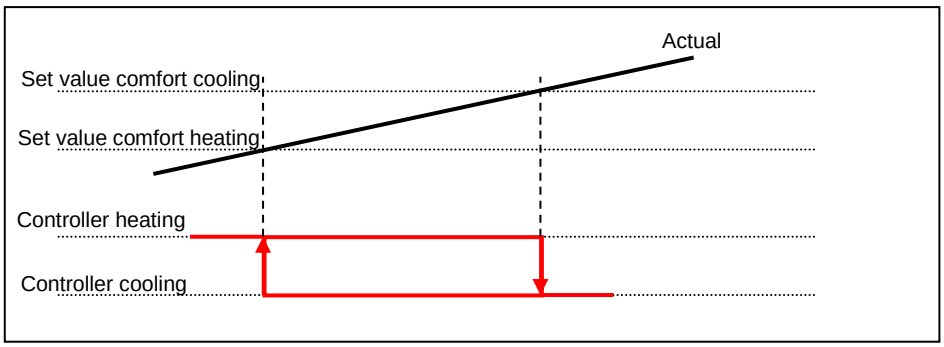
The currently selected operating mode is viasualized via ***"xComfort", "xStandby", "xNight"*** and ***"xFrost_Heat"***.

If the function module is used for cooling purposes, another *"xDewpoint"* input is required. If a dew point alarm is signalled on this input, the cooling / heating system switches off immediately.

The function block has ten monitor outputs for displaying the specified temperatures: ***"rSetpointHeating", "rSetpointCooling", "rComfortHeating", "rComfortCooling", "rStandbyHeating", "rStandbyCooling", "rNightHeating", "rNightCooling", "rSetpointFrost" and "rSetpointHeat".*** The current set values of the individual operating modes are put out via these outputs.

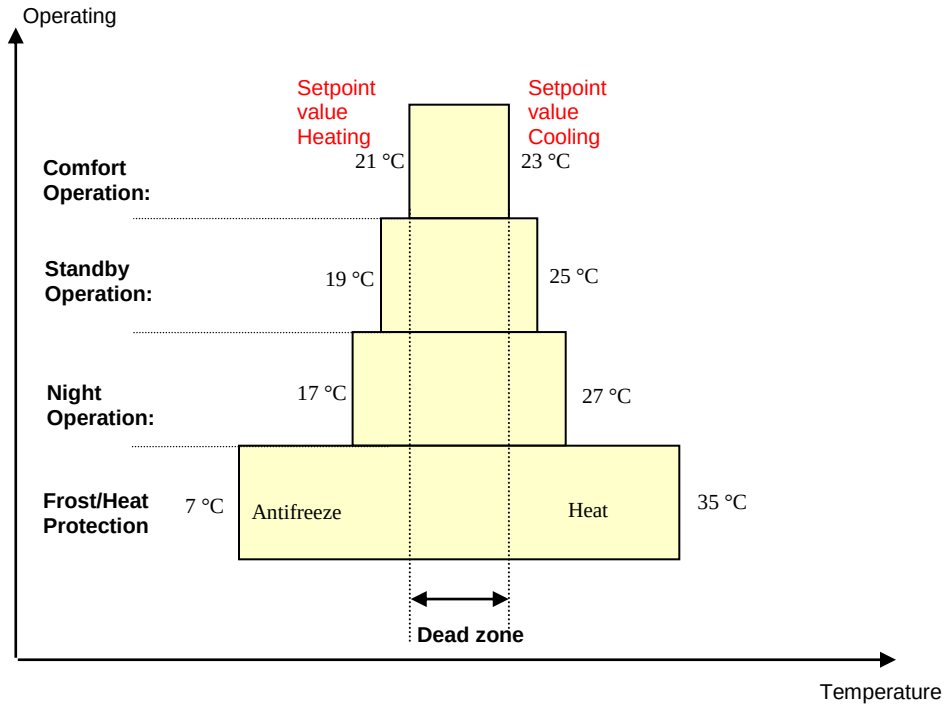
The outputs ***"xHeating"*** and ***"xCooling"*** show which mode (heating or cooling) is active. If the set value for heating and cooling is 0%, then the two outputs *"xHeating"* and *"xCooling"* have the signal "FALSE".

Switching between heating and cooling takes place automatically (see diagram below). The controller is either in the heating mode or in the cooling mode. The mode that is currently not active is switched to 0%.



Note:

The D part is set to zero with most of the room heating controllers because a PI controller has sufficient precision and is easier to set.



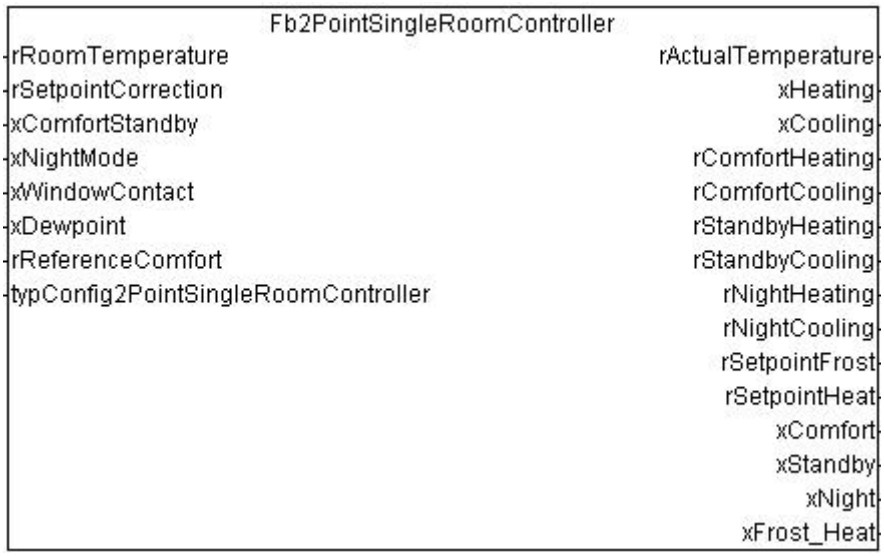
Operating status	Set value heating	Set value cooling
Comfort mode	**Basic set value 21 °C**	Basic set value + Dead zone 2 K
Standby mode	Basic set value - temperature decrease Standby mode	Basic set value Dead zone + temperature increase Standby mode
Night mode	Basic set value - Temperature decrease Night mode	Basic set value Dead zone + Temperature increase Night mode
Frost/heat protection	Setpoint value frost protection 7 °C	Setpoint value heat protection 35 °C
Dew point alarm	Setting value 0 (heating off)	Setting value 0 (cooling off)

2-Point Individual Room Controller (Fb2PointSingleRoomController)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	Fb2PointSingleRoomController	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
rRoomTemperature	REAL	Actual room temperature [°C]
rSetpointCorrection	REAL	Set value correction room temperature [K]
xComfortStandby	BOOL	Operating mode Comfort/Stand-by (1 / 0)
xNightMode	BOOL	Polling of operating mode "Night"
xWindowContact	BOOL	Window contact check-back signal for operating mode "Frost/Overheating protection“
xDewpoint	BOOL	Polling of operating mode “dew point alarm“
rReferenceComfort	REAL	Basic set value comfort mode [°C] Default setting = 21 °C
typConfig2PointSingle RoomController	←	Configuration parameters:
.rOffsetStandbyHeating	REAL	Stand-by temperature decrease [K] Default setting = 2 K
.rOffsetStandbyCooling	REAL	Stand-by temperature increase [K] Default setting = 2 K
.rOffsetNightHeating	REAL	Temperature decrease night [K] Default setting = 4 K
.rOffsetNightCooling	REAL	Temperature increase night [K] Default setting = 4 K
.rHysteresis	REAL	Deviation from set value [K] Default setting = 0.3 K
.rDeadZone	REAL	Dead zone between heating and cooling [K] Default setting = 2 K
.rOffset	REAL	Measured value compensation for room temperature input [K] Default setting = 0 K
Return value:	**Data type:**	**Comment:**
rActualTemperature	REAL	Current room temperature [°C]
xHeating	BOOL	Switching signal heating
xCooling	BOOL	Switching signal cooling
rComfortHeating	REAL	Current set value for "comfort heating"
rComfortCooling	REAL	Current set value for "comfort cooling"
rStandbyHeating	REAL	Current set value for "standby heating"
rStandbyCooling	REAL	Current set value for "standby cooling"
rNightHeating	REAL	Current set value for "night heating"

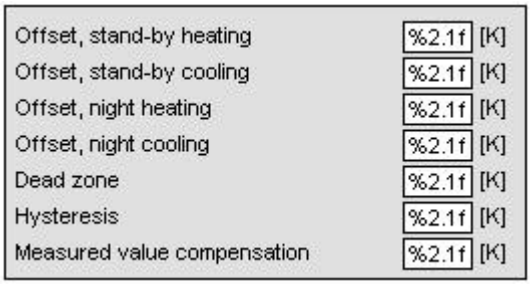
rNightCooling	REAL	Current set value for "night cooling"
rSetpointFrost	REAL	Output of set value for frost protection
rSetpointHeat	REAL	Output of set value for heat protection
xComfort	BOOL	Display of operating mode "Comfort"
xStandby	BOOL	Display of operating mode "Stand-by"
xNight	BOOL	Display of operating mode "Night"
xFrost_Heat	BOOL	Display of operating mode "Frost / Heat"

Graphical illustration:

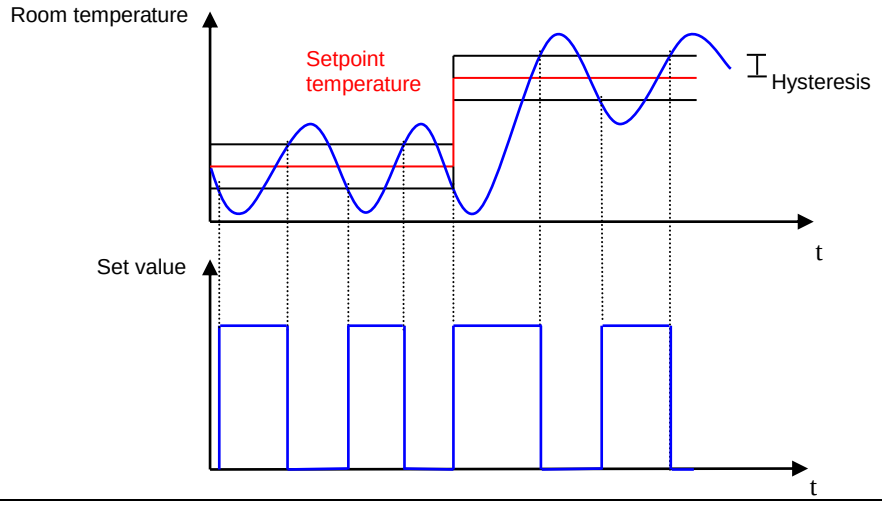


Visualization objects:

Config2PointSingleRoom Controller



Time referenced behavior:



Function description:

The **Fb2PointSingleRoomController** function block allows individual room reference temperature control while taking local influences into account.

Configuration parameters:

The configuration structure ***"typConfig2PointSingleRoomController"*** contains the following parameters:

- ***".rOffsetStandbyHeating"*** defines the offset to the base reference value (heating) in the Stand-by mode.
- ***".rOffsetStandbyCooling"*** defines the offset to the base reference value (cooling) in the Stand-by mode.
- ***".rOffsetNightHeating"*** defines the offset to the reference value (heating) in the Standby mode during night-time temperature reduction.
- ***".rOffsetNightCooling"*** defines the offset to the reference value (cooling) in the Standby mode during night-time temperature reduction.
- ***".rDeadZone"*** defines the dead zone between heating and cooling. The selected size of this dead zone must not be too small in order to avoid a permanent changing back and forth between heating and cooling.
- ***".rHysteresis"*** defines the switching hysteresis for the 2-point controller. A small hysteresis provokes a frequent switching of the valve voltage, but small set value differences. A large hysteresis causes large deviations from the set value, but only leads to occasional switching.
- ***".rOffset"*** enables measured value compensation for the room temperature sensor.

The room temperature ***"rActualTemperature"*** is yielded from the measured room temperature ***"rRoomTemperature"*** and the variable measured value compensation.

The 2-point controller compares the room temperature *"rActualTemperature"* (actual value) with the desired heating and cooling reference values and sends the corresponding switching telegrams for heating ***"xHeating"*** and cooling ***"xCooling"***.

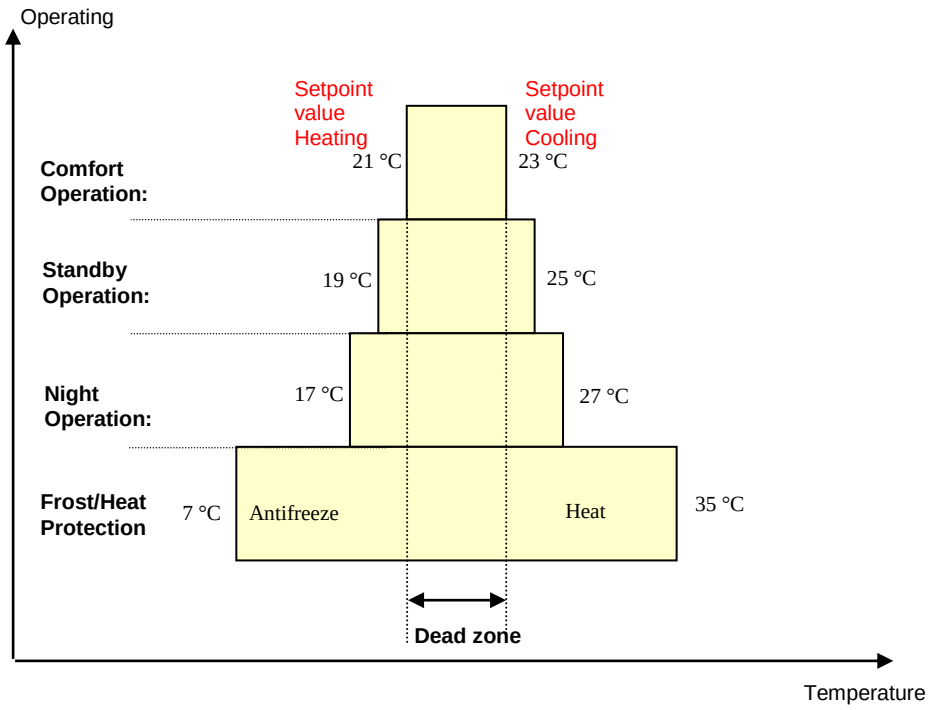
The controller detects four operating modes to each of which is assigned its own set value. The ***"rReferenceComfort"*** set value is used as a basic set value. All other set values refer to the basic set value and provoke each a set value increase or set value decrease by a parameterized value.

The reference value for the Comfort mode can be infinitely shifted via the ***"rSetpointCorrection"*** input. The active operating mode (Comfort, Stand-by, Night, Antifreeze protection) is determined via the ***"xComfortStandby", "xNightMode"*** and ***"xWindowContact"*** inputs.

The currently selected operating mode is viasualized via ***"xComfort", "xStandby", "xNight"*** and ***"xFrost_Heat"***.

If the function module is used for cooling purposes, another "xDewpoint" input is required.If a dew point alarm is signaled at this input, the cooling/heating valves close accordingly.

The function module has eight monitor outputs "rComfortHeating", "rComfortCooling", "rStandbyHeating", "rStandbyCooling", "rNightHeating", "rNightCooling", "rSetpointFrost" and "rSetpointHeat". The current set values of the individual operating modes are put out via these outputs.



Operating status	Set value heating	Set value cooling
Comfort mode	**Basic set value 21 °C**	Basic set value + Dead zone 2 K
Standby mode	Basic set value - Temperature decrease Standby mode	Basic set value Dead zone + Temperature increase Standby mode
Night mode	Basic set value - Temperature decrease Night mode	Basic set value Dead zone + Temperature increase Night mode
Frost/heat protection	Setpoint value frost protection 7 °C	Setpoint value heat protection 35 °C
Dew point alarm	Setting value 0 (heating off)	Setting value 0 (cooling off)

WAGO Kontakttechnik GmbH & Co. KG
P:O. box 2880 • D-32385 Minden
Hansastr. 27 • D-32423 Minden
Phone: 05 71 / 8 87-0
Fax.: +49 (0) 5 71 / 8 87-169
E-Mail: info@wago.com
Web: http://www.wago.com

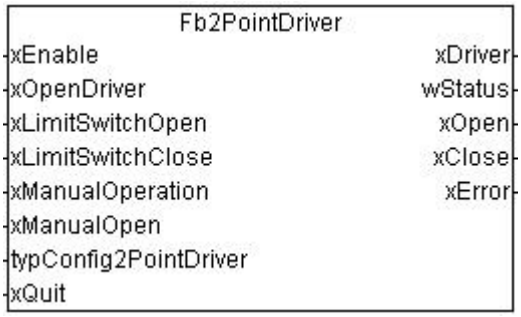
14 General Drivers

Control of 2-Point Drivers (Fb2PointDriver)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	Fb2PointDriver	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable 2-point control Default setting = TRUE
xOpenDriver	BOOL	Move driver to "open" position Default setting = TRUE
xLimitSwitchOpen	BOOL	Check-back signal from limit switch (open)
xLimitSwitchClose	BOOL	Check-back signal from limit switch (closed)
xManualOperation	BOOL	Enable manual operation
xManualOpen	BOOL	Open or close manually in the Manual mode Open = TRUE
typConfig2PointDriver	←	Configuration parameters:
.tMaxRuntime	TIME	Maximum runtime of the driver Default setting: t#30s
.tMaxOff	TIME	Maximum turn-off time for driver when blocking protection is activated Default setting = t#48h
.tSwitchOn	TIME	Turn-on time of driver when blocking protection is enabled Default setting: t#60s
.xLimitSwitchOpen	BOOL	Limit switch driver open available Default setting = TRUE
.xLimitSwitchClose	BOOL	Limit switch driver closed available Default setting = FALSE
.xBlockingProtection	BOOL	Activate blocking protection Default setting = TRUE
xQuit	BOOL	Error message acknowledgement
Return value:	**Data type:**	**Comment:**
xDriver	BOOL	Driver control

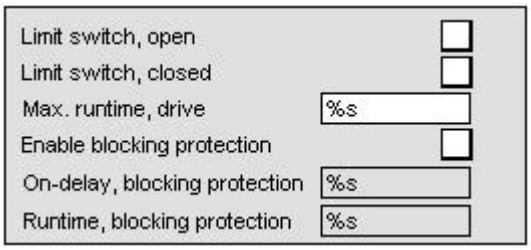
wStatus	WORD	Display current status 0 = OK 3 = Open 4 = Closed 36 = In motion 46 = Error limit switch
xOpen	BOOL	Driver open
xClose	BOOL	Driver closed
xError	BOOL	Driver error

Graphical illustration:



Visualization objects:

Config2PointDriver



Function description:

The **Fb2PointDriver** function block is used to control 2-point drivers with optional limit switches.

Configuration parameters:

The configuration structure ***"typConfig2PointDriver"*** contains the following parameters:

- ***".tMaxRuntime"*** monitors the maximum runtime for the driver when limit switches are provided. If no limit switches are present, this parameter is used for the runtime of the driver.
- ***".xLimitSwitchOpen"*** indicates whether a limit switch is available for "Driver open".
- ***".xLimitSwitchClose"*** indicates whether a limit switch is available for "Driver closed".
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".tMaxOff"*** defines the maximum turn-off time until the blocking protection function is started.
- ***".tSwitchOn"*** defines the runtime for blocking protection.

The driver is opened in the Automatic mode when the system has been enabled via ***"xEnable"*** and the ***"xOpenDriver"*** input has been activated.

When the Manual mode is activated via the ***"xManualOperation"***, the driver is controlled via the ***"xManualOpen"*** input.

The driver is controlled via the ***"rDriver"*** output.

The runtime of the driver is monitored when limit switches are provided for each direction of movement. When the maximum runtime is exceeded, the driver is closed and the ***"xError"*** output activated.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

In order to avoid blocking of the driver after extended outage periods, the driver can be put into operation at least once within a certain period of time. The blocking protection function must be activated for this.

The ***"xOpen"*** and ***"xClose"*** outputs indicate the status of the driver (opened/closed).

The current status for the driver is output via the ***"wStatus"*** output.

Note:

1.) If no limit switch is provided, the driver position is determined over time.

2.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

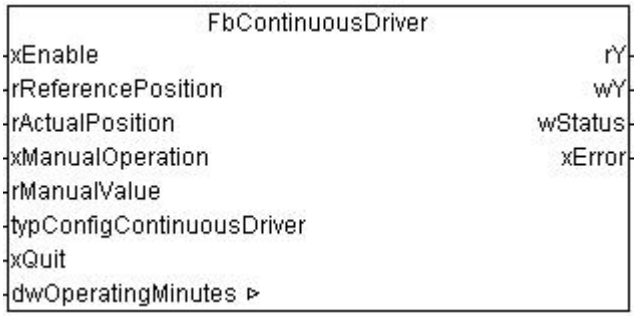
3.) Blocking protection can also be activated by a timer program, so that a potential driver error message is issued only during a defined time period.

Control of Continuous Drivers (FbContinuousDriver)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbContinuousDriver	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable control Default setting = TRUE
rReferencePosition	REAL	Specified position of continuous driver [%]
rActualPosition	REAL	Actual position of continuous driver [%]
xManualOperation	BOOL	Enable manual operation
rManualValue	REAL	Set value manual operation [%] Value range = 0 – 100
typConfigContinuous Driver	←	Configuration parameters:
.tOnDelayAlarm	TIME	ON-delay for alarm Default setting = t#5 m
.rMinDeviation Movement	REAL	Minimum deviation for detection of motion [%] Default setting = 5
.rTolerance	REAL	Max. deviation, specified/actual position [%] Default setting = 3
.rY_Min	REAL	Min. setting value for driver [%] Default setting = 3
.tMaxOff	TIME	Maximum turn-off time for driver when blocking protection is activated Default setting = t#48h
.tSwitchOn	TIME	Turn-on time of driver when blocking protection is enabled Default setting: t#60s
.rY_BlockingProtection	REAL	Specified position for blocking protection [%] Value range = 0 – 100 Default setting = 100
.xBlockingProtection	BOOL	Enable blocking protection Default setting = TRUE
.xFeedbackDevice	BOOL	Check-back signal, drive position detected Default setting = FALSE
xQuit	BOOL	Error message acknowledgement
Input/output parameters:	**Data type:**	**Comment:**
dwOperatingMinutes	DWORD	Operating minutes for continuous driver

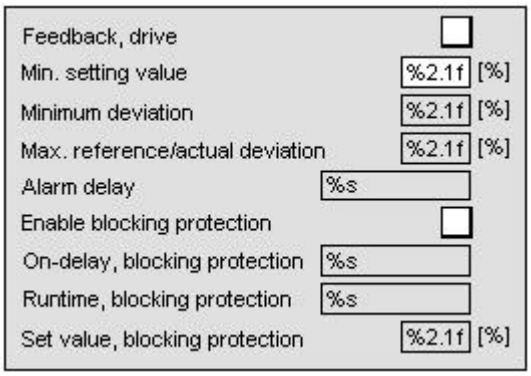
Return value:	Data type:	Comment:
rY	REAL	Set value, driver Value range = 0 – 100
wY	WORD	Set value, driver Value range = 0 – 32767
wStatus	WORD	Display current status 0 = OK 1 = On 2 = Off 36 = In motion 37 = Error
xError	BOOL	Driver error

Graphical illustration:



Visualization objects:

ConfigContinuousDriver



Function description:

The **FbContinuousDriver** function block is used for controlling continuous drivers. A driver position can also be monitored as an option.

<u>Configuration parameters:</u>

The configuration structure ***"typConfigContinuousDriver"*** contains the following parameters:

- ***".tOnDelayAlarm"*** defines the time period that can elapse until an alarm is issued for a permanent deviation of the driver position.
- ***".rMinDeviationMovement"*** defines the minimum deviation between the specified position ***"rReferencePosition"*** and the actual position ***"rActualPosition"*** for movement detection.

- ***".rTolerance"*** defines the permissible deviation between specified/actual position for position monitoring
- ***".rY_Min"*** defines the set value that must at least be reached to change the driver position.
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".tMaxOff"*** defines the maximum turn-off time until the blocking protection function is started.
- ***".tSwitchOn"*** defines the runtime for blocking protection.
- ***".rY_BlockingProtection"*** defines the specified position during the blocking protection function.
- ***".xFeedbackDevice"*** specifies whether a continuous check-back signal is present from the driver.

The driver control is enabled via the ***"xEnable"*** input.

When the Manual mode is activated via the input ***"xManualOperation"***, the driver is controlled via the ***"rManualValue"*** input.

The driver is controlled via the ***"rY"*** output.

The output value ***"wY"*** has the same meaning as the *"rY"* output, the output just has the standardized values between 0 – 32767.

When the position check-back signal is present with a permanent position deviation, the driver is closed and the ***"xError"*** is activated when the delay period is exceeded.

The error message can be acknowledged via a flank at the ***"xQuit"*** input and the function block is enabled again.

In order to avoid blocking of the driver after extended outage periods, the driver can be put into operation at least once within a certain period of time. The blocking protection function must be activated for this. The driver is moved to a settable position during the blocking protection function period.

The current status for the driver is output via the ***"wStatus"*** output.

The input/output variable ***"dwOperatingMinutes"*** indicates the operating minutes for the continuous driver. The operating minutes are counted when *"rY" is greater than ".rY_Min".*

Note:

1.) The **FuStatus** function converts the "*wStatus"* status message into a plain text message.

2.) The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

3.) Blocking protection can also be activated by a timer program, so that a potential error is issued only during a defined time period.

PWM Output (FbPWM)

WAGO-I/O-*PRO* Library Elements			
Category:	Building Automation		
Name:	FbPWM		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		

Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Activates calculation of the PWM signal
rY	REAL	(Valve) set value from the controller [%] Value range = 0 – 100
typConfigPWM	←	Configuration parameters:
.tCycleDuration	TIME	Cycle duration of the pulse width modulation Default setting = 10 min
.tMinTurnOnTime	TIME	Minimum power-on time of the pulsed digital output Default setting = 60 s

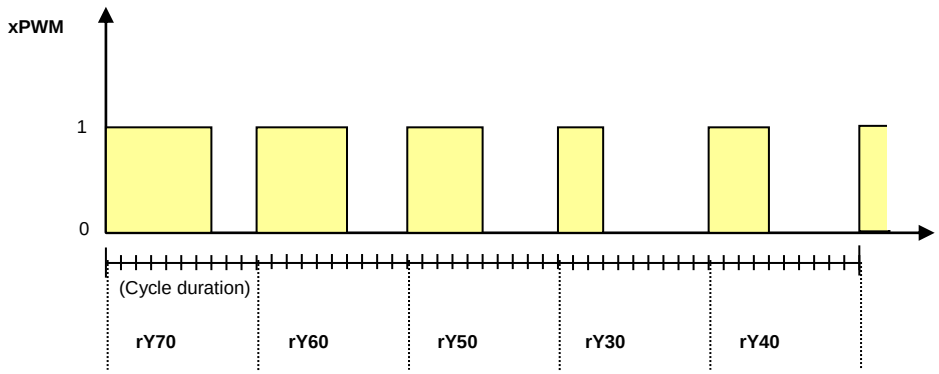
Return value:	**Data type:**	**Comment:**
xPWM	BOOL	Pulse width modulated output signal
rY_PWM	REAL	Display set value Value range = 0 – 100

Graphical illustration:

Visualization objects:

ConfigPWM

Time referenced behavior:



Function description:

FbPWM generates a pulse-width modulated output signal from a percentage set value.

Configuration parameters:

The configuration structure ***"typConfigPWM"*** contains the following parameters:

- ***".tCycleDuration"*** defines the time period over which the PWM signal is calculated.
- ***".tMinTurnOnTime"*** defines the smallest activation time for the PWM signal. The shortest time should be between 1 % and 50 % of the period duration. If the calculated activation period is less than the minimum activation time, then the *"xPWM"* output remains deactivated.

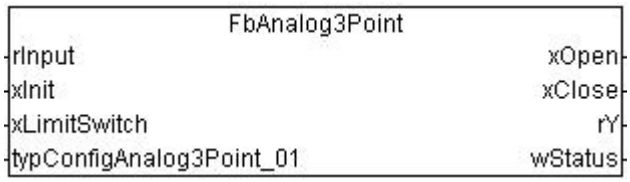
When the ***"xEnable"*** input is activated, the PWM signal is calculated from the ***"rY"*** input variable and the signal output at the ***"xPWM"*** output.

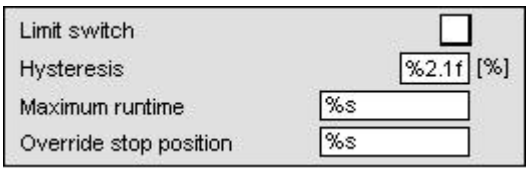
The *"xPWM"* output is deactivated as soon as the *"xEnable"* input is deactivated. A new cycle duration begins when the PW signal is enabled again.

The **FbPWM** function block works "dynamically" to achieve quicker response times. The activation period for the digital output signal is calculated continuously. Thus, the switching times are also adjusted during the active periods.

Analog 3-Point Signal (FbAnalog3Point)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbAnalog3Point	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
rInput	REAL	Input value [%] Value range = 0 – 100
xInit	BOOL	A positive flank starts the reference run
xLimitSwitch	BOOL	Limit switch (opend, closed)
typConfigAnalog3Point_01	←	Configuration parameters:
.rHysteresis	REAL	Hysteresis Value range = 1 – 100 Default setting = 1
.tMaxRunTime	TIME	Max. running time, which the actuator needs for the max. stroke Default setting: t#120s
.tOverride	TIME	Run-on time after reaching end positions
.xLimitSwitch	BOOL	Monitoring of the limit switches Default setting = FALSE
Return value:	**Data type:**	**Comment:**
xOpen	BOOL	Engine coil OPEN
xClose	BOOL	Engine coil CLOSED
rY	REAL	Calculated set value
wStatus	WORD	Display current status 0 = OK 46 = Error limit switch 47 = Synchronisation
Graphical illustration:		



Visualization objects:	
ConfigAnalog3Point	

Function description:

The **FbAnalog3Point** function block converts an analog set value into a 3-point signal. The actuating drive has the status OFF, ON and CLOSED. The setting values are calculated dynamically for this.

Configuration parameters:

The configuration structure ***"typConfigAnalog3Point"*** contains the following parameters:

- ***".rHysteresis"*** defines the switching hysteresis for the 3-point signal. The hysteresis ensures that the motor does not execute a full cycle on small changes in the input variable.
- ***".tMaxRunTime"*** defines the maximum runtime for the actuating drive.
- ***".tOverride"*** defines the time that the output is overriden when the calculated end position is reached. (Override used for position synchronization)
- ***".xLimitSwitch"*** indicates whether the limit switches for "opened" and "closed" are to be monitored.

The input value ***"rInput"*** is converted into a running time for the control valve.

The engine position is stored within the module and is displayed at the output ***"rY"***.

If the value at the *"rInput"* input differs from the output value *"rY"* by the set hysteresis, the driver is actuated via the ***"xOpen"*** and ***"xClose"*** in accordance with the given sign for the difference.

The check-back signal from the limit switches can be linked via an OR element to the ***"xLimitSwitch"*** input.

A limit switch error is issued at the ***"wStatus"*** output under the following conditions when the monitoring function is activated:

1.) When *"rY"* is situated between 10% and 90% and the *"xLimitSwitch"* input is TRUE

2.) When the override time elapses at "rY" 0% or 100% and the *"xLimitSwitch"* input is FALSE

The error message is reset only by a synchronization run. A synchronization run is performed either by **starting the program**, or by a positive edge at the ***"xInit"*** input. During the synchronizatin run, the actuator is closed for the set maximum runtime, plus the override period and the setting value re-referenced. The synchronization run is indicated at the *"wStatus"* output.

The position of the motor is determined using a timing element. Therefore, a synchronization is performed each time an end position is reached.

Note:

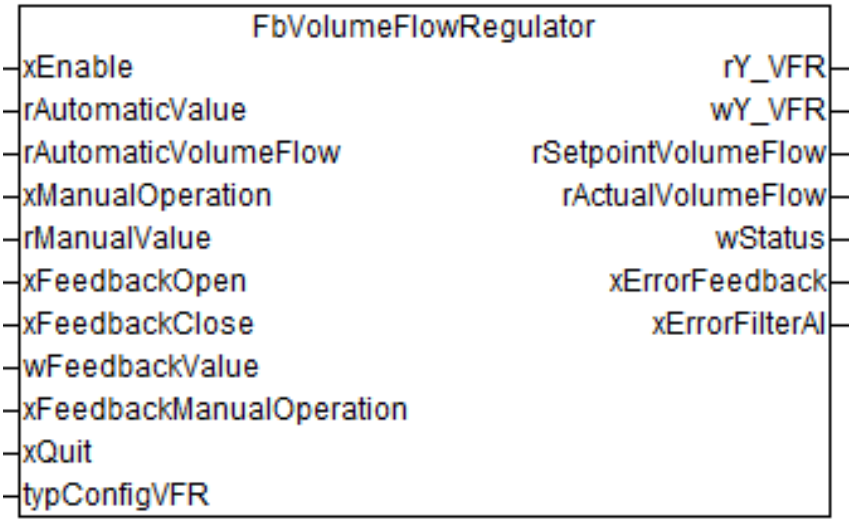
- The **FuStatus** function converts the "*wStatus"* status message into a plain text message.
- Actuation can be continued even after the driver linked to the system has reached its end position by overriding the driver. It should be clarified beforehand with the valve manufacturer, whether this status has no negative effect on the valve. We recommend control valves with built-in limit switches.

15 Volume Flow Regulator

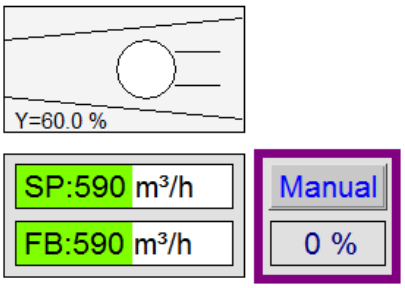
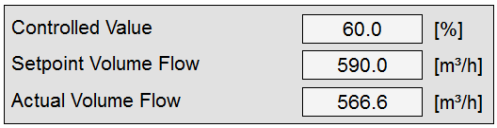
Volume Flow Regulator (FbVolumeFlowRegulator)

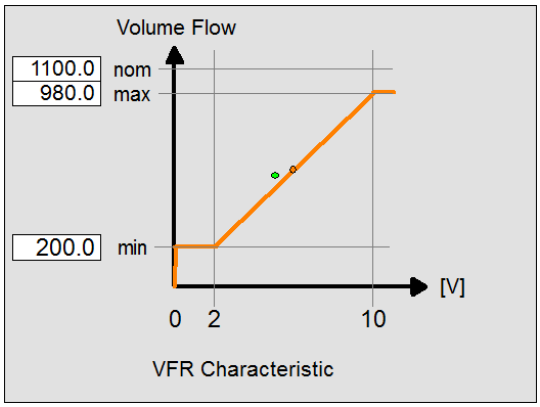
WAGO-I/O-*PRO*-Library Elements		
Category:	Building technology	
Name:	FbVolumeFlowRegulator	
Type:	Function ☐	Function block X ☐ Program ☐
Name of Library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input Parameter:	**Data Type:**	**Comment:**
xEnable	BOOL	Function block enable
rAutomaticValue	REAL	Automatic value [0 ... 100%]
rAutomaticVolumeFlow	REAL	Automatic value [0 ... Vmax]; BIT4 of the configuration must be activated for this input to be used.
xManualOperation	BOOL	Activate manual mode.
rManualValue	REAL	Manual value [0 ... 100%]
xFeedbackOpen	BOOL	Feedback open
xFeedbackClose	BOOL	Feedback closed
wFeedbackValue	WORD	Volume flow feedback to analog input
xFeedbackManualOperation	BOOL	Local manual intervention feedback (e.g., control cabinet)
xQuit	BOOL	Confirmation input for filter module
typConfigVFR	STRUCT	Configuration structure
Return Value:	**Data Type:**	**Comment:**
rY_VFR	REAL	Actuation signal [0 ... 100 %]
wY_VFR	WORD	Actuation signal [0 ... 32767]
rSetpointVolumeFlow	REAL	Volume flow setpoint [m³/h]
rActualVolumeFlow	REAL	Calculated actual volume flow [m³/h]
wStatus	WORD	Display of current status 0 = OK 3 = Open 4 = Closed 35 = Manual mode 36 = In motion 37 = Error (filter module) 42 = Feedback error
xErrorFeedback	BOOL	Feedback monitoring error
xErrorFilterAI	BOOL	Internal filter module error

Graphical illustration:



Visualization Objects:

VFR	
VFR_Values	
VFR_Error	

VFR_Characteristic	
VFR_Config	Enable Input "rAutomaticVolumeFlow" ☐ Enable Runtime Monitoring ☐ Enable Binary Feedback ☐ Enable Analogeous Feedback ☑ Enable 2-10V Control ☑ Min. Volume Flow 200.0 [m³/h] Max. Volume Flow 980.0 [m³/h] Nominal Volume Flow 1100.0 [m³/h] Min. Signal for Opening 0.0 [%] Feedback Tolerance 100.0 [m³/h] Feedback Monitoring T#4m0s0ms Filter configuration Scan rate T#100ms Time constant T#2s0ms Min. output value 0.0 Max. output value 100.0 Offset, output value 0.0 Default value 20 Lower alarm limit -32767 Upper alarm limit 32768 Alarm delay T#10s0ms Automatic error acknowledgement ☑
Function description:	

The function block **FbVolumeFlowRegulator** is used to actuate motorized volume flow controllers with optional position feedback and optional end position switch messages.

If the block is activated at input ***xEnable***, the setpoint [0 … 100 %] is specified via the input variable ***rAutomaticValue***. As an alternative, the setpoint can be directly specified as the volume flow [m³/h] within the parameterized limits, via the variable ***rAutomaticVolumeFlow***. This requires setting BIT0 in the variable ***wConfig*** (in the configuration structure).

The function block can be set into manual mode independently of input ***xEnable***. This requires setting the input variable ***xManualOperation*** to the value TRUE. The setpoint is then specified via the variable ***rManualValue***, within the value range 0 … 100 %.

If the volume flow controller to be controlled has end positions OPEN and CLOSED; these messages are connected with the inputs ***xFeedbackOpen*** or ***xFeedbackClosed***. The physical feedback must also be activated, by setting BIT2 in the variable ***wConfig*** (in the configuration structure). If there are no end positions, the function block emulates them internally as follows:

OPEN = rActualVolumeFlow ≥ (typConfigVFR.rVolumeFlow_Nominal – typConfigVFR.rFeedbackTolerance)

CLOSED = rActualVolumeFlow ≤ MAX(typConfigVFR.rVolumeFlow_Min, typConfigVFR.rFeedbackTolerance

The signal feedback of the current volume flow is sent to the function block via the input ***wFeedbackValue***. This allows the I/O module analog value to be immediately linked. The signal is internally sent via a function block for internal smoothing and scaling; the block's parameters are stored in the configuration structure. Typically, a volume flow controller delivers a feedback signal of 0 … 10 V, which is scaled in the range 0 … $V_{nominal}$. Then the signal is scaled back again within the function block within the limits ***typConfigVFR. typConfigFilterAI.rMin*** and ***typConfigVFR. typConfigFilterAI.rMax*** and read out at the output ***rActualVolumeFlow***. The analog feedback is activated by setting BIT3 in the variable ***wConfig*** (in the configuration structure). If the parameterized filter limit values (***typConfigVFR .typConfigLowPassFilterAI.rLowLimitAlarm***/***typConfigVFR .typConfigLowPassFilterAI.rHighLimitAlarm***) are infringed, an alarm is issued at the output ***xErrorFilterAI***. If no feedback signal is present, the variable ***rActualVolumeFlow*** is described by the volume flow setpoint (***rSetpointVolumeFlow***).

If the volume flow controller is overruled by a local manual intervention (e.g., control cabinet), the message is connected with the input ***xFeedbackManualOperation***. The automatic signal (***rAutomaticValue***/***rAutomaticVolumeFlow***) is then suppressed.

The volume flow controller is actuated via the output variables ***rY_VFR*** (0 … 100) and ***wY_VFR*** (0 … 32767). The currently calculated volume flow setpoint is read out at the output variable ***rSetpointVolumeFlow***.

The current status of the function block is read out via the output variable ***wStatus***.

Note:

The function **FuStatus** converts the status message "*wStatus*" into a plain text message.

Example:
3 supply pumps
1 pump is required.
Switch according to operating hours.
Automatic fault changeover
10 circulation cooling units
Request for the number of units by cooling load (1 ... 6)
Switch according to operating hours during operation.
Automatic fault changeover

16 Additional Functions

Blinker (FbBlinker)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbBlinker	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable Blinker
typConfigBlinker	←	Configuration parameters:
.tTimeHigh	TIME	Turn-on time for the blinker Default = t#500ms
.tTimeLow	TIME	Turn-off time for blinker Default = t#500ms
Return value:	**Data type:**	**Comment:**
xOutput	BOOL	Blink signal
Graphical illustration:		
FbBlinker xEnable — xOutput typConfigBlinker		
Visualization objects:		
ConfigBlinker	On-time %s Off-time %s	
Function description:		

The **FbBlinker** function block generates a blinking signal.

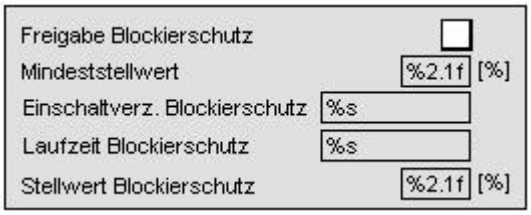
The configuration structure ***"typConfigBlinker"*** contains the following parameters:

- ***".tTimeHigh"*** defines the switch-on time for the blinker.
- ***".tTimeLow"*** defines the turn-off time for the blinker.

When the function block is activated via the ***"xEnable"*** input, the blinking signal is generated at the ***"xOutput"*** output.

Blocking Protection for Continuous Drivers (FbBlockingProtectionAnalog)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FbBlockingProtectionAnalog		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
xEnable	BOOL	Module enable	
rY_Driver	REAL	Specified position of the driver [%]	
typConfigBlocking ProtectionAnalog	←	Configuration parameters:	
.rY_Min	REAL	Minimum set value for the driver for detecting downtimes [%] Default setting = 3	
.tMaxOff	TIME	Maximum turn-off time for driver when blocking protection is activated Default setting = t#24h	
.tSwitchOn	TIME	Turn-on time of driver when blocking protection is enabled Default setting: t#60s	
.rY_BlockingProtection	REAL	Specified position for blocking protection [%] Value range = 0 – 100 Default setting = 100	
.xBlockingProtection	BOOL	Enable blocking protection Default setting = TRUE	
Return value:	**Data type:**	**Comment:**	
rY	REAL	Set value, driver Value range = 0 – 100	
wY	WORD	Set value, driver Value range = 0 – 32767	
Graphical illustration:			

Visualization objects:	
ConfigBlockingProtection Analog	

Function description:

The **FbBlockingProtectionAnalog** function block provides a blocking protection function for analog actuating drives. In order to avoid blocking of the driver after extended outage periods, the driver can be put into operation at least once within a certain period of time.

The configuration structure ***"typConfigBlockingProtectionAnalog"*** contains the following parameters:

- ***".rY_Min"*** defines the set value that must at least be reached to change the driver position.
- ***".xBlockingProtection"*** enables the blocking protection function.
- ***".tMaxOff"*** defines the maximum turn-off time until the blocking protection function is started.
- ***".tSwitchOn"*** defines the runtime for blocking protection.
- ***".rY_BlockingProtection"*** defines the specified position during the blocking protection function.

The blocking protection is only checked in position 0 - *„.rY_Min“* and activated after *„.tMaxOff“*.

The function block is enabled via the ***"xEnable"*** input.

The set value from the ***"rY_Driver"*** is output directly at the ***"rY"*** output as long as the blocking protection function is not active.

The output value ***"wY"*** has the same meaning as the *"rY"* output, the output just has the standardized values between 0 – 32767.

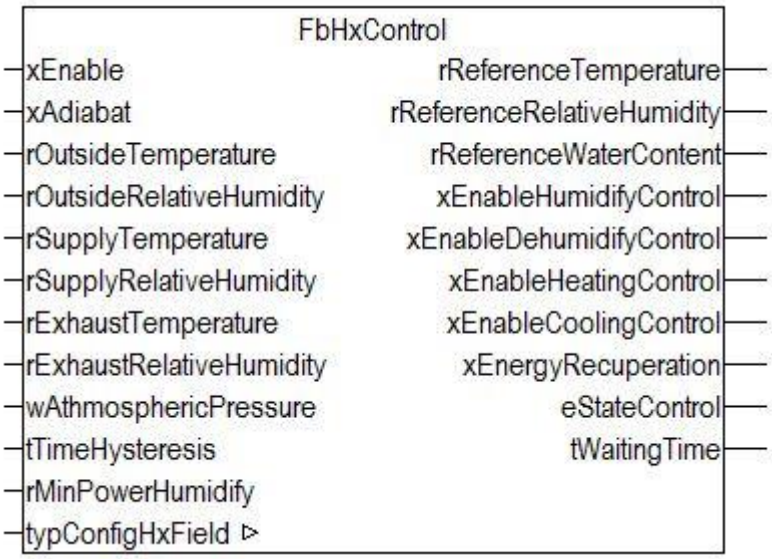
Note:

Blocking protection can also be activated by a timer program, so that a potential driver error message is issued only during a defined time period.

Comfort zone controller (FbHxControl)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbHxControl	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Enable h–x control
xAdiabat	BOOL	Adiabatic humidification system
rOutsideTemperature	REAL	Outside temperature
rOutsideRelativeHumidity	REAL	Relative humidity of the outside air
rSupplyTemperature	REAL	Supply air temperature Only required for visualization purposes
rSupplyRelativeHumidity	REAL	Relative humidity of the supply air Only required for visualization purposes
rExhaustTemperature	REAL	Exhaust air Only required for visualization purposes
rExhaustRelativeHumidity	REAL	Relative humidity of the exhaust air Only required for visualization purposes
wAthmosphericPressure	WORD	Atmospheric pressure Default setting = 1013
tTimeHysteresis	TIME	Hysteresis time for sector change Default setting = T#5s
rMinPowerHumidify	REAL	Hysteresis of minimum power for adiabatic humidification
Input / output parameters:	**Data Type:**	**Comment:**
typConfigHxField	typConfigHxField	Configuration parameters of the comfort zone
.rA	REAL	Lower boundary temperature Default setting = 20
.iB	INT	Left boundary relative humidity Default setting = 30
.rC	REAL	Upper boundary temperature Default setting = 26
.rD	REAL	Right boundary absolute humidity Default setting = 11
.iE	INT	Right boundary relative humidity Default setting = 65
xShowVisu	BOOL	Switch variable for the configuration menu
Return value:	**Data type:**	**Comment:**
rReferenceTemperature	REAL	Reference value room temperature [°C]
rReferenceRelativeHumidity	REAL	Reference value relative humidity [%]

WAGO-I/O-PRO Library Elements		
rReferenceWaterContent	REAL	Reference value water content [g/kg]
xEnableHumidifyControl	BOOL	Activation of humidification
xEnableDehumidifyControl	BOOL	Activation of dehumidification
xEnableHeatingControl	BOOL	Activation of heating
xEnableCoolingControl	BOOL	Activation of cooling
xEnergyRecuperation	BOOL	Activation of heat recovery
eStateControl	eStateControl	Status
.hxControlDisable		Controller is off.
.hxCooling		Cooling sequence active
.hxHeating		Heating sequence active
.hxHumidify		Humidification sequence active
.hxDehumidify		Dehumidification sequence active
.hxHeating_and_Humidify		Heating and humidification sequence active
.hxCooling_and_Humidify		Cooling and humidification sequence active
.hxEnergy_Recuperation		Heat recovery sequence active
tWaitingTime	TIME	Remaining time of the tTimeHysteresis input
Graphical illustration:		



Function description:

The comfort zone controller "**FbHxControl**" governs the required control sequences of the secondary controller in order to ensure control that makes sense in terms of energy without having a negative effect on human comfort.

Comfort zone control

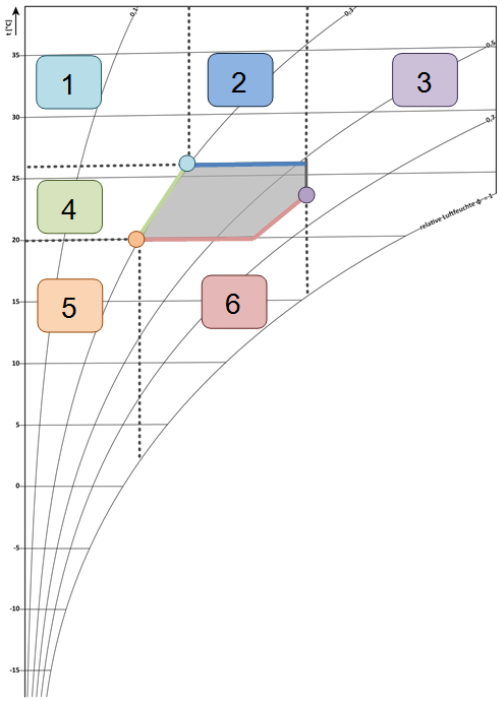


Figure 1: Comfort Zone

The so-called comfort zone specifies a range in the h–x diagram in which humans find the climate comfortable. In this zone, humans do not find the air too dry or too humid for the respective temperatures, for example. If the state of the air moves out of this range, humans can find the air too warm and too dry, for example.

For the comfort zone, there is no specific point in the h–x diagram provided as a setpoint for the system to move towards (for example, 25 °C and 50 % relative humidity); rather there is a specified zone. In particular, the controller moves towards the points and boundaries of the specified zone that make the most sense in terms of energy. Depending on the starting point, the physical properties of the air may make it necessary to execute a wide variety of control sequences in order to reach the targeted points and boundaries of the zone. This allows the h–x diagram around the comfort zone to be divided into six sectors with different activation points (see Figure 1: Comfort Zone). This type of control is very effective from the point of view of energy, since only as much energy is used as is necessary to reach the zone and not to remain in it.

1. Cooling and humidification
2. Cooling
3. Dehumidification and subsequent heating if necessary
4. Humidification
5. Heating and humidification
6. Heating

Configuration parameters

The "**typConfigHxField**" configuration structure specifies the comfort zone and contains the following parameters:

- "**rA**" Lower boundary of the temperature
- "**iB**" Left boundary of the relative humidity
- "**rC**" Upper boundary of the temperature
- "**rD**" Right boundary of the absolute relative humidity
- "**iE**" Right boundary of the relative humidity

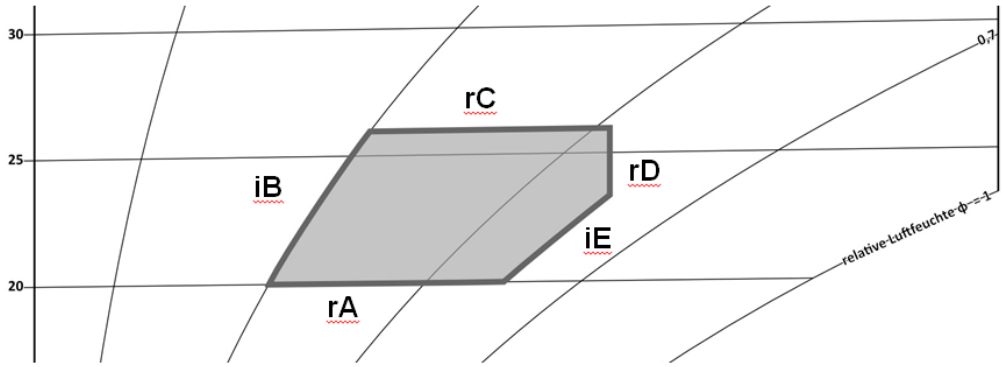


Figure 2: Comfort Zone Parameter typConfigHxField

The "**tTimeHysteresis**" input serves to prevent flitting back and forth between the ranges. If the state of the air moves from one range into the other, the controller will use the latter range for control for the amount of time set in the "**tTimeHysteresis**" input.

The "**xAdiabat**" input specifies what kind of humidification system is used. The "**rMinPowerHumidify**" input is directly related to the humidification. This is illustrated in Figure 3: rMinPowerHumidify. The behavior of the air during humidification would be different depending on the humidification system. In isothermal humidification, besides being added, water is also heated so the air is not cooled by the humidification. In this case, the comfort zone would be approached parallel to the comfort zone. In adiabatic humidification in contrast, no energy is expended to heat the water, so the air is cooled simultaneously. To avoid wasting too much energy or drifting out of the comfort zone, a delta can be set with the "**rMinPowerHumidify**" input starting at which the humidification activates. If the state of the air is within this delta, it is not humidified.

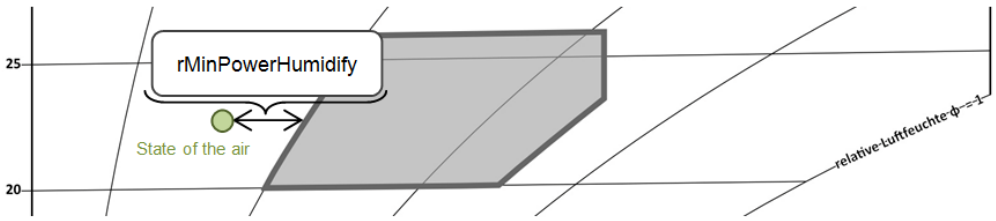
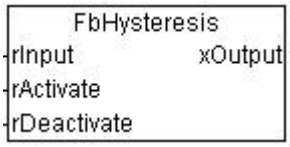


Figure 3: rMinPowerHumidify

Hysteresis (FbHysteresis)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FbHysteresis		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
rInput	REAL	Input value	
rActivate	REAL	Threshold value at which the output signal is set to TRUE	
rDeactivate	REAL	Threshold value at which the output signal is set to FALSE	
Return value:	**Data type:**	**Comment:**	
xOutput	BOOL	Output Signal	
Graphical illustration:			



Visualization objects:	
ConfigHysteresis	Activate threshold value %2.1f De-activate threshold value %2.1f

Function description:

This function block permits a switching function with adjustable hysteresis.

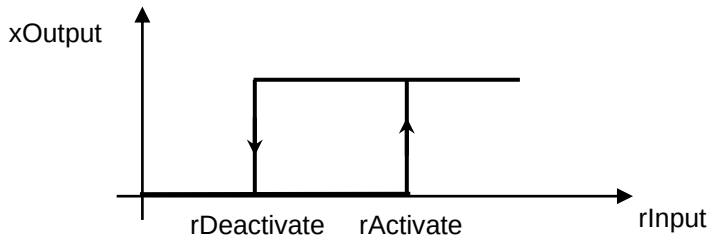
Two variations are to be considered during the analysis of the input values:

1) rActivate > rDeactivate

The output signal ***"xOutput"*** is set to TRUE, if the condition ***"rInput"*** ≥ ***"rActivate"*** is fulfilled.

The output signal *"xOutput"* is set to FALSE, if the condition *"rInput"* ≤ *"rDeactivate"* is fulfilled.

The output signal does not change as long as the input value moves between the values "rActivate" and "rDeactivate".

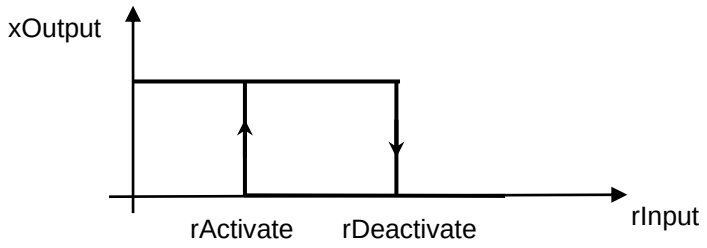


2) rActivate ≤ rDeactivate

The output signal *“xOutput”* is set to TRUE, if the condition *"rInput" ≤ "rActivate"* is fulfilled.

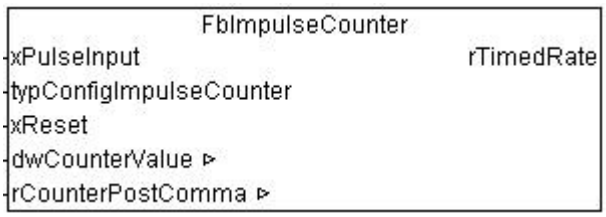
The output signal *“xOutput”* is set to FALSE, if the condition *“rInput“ ≥ “rDeactivate“* is fulfilled.

The output signal does not change as long as the input value moves between the values *“rActivate“* and *“rDeactivate“*.



Impulse Counter (FbImpulseCounter)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbImpulseCounter	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment**
xPulseInput	BOOL	Impulse input
typConfigImpulseCounter	←	Configuration parameters:
.rUnitPerPulse	REAL	Unit per impulse, e.g., 0.01 kWh Default setting = 1
.wBaseTimePeriod	WORD	Time base [s] Default setting = 1
xReset	BOOL	A positive edge initializes the counter.
Input/output parameters:	**Data type:**	**Comment:**
dwCounterValue	DWORD	Calculated consumption
rCounterPostComma	REAL	Value of the calculated consumption after the comma
Return value:	**Data type:**	**Comment:**
rTimedRate	REAL	Currently needed power
Graphical illustration:		



Visualization objects:	
ConfigImpulseCounter	Unit per impulse (e.g., 0.1 KWh) %2.2f Time base %d [s]

Function description:

The **FbImpulseCounter** function block is used for integrating meters with an impulse interface (e.g., electricity, heat or water meters).

The configuration structure ***"typConfigImpulseCounter"*** contains the following parameters:

- ***".rUnitPerPulse"*** defines the unit per impulse, e.g., 1 impulse = 0.01kWh
- ***".wBaseTimePeriod"*** defines the time base for output calculation.

This function block counts the impulses at the ***"xPulseInput"*** and calculates the consumption values from this (energy).

The counter values are deleted by a positive edge at the ***"xReset"*** input.

If the counter is to be initialized with values, the variables ***"dwCounterValue"*** and ***"rCounterPostComma"*** can be directly overwritten.

Power measurement:

The pulses are extrapolated with their valence for the defined time base in order to determine the current output ***"rTimedRate"***.

Notes:

1.) The calculation of the performance is not exact and regular. The output value for the performance therefore only gives an approximate overview of the currently needed performance.

2.) The program cycle time must be less than the time between two pulses.

3.) The input/output variables *"dwCounterValue"* and *"rCounterPostComma"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

4.) If more than 2 decimal places of the *„typConfigImpulseCounter.rUnitPerPulse"* are used, the global variable *„g_bDecimalPlaces"* must be changed.

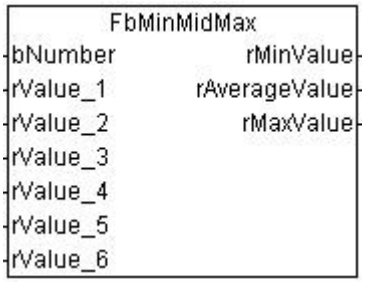
Min. Value, Avg. Value and Max. Value (FbMinMidMax)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbMinMidMax	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	

Input parameters:	**Data type:**	**Comment:**
bNumber	BYTE	Number of inputs for calculating the minimum, average and maximum values
rValue_1	REAL	Inputvalue_1
rValue_2	REAL	Inputvalue_2
rValue_3	REAL	Inputvalue_3
rValue_4	REAL	Inputvalue_4
rValue_5	REAL	Inputvalue_5
rValue_6	REAL	Inputvalue_6

Return value:	**Data type:**	**Comment:**
rMinValue	REAL	Minimum value
rAverageValue	REAL	Average Value
rMaxValue	REAL	Maximum value

Graphical illustration:



Function description:

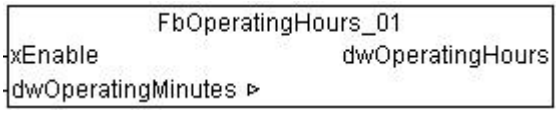
The **FbMinMidMax** function block calculates a minimum value, an average value and a maximum value from up six values.

The value ***"bNumber"*** indicates how many inputs are analyzed for the calculation of these values.

The minimum value ***"rMinValue",*** the average value ***"rAverageValue"*** and the maximum value ***"rMaxValue"*** are calculated from the input values ***"rValue_1" – "rValue_6"***.

Operating Hours Counter (FbOperatingHours_01)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	FbOperatingHours_01		
Type:	Function ☐	Function block ☒	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment**	
xEnable	BOOL	Enable Operating hours counter	
Input/output parameters:	**Data type:**	**Comment:**	
dwOperatingMinutes	DWORD	Minutes of operation	
Return value:	**Data type:**	**Comment:**	
dwOperatingHours	DWORD	Operating Hours	
Graphical illustration:			



Function description:

The **FbOperatingHours_01** function block determines the operating hours, expressed in minutes.

When the ***"xEnable"*** input is activated, the minutes of operation ***"dwOperatingMinutes"*** are counted upward minute by minute.

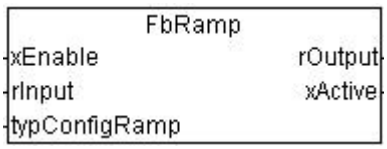
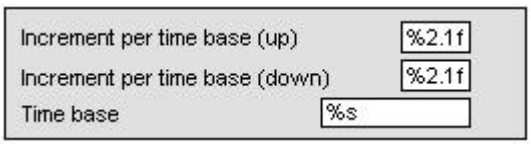
If the counter is to be initialized with values, the variable *"dwOperatingMinutes"* can be directly overwritten.

The operating hours calculated from the minutes of operation are indicated at the ***"dwOperatingHours"*** output.

Notes:

The operating minutes function *"dwOperatingMinutes"* should be defined as RETAIN PERSISTENT so that the set values are retained in the event of a loss of power or after a project upload.

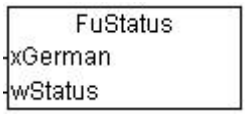
Ramp (FbRamp)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FbRamp	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xEnable	BOOL	Activation of the ramp function
rInput	REAL	Input value for the ramp
typConfigRamp	←	Configuration parameters:
.rStepRangeUp	REAL	Maximum value change rate (upwards) per unit of time Default setting = 1
.rStepRangeDown	REAL	Maximum value change rate (downwards) per unit of time Default setting = 1
.tTimeBase	TIME	Unit of time for change rate Default setting = t#1 m
Return value:	**Data type:**	**Comment:**
rOutput	REAL	Output value for ramp
xActive	BOOL	The ramp is active
Graphical illustration:		
		
Visualization objects:		
ConfigRamp		

Function description:
The **FbRamp** function block ensures a defined rising or falling rate for a particular setting. The configuration structure ***"typConfigRamp"*** contains the following parameters: • ***".tTimeBase"*** defines the time base for rising and falling rate. • ***".rStepRangeUp"*** defines the maximum upward value change per time base *".tTimeBase"* (rising rate) • ***".rStepRangeDown"*** defines the maximum downward value change per time base *".tTimeBase"* (falling rate) When the function block is activated via the *"xEnable"* input, the output signal *"rOutput"* follows the input signal *"rInput"* only as long as the rising or falling rate of the input signal is less than the maximum rising or falling rate. If the input signal changes more rapidly, the output follows the input signal at the defined maximum rising or falling rate. When the function block is deactivated, the output signal *"rOutput"* follows the *"rInput"* input signal directly. The ***"xActive"*** output indicates whether the ramp is active.

Status Indication as STRING (FuStatus)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FuStatus	
Type:	Function ☒	Function block ☐ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
xGerman	BOOL	Language selection TRUE = German FALSE = English
wStatus	WORD	Status message for the HVAC function blocks (number code).
Return value:	**Data type:**	**Comment:**
FuStatus	STRING	Status message as plain text
Graphical illustration:		



Function description:

The **FuStatus** function converts the status message for the HVAC function blocks into a STRING.

When the ***"xGerman"*** input is activated, the status messages are given in German at the function output.

When the ***"xGerman"*** input is deactivated, the status messages are given in English at the function output.

The ***"wStatus"*** contains the numerical code for the text message.

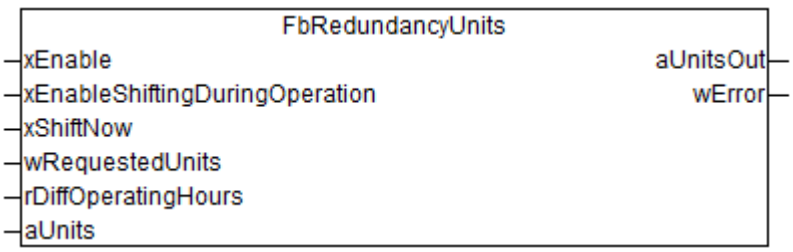
Unit monitoring (FbRedundancyUnits)

WAGO-I/O-*PRO* Library Elements		
Category:	Building technology	
Name:	FbRedundancyUnits	
Type:	Function ☐	Function block ☒ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	

Input parameter:	**Data type:**	**Comment:**
xEnable	BOOL	Function block enable
xEnableShiftingDuringOperation	BOOL	Enable whether units may shift during operation
xShiftNow	BOOL	Signal to execute an immediate unit shift
wRequestedUnits	WORD	Number of units requested Default setting = 0
rDiffOperationHours	REAL	Maximum deviation of operating hours until a unit shift is executed Default setting = 24.0 [h]
aUnits	ARRAY [0..wMaxRedundancyUnits] OF typUnit	Contains the operating hours and a status signal of all units
wMaxUnits	WORD (CONSTANT)	Number of all physically available units (maximum limit of the arrays) Default setting = 6

Return value:	**Data type:**	**Comment:**
aUnitsOut	ARRAY [0..wMaxRedundancyUnits] OF BOOL	Binary output array of the units requested
wError	WORD	Internal error code: 1: Insufficient Units available

Graphical illustration:



Function description:
The **FbRedundancyUnits** function block is used to ensure uniform utilization of all units in the case of redundantly designed units and to enable replacement units in case of breakdown. To avoid excessive wear, e.g., of an individual pump, the operating hours of all available units are compared to then switch on the specific unit or units with the lowest number of hours of operation. If an enabled unit fails (e.g., due to a fault), the next possible unit is automatically activated. The operating hours are also taken into account. The module can ben enabled or disabled at the ***"xEnable"*** input. Whether the monitored units may be switched during operation is can be enabled or disabled at the ***"xEnableShifitingDuringOperation"*** input. If the input is set to TRUE, the unit is shifted automatically once the difference in operating ours is greater than or equal to the time specified at the ***"rDiffOperatingHours"*** input. If the input is set to FALSE, the unit is shifted once the unit currently running is no longer available, the time of the unit to be switched is changed or the unit is manually switched at the ***"xShiftNow"*** input. When manually switching the unit, please note that the maximum time difference at the *"rDiffOperatingHours"* input is not taken into consideration. The number of units to be switched on by the module is specified by the ***"wRequestedUnits"*** parameter and can be dynamically adjusted to the operating status, e.g., for an output increase/decrease. The current switched status is output in the binary ***"aUnitsOut"*** output array where the units with the same index [0..wMaxRedundancyUnits] at the input and output are identical. The ***"aUnits"*** input array must be configured externally and contains the following parameters of the individual units: • ***"xUnitAvailable"*** specifies whether the unit is available and error free. • ***"dwUnitOperatingMinutes"*** specifies the current number of operating hours in minutes. • ***"wUnitId"*** is an internal parameter and may not be changed/defined. If not enough units are available to switch on, error code "1" is output at the ***"wError"*** output.

Examples:

- 3 supply pumps
  - 1 pump is required.
  - Switch independent of operating hours
  - Automatic fault changeover
- 10 circulation cooling units
  - Requirement for the number of units by cooling load (1-6)
  - Switch independent of operating hours during operation
  - Automatic fault changeover

WAGO-I/O-*PRO* Library Elements		
Category:	Building technology	
Name:	typUnit	
Type:	Data type ☒	Enumeration ☐
Applicable to:	See Release Note	
Declaration:		

```
TYPE typUnit:
STRUCT
        xUnitAvailable                  :BOOL; [writeable]
        dwUnitOperatingMinutes          :DWORD; [writeable]
        wUnitId                         :WORD; [not writeable]
END_STRUCT
END_TYPE
```

17 Characteristic Curves

KTY Characteristic Curve (FuKTY)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	FuKTY	
Type:	Function ☒	Function block ☐ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	
Input parameters:	**Data type:**	**Comment:**
rKTY	REAL	Resistance level at the KTY sensor [Ω]
rKTY25	REAL	Resistance level at the KTY sensor at 25 °C [Ω] Default setting = 2000
Return value:	**Data type:**	**Comment:**
FuKTY	REAL	Temperature measured by the KTY sensor [°C]
Graphical illustration:		

Function description:

The **FuKTY** function calculates the measured temperature from the KTY sensor resistance value.

The measured resistance of the KTY sensor is linked to the ***"rKTY"*** input.

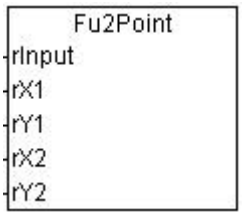
On account of the different KTY characteristic curves, the resistance value for 25°C should be determined from the associated characteristic and this defined at the *"rKTY25"* input.

The measured temperature is signaled at the **FuKTY** function output.

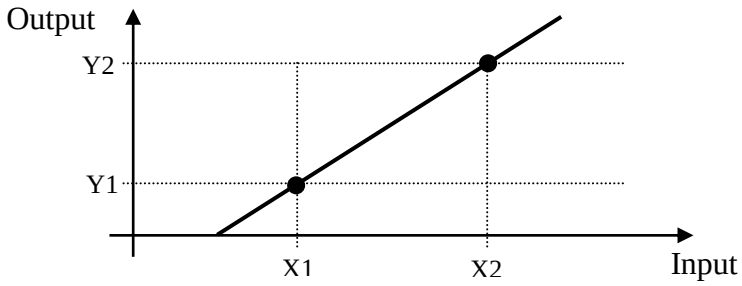
Two-Point Characteristic (Fu2Point)

WAGO-I/O-PRO Library Elements			
Category:	Building Automation		
Name:	Fu2Point		
Type:	Function ☒	Function block ☐	Program ☐
Name of library:	Building_HVAC_03.lib		
Applicable to:	See Release Note		
Input parameters:	**Data type:**	**Comment:**	
rInput	REAL	Input value	
rX1	REAL	x-coordinate of the first value	
rY1	REAL	y-coordinate of the first value	
rX2	REAL	x-coordinate of the second value	
rY2	REAL	y-coordinate of the second value	
Return value:	**Data type:**	**Comment:**	
Fu2Point	REAL	Output value	

Graphical illustration:



Characteristic:



Function description:

The **Fu2Point** function describes a linear equation defined by the two points (***"rX1", "rY1"***) and (***"rX2", "rY2"***).

The input value ***"rInput"*** is converted in accordance with the linear equation and output at the **Fu2Point** function output.

If *"rX1"* and *"rX2"* are identical (vertical characteristic), the output is set to zero. If *"rY1"* and *"rY2"* are identical, the output is set to *"rY1"*.

Note:
The reference points X must always be entered in an ascending order
(rX1 < rX2).

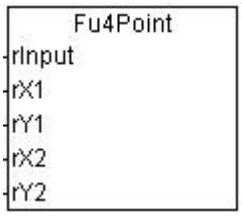
Four-Point Characteristic Curve (Fu4Point)

WAGO-I/O-PRO Library Elements		
Category:	Building Automation	
Name:	Fu4Point	
Type:	Function ☒	Function block ☐ / Program ☐
Name of library:	Building_HVAC_03.lib	
Applicable to:	See Release Note	

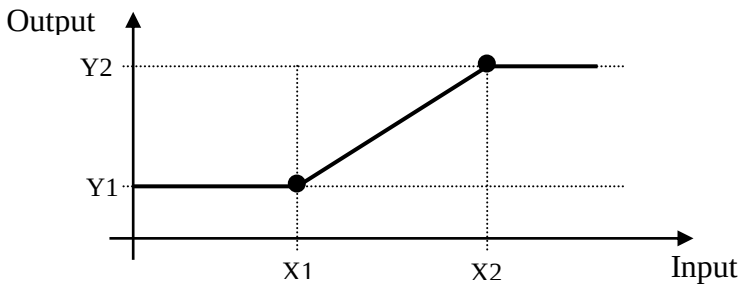
Input parameters:	**Data type:**	**Comment:**
rInput	REAL	Input value
rX1	REAL	x-coordinate of the first value
rY1	REAL	y-coordinate of the first value
rX2	REAL	x-coordinate of the second value
rY2	REAL	y-coordinate of the second value

Return value:	**Data type:**	**Comment:**
Fu4Point	REAL	Output value

Graphical illustration:



Characteristic:



Function description:

The **Fu4Point** function describes a linear equation with lower and upper delimitation. It is described by the two inflection pairs ("***rX1", "rY1"***) and ("***rX2", "rY2"***).

The output value is limited to the minimum value Y1 or to the maximum value Y2 in the case of input values ***"rInput"*** smaller than X1 and larger than X2. The output value changes according to a the linear equation between these two values.

Note:
The reference points X must always be entered in an ascending order
(rX1 < rX2).

WAGO Kontakttechnik GmbH & Co. KG
PO Box 2880 D-32385 Minden
Hansastraße 27 D-32423 Minden
Phone: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69
E-Mail: info@wago.com

Internet: http://www.wago.com