PRODUCT MANUAL

ABB i-bus® KNX

VC/S x.2.2.2

Valve Drive Controller

Table of contents

1 About this document

1.1 Using the product manual

This manual provides detailed technical information on the function, installation and programming of the ABB i-bus® KNX device.

1.2 Legal disclaimer

ABB AG reserves the right to make changes to the product or modify the contents of this document without prior notice.

The agreed properties are definitive for any orders placed. ABB AG does not accept any responsibility whatsoever for potential errors or possible lack of information in this document.

ABB AG reserves all rights in this document and in the subject matter and illustrations contained therein. Reproduction, transfer to third parties or processing of the content – including sections thereof – is not permitted without the prior written consent of ABB AG.

1.3 Explanation of symbols

1. 2. ⇨	Instructions in specified sequence and result
▶	Individual actions
a)	Priorities
1)	Processes run by the device in a specific sequence
•	List level 2
–	List level 2

Tab. 1: Explanation of symbols

Notes and warnings are represented as follows in this manual:

DANGER
This symbol is a warning about electrical voltage and indicates high-risk hazards that will definitely result in death or serious injury unless avoided.

DANGER
Indicates high-risk hazards that will definitely result in death or serious injury unless avoided.

WARNING
Indicates medium-risk hazards that could result in death or serious injury unless avoided.

CAUTION
Indicates low-risk hazards that could result in slight or moderate injury unless avoided.

CAUTION
Indicates a risk of malfunctions or damage to property and equipment, but with no risk to life and limb.

Example
For use in application, installation and programming examples

Note
For use in tips on use and operation

1.4 2D code

The packaging and the device are labeled with a 2D code. These codes are used for unique identification of the device and include the following information:

- Link to the product page
- Order number
- ABB device serial number

The 2D codes can be read using any mobile device with an appropriate 2D code reader.

By scanning the 2D codes with the ABB Product Scanner, you can open additional digital services.

2 Safety

2.1 General safety instructions

- Protect the device from moisture, dirt and damage during transport, storage and operation.
- Operate the device only in a closed housing (distribution board).
- Operate the device only within the specified technical data.
- Mounting, installation, commissioning and maintenance must be carried out only by qualified electricians.
- Disconnect device from the supply of electrical power before mounting.

2.2 Qualification of the specialist personnel

Programming the device requires detailed specialist knowledge – particularly about the ETS commissioning software – through KNX training courses.

2.3 Proper use

The Valve Drive Controllers VC/S x.2.2.2 are intended to be used to activate 2-point valve drives for underfloor heating, radiators or cooling ceilings in a KNX environment.

3 Product overview

3.1 Device description

The devices are modular installation devices (MDRC) in pro*M* design. They are designed for installation in electrical distribution boards and small housings with a 35 mm mounting rail (according to EN 60715).

The devices are KNX-certified and can be used as products in a KNX system
→ EU declaration of conformity.

The devices are powered via the bus (ABB i-bus® KNX) and require an additional supply voltage for the valve outputs.

The connection to the bus (ABB i-bus® KNX) is made via a KNX bus connection terminal on the front of the housing.

The connections at the inputs or outputs are made via screw terminals
→ terminal designation on the housing.

The software application Engineering Tool Software (ETS) is used for physical address assignment and parameterization.

3.1.1 Membrane keypad

The devices can be operated manually using the membrane keypad.

Note
The functions *Service* and *Forced operation*, as well as operation using the i-bus® Tool, take priority over the *Manual operation* mode. If an output is blocked by a higher-priority function or is being operated using the i-bus® Tool, it cannot be operated using the membrane keypad.

3.2 Product name description

The table below lists the product name descriptions of all devices in the product family.

Abbreviation	Description		
VC	Valve Drive Controller		
/S	MDRC		
x.	6	=	6-fold
	12	=	12-fold
x.	2	=	2 terminals (L and N) per valve drive
x.	2	=	Manual Operation
x	x	=	Version number (x = 1, 2, etc.)

Tab. 2: Product name description

3.3 Ordering details

Description	MB	Type	Order no.	Packaging unit [pcs.]	Weight (incl. packaging) [kg]
Valve Drive Controller	4	VC/S 6.2.2.2	2CDG110330R0011	1	0.210
Valve Drive Controller	8	VC/S 12.2.2.2	2CDG110331R0011	1	0.362

Tab. 3: Ordering details

3.4 Connections

The devices have the following connections:
- Depending on the device type, 6 or 12 valve outputs for activating 2-point valve drives
- Depending on the device type, 1 or 2 inputs for connecting the supply voltage
- 1 bus connection

3.4.1 Inputs

WARNING – Severe injuries due to touch voltage
With SELV and non-SELV (> 50 V) mixed operation at the power supplies, SELV is no longer ensured. Touching can result in currents flowing through the body and causing severe injuries.
▸ SELV and non-SELV (> 50 V) mixed operation is impermissible.

VC/S 6.2.2.2:
- Supply voltage (24 … 230 V AC) connects at terminal pair 1/2 (power supply for valve outputs A … F).

VC/S 12.2.2.2:
- Supply voltage (24 … 230 V AC) connects at terminal pairs 1/2 (power supply for valve outputs A … F) and 15/16 (power supply for valve outputs G … L).
- Terminal pairs 1/2 and 15/16 are not connected internally within the device
- Two isolated RCD circuits can be connected
- SELV and non-SELV mixed operation is impermissible

Note
The Group Objects *Status Fault supply voltage* and *Status Fault valve* report a fault if there is no supply voltage (VC/S 6.2.2.2: terminal pair 1/2, VC/S 12.2.2.2: terminal pairs 1/2 and 15/16). If the upper outputs (valve outputs A ... F) or the lower outputs (valve outputs G ... L) are not in use on the VC/S 12.2.2.2 and the corresponding supply voltage is not connected, the corresponding outputs should be deactivated (→ parameter *Output X Valve type*) to avoid unwanted fault messages.

3.4.2 Outputs

Note
Several valve drives can be connected in parallel at each output. The possible number of valve drives per output depends on the maximum inrush current of the valve drives used. The maximum permissible rated current of the output pairs (0.25 A per output pair) must not be exceeded.

Each two adjacent outputs (A/B, C/D, etc.) form an output pair internal to the device. Each output pair has fault detection, overload protection and short-circuit resistance.

Connection options per ABB TSA/K valve drive output pair:
- 6x TSA/K 230.2
- 3x TSA/K 24.2

The valve drives can be freely arranged on any output pair, but symmetrical connection is recommended whenever possible.

3.5 Product family

The product family described in this document includes the following devices:

Device type	Name	Features
VC/S 6.2.2.2	Valve Drive Controller	6-fold, Manual Operation, MDRC
VC/S 12.2.2.2	Valve Drive Controller	12-fold, Manual Operation, MDRC

Tab. 4: Product family

3.5.1 Dimension drawing

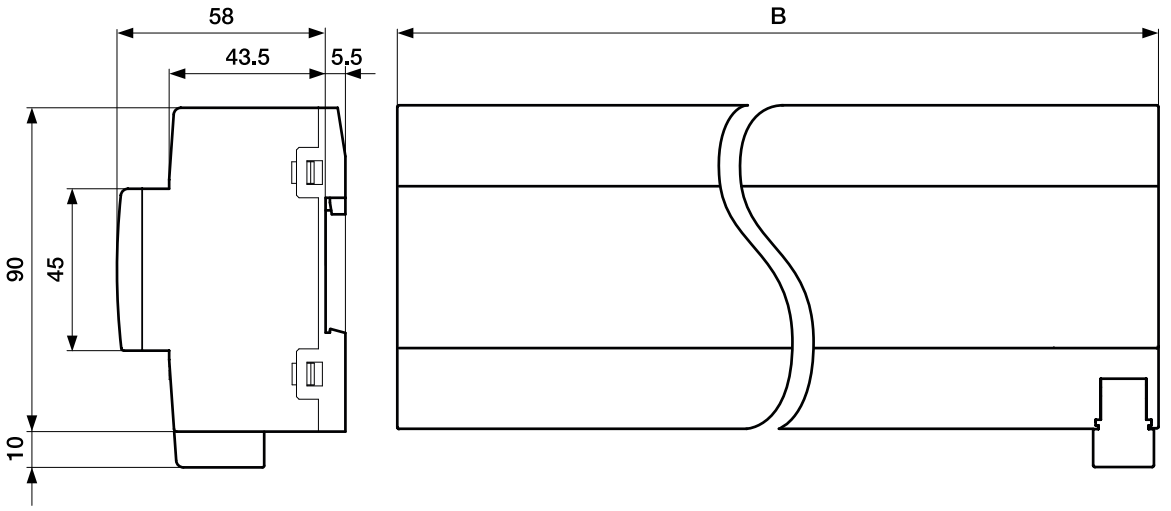


Fig. 1: Dimension drawing for product family

Device type	B
VC/S 6.2.2.2	4 space units, 70 mm
VC/S 12.2.2.2	8 space units, 140 mm

Tab. 5: Device width (space units/millimeters)

3.5.2 Connection diagram

Note
The connection variants are explained in the following based on examples.

3.5.2.1 Connection variant 1: 230 V with ABB TSA/K 230.2, Non-SELV

Note
Several valve drives can be connected in parallel at each output. The possible number of valve drives per output depends on the maximum inrush current of the valve drives used. The maximum permissible rated current of the output pairs (0.25 A per output pair) must not be exceeded.

More information: → Connections, Page 9.

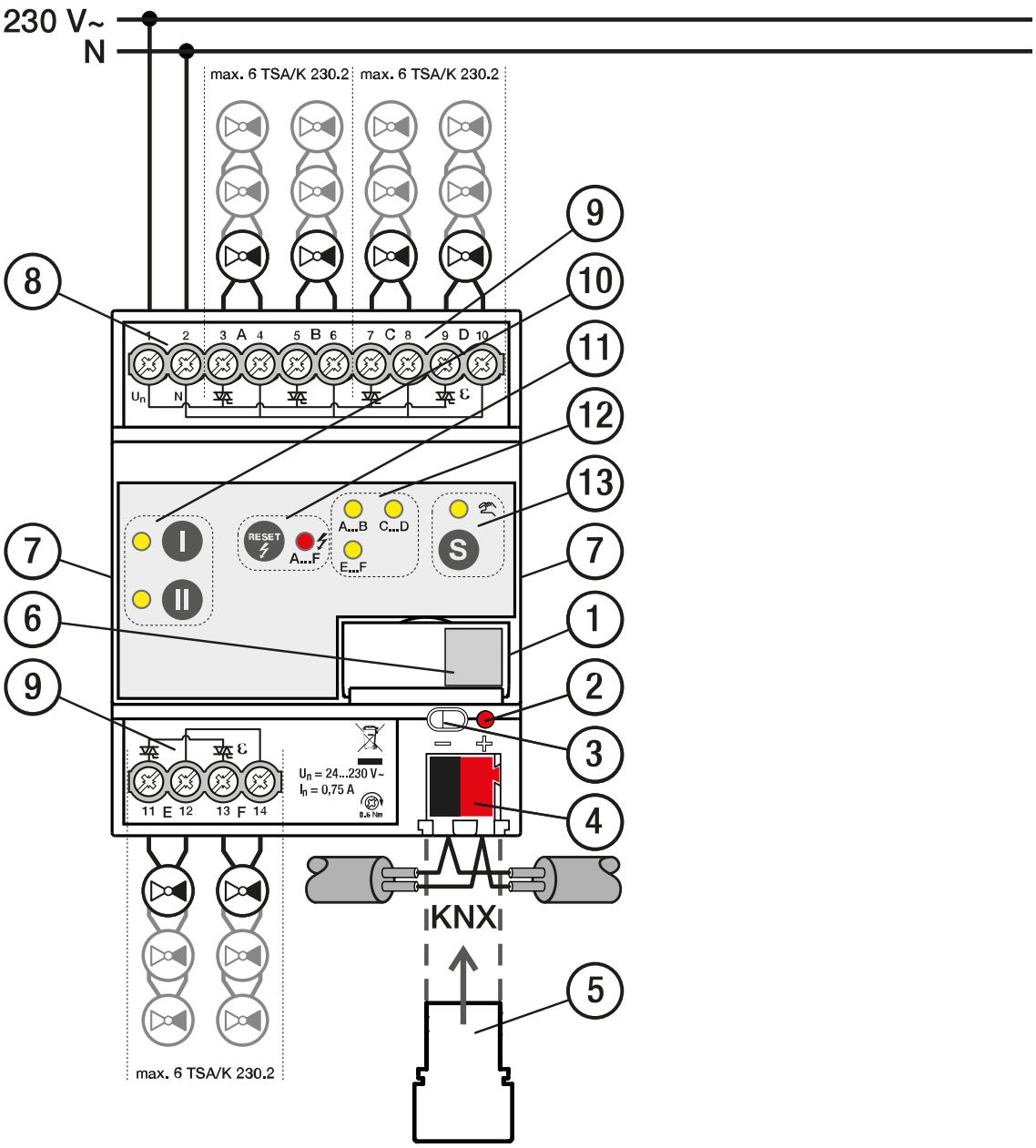


Fig. 2: Connection variant 1: VC/S 6.2.2.2 with ABB TSA/K 230.2, Non-SELV

9AKK108464A1003

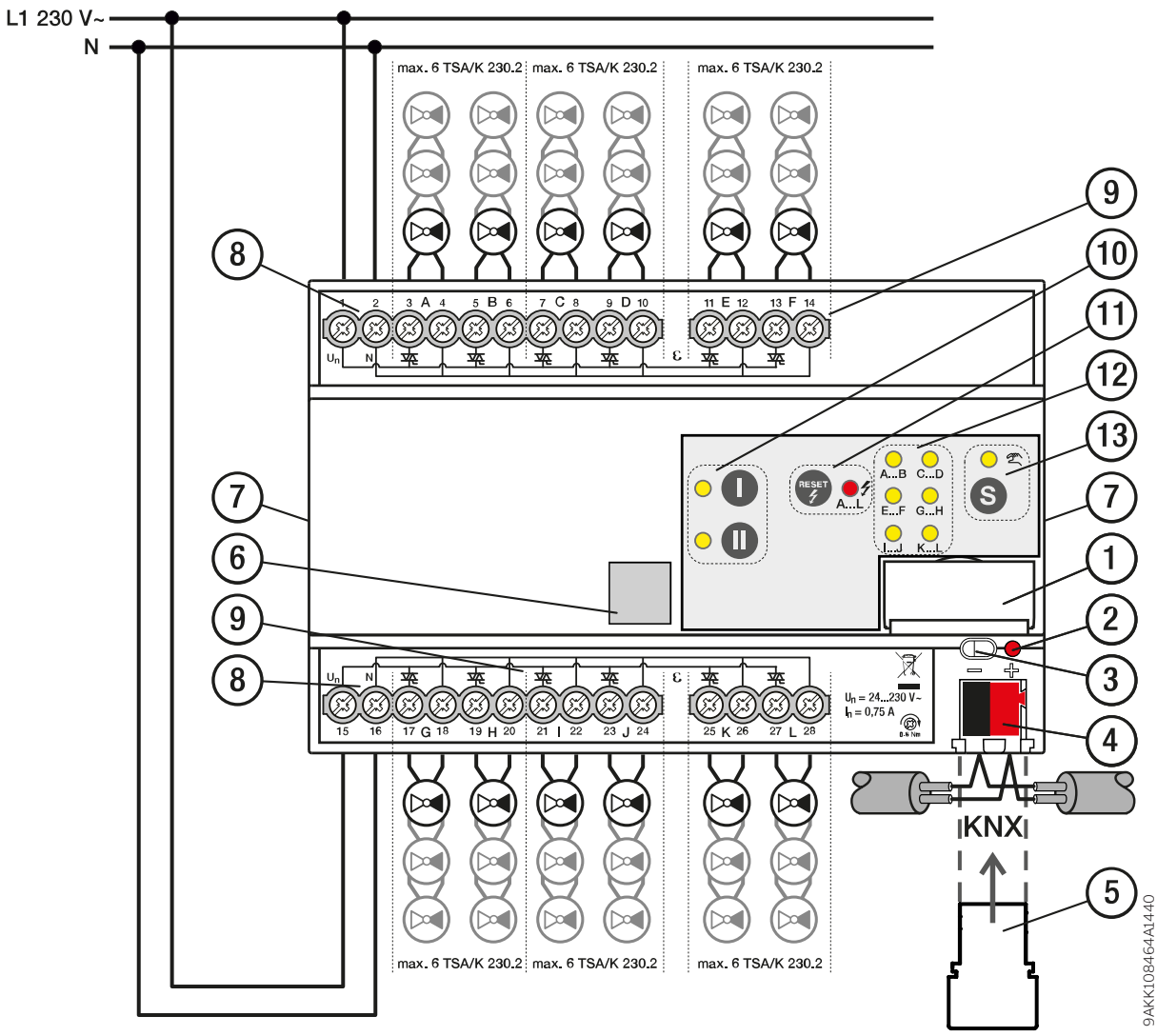


Fig. 3: Connection variant 1: VC/S 12.2.2.2 with ABB TSA/K 230.2, one 230 V RCD circuit, Non-SELV

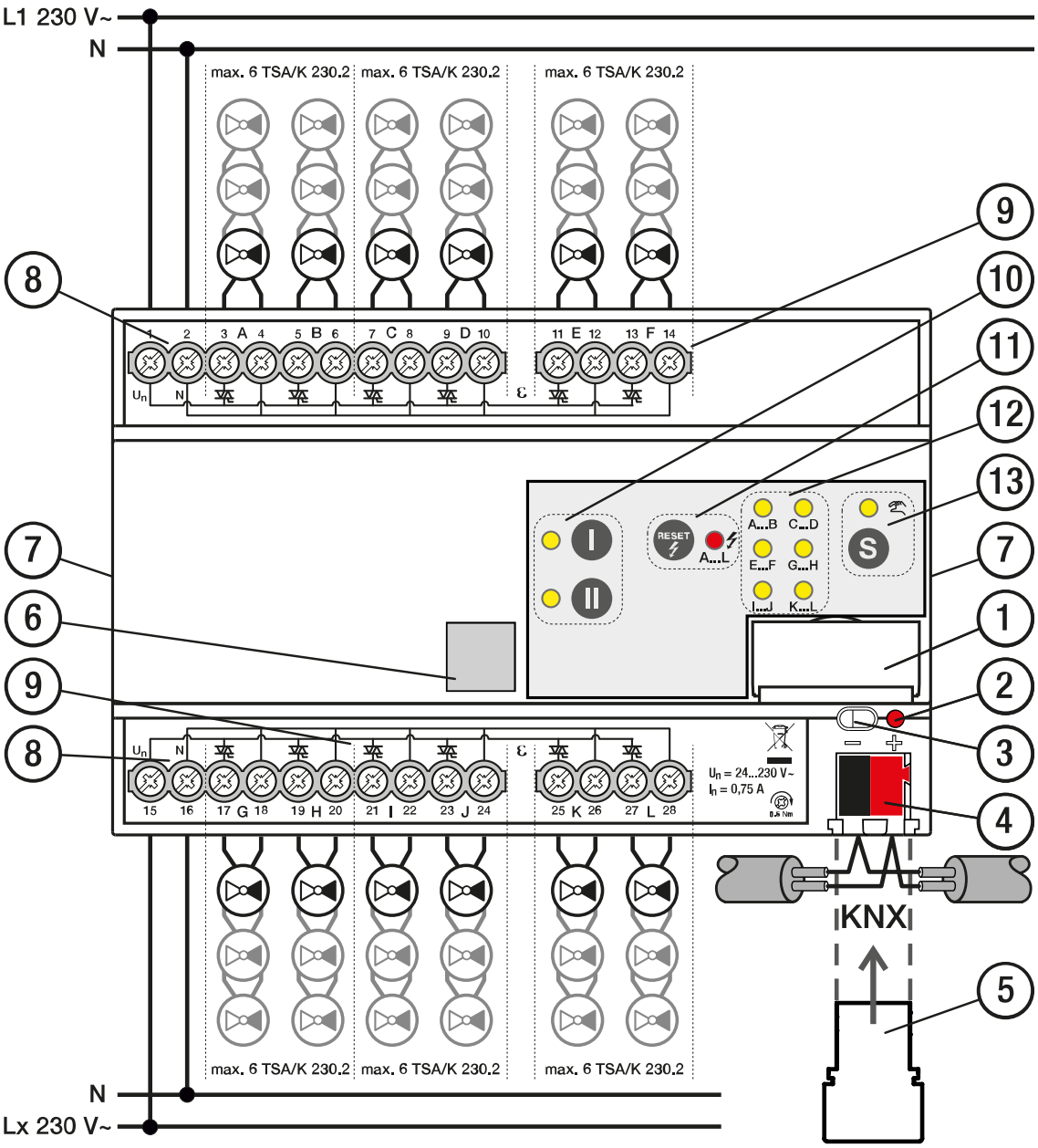


Fig. 4: Connection variant 1: VC/S 12.2.2.2 with ABB TSA/K 230.2, two 230 V RCD circuits, Non-SELV

—

Legend

1 label carrier
2 *Programming* LED
3 *Programming* button
4 KNX bus connection terminal
5 Cover cap
6 2D code
7 Device certificate/Identification label (on the side)
8 Supply voltage input
9 Valve output
10 *Valve output* button/LED
11 *Reset* button / *valve output error* LED
12 *Group* LED
13 *S* button / *manual operation* LED

9AKK108464A1005

3.5.2.2 Connection variant 2: 24 V with ABB TSA/K 24.2, SELV

Note
Several valve drives can be connected in parallel at each output. The possible number of valve drives per output depends on the maximum inrush current of the valve drives used. The maximum permissible rated current of the output pairs (0.25 A per output pair) must not be exceeded.

More information: → Connections, Page 9.

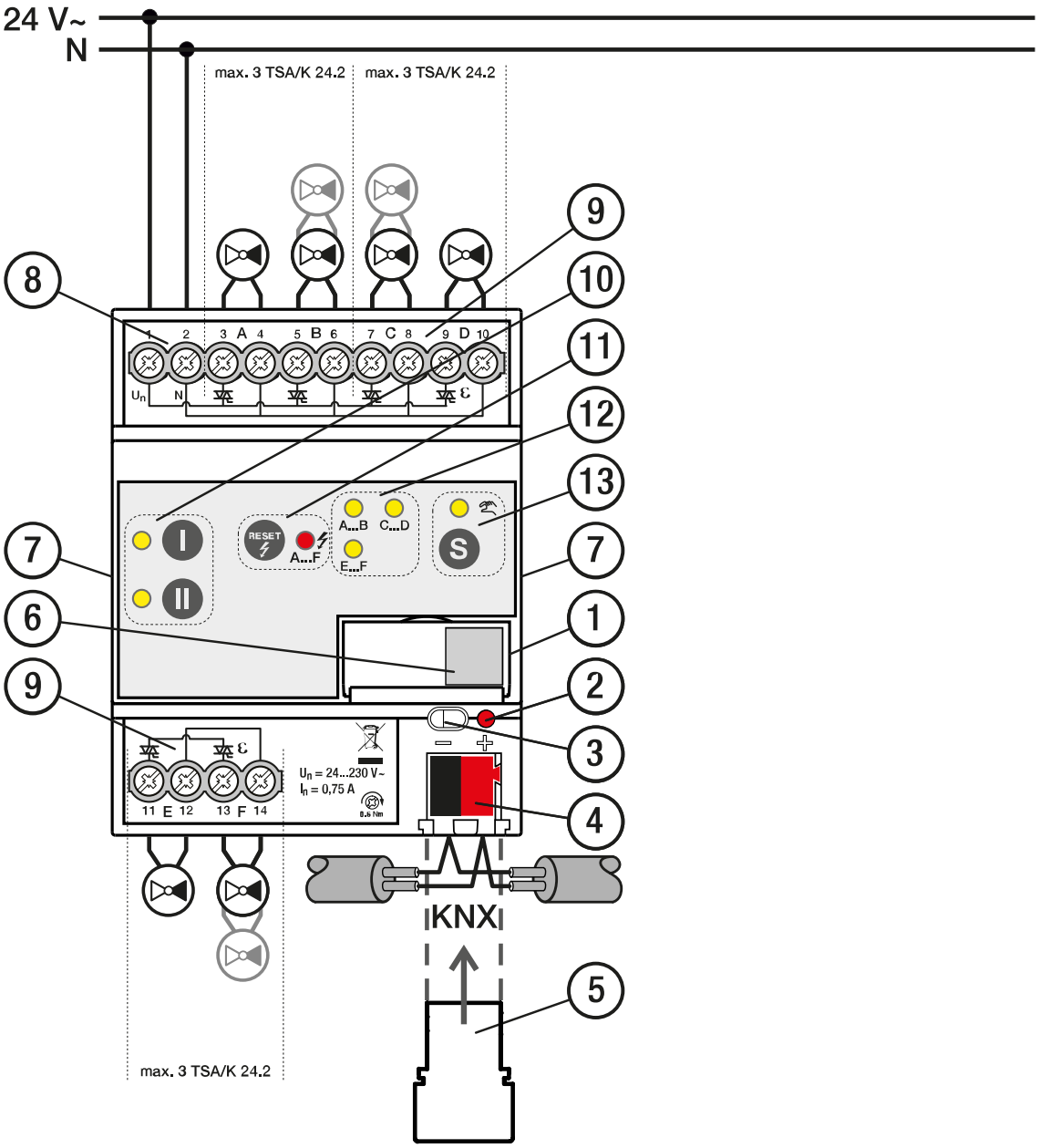


Fig. 5: Connection variant 2: VC/S 6.2.2.2 with ABB TSA/K 24.2, SELV

9AKK108464A1002

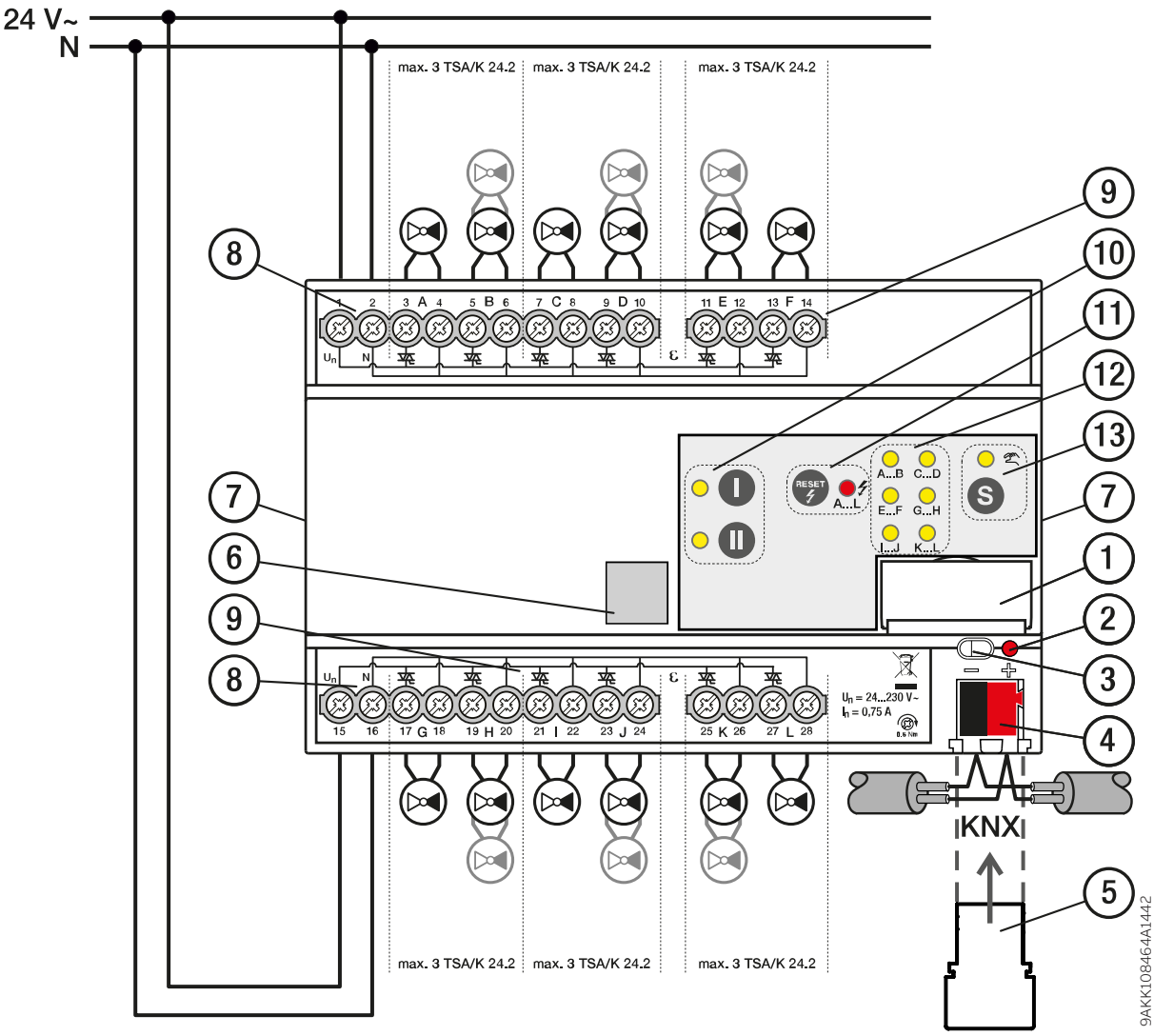


Fig. 6: Connection variant 2: VC/S 12.2.2.2 with ABB TSA/K 24.2, one 24 V RCD circuit, SELV

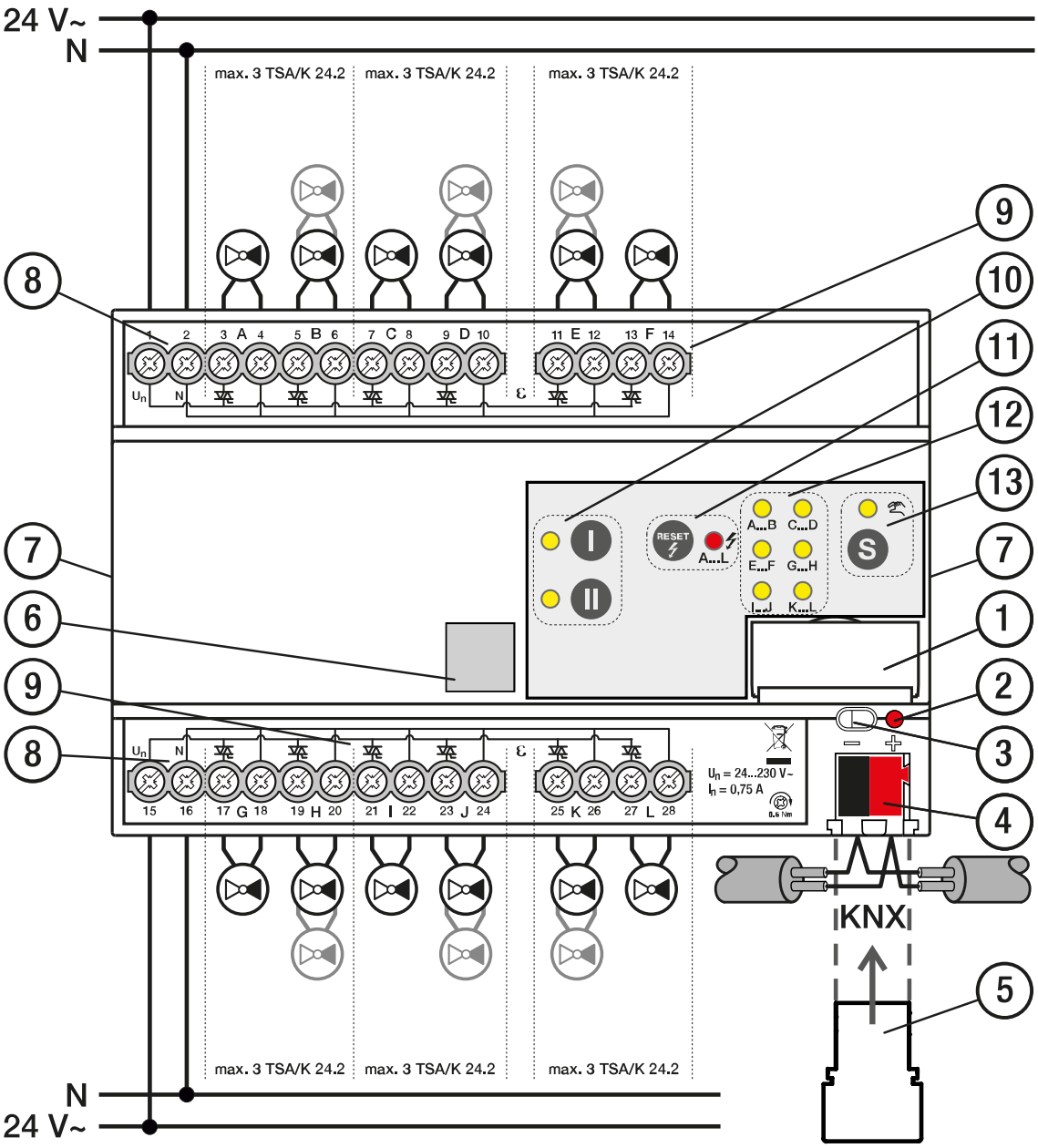


Fig. 7: Connection variant 2: VC/S 12.2.2.2 with ABB TSA/K 24.2, two 24 V RCD circuits, SELV

—

Legend

1 label carrier
2 *Programming* LED
3 *Programming* button
4 KNX bus connection terminal
5 Cover cap
6 2D code
7 Device certificate/Identification label (on the side)
8 Supply voltage input
9 Valve output
10 *Valve output* button/LED
11 *Reset* button / *valve output error* LED
12 *Group* LED
13 *S* button / *manual operation* LED

3.5.2.3 Connection variant 3: 24 ... 50 V, SELV

WARNING – Severe injuries due to touch voltage
With SELV and non-SELV (> 50 V) mixed operation at the power supplies, SELV is no longer ensured. Touching can result in currents flowing through the body and causing severe injuries.
- SELV and non-SELV (> 50 V) mixed operation is impermissible.

Note
Several valve drives can be connected in parallel at each output. The possible number of valve drives per output depends on the maximum inrush current of the valve drives used. The maximum permissible rated current of the output pairs (0.25 A per output pair) must not be exceeded.

More information: → Connections, Page 9.

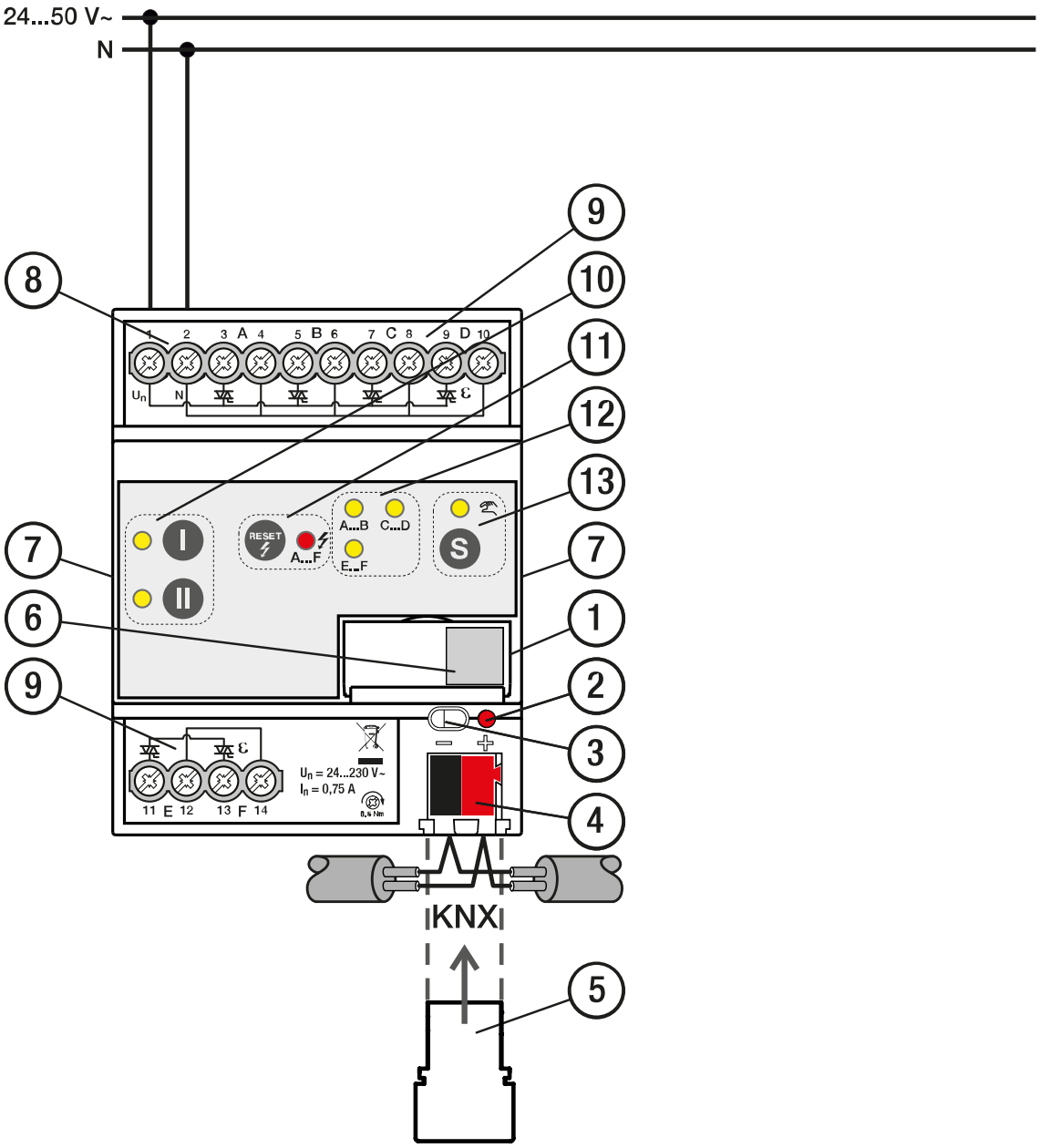


Fig. 8: Connection variant 3: VC/S 6.2.2.2, 24 … 50 V, SELV

9AKK108464A1443

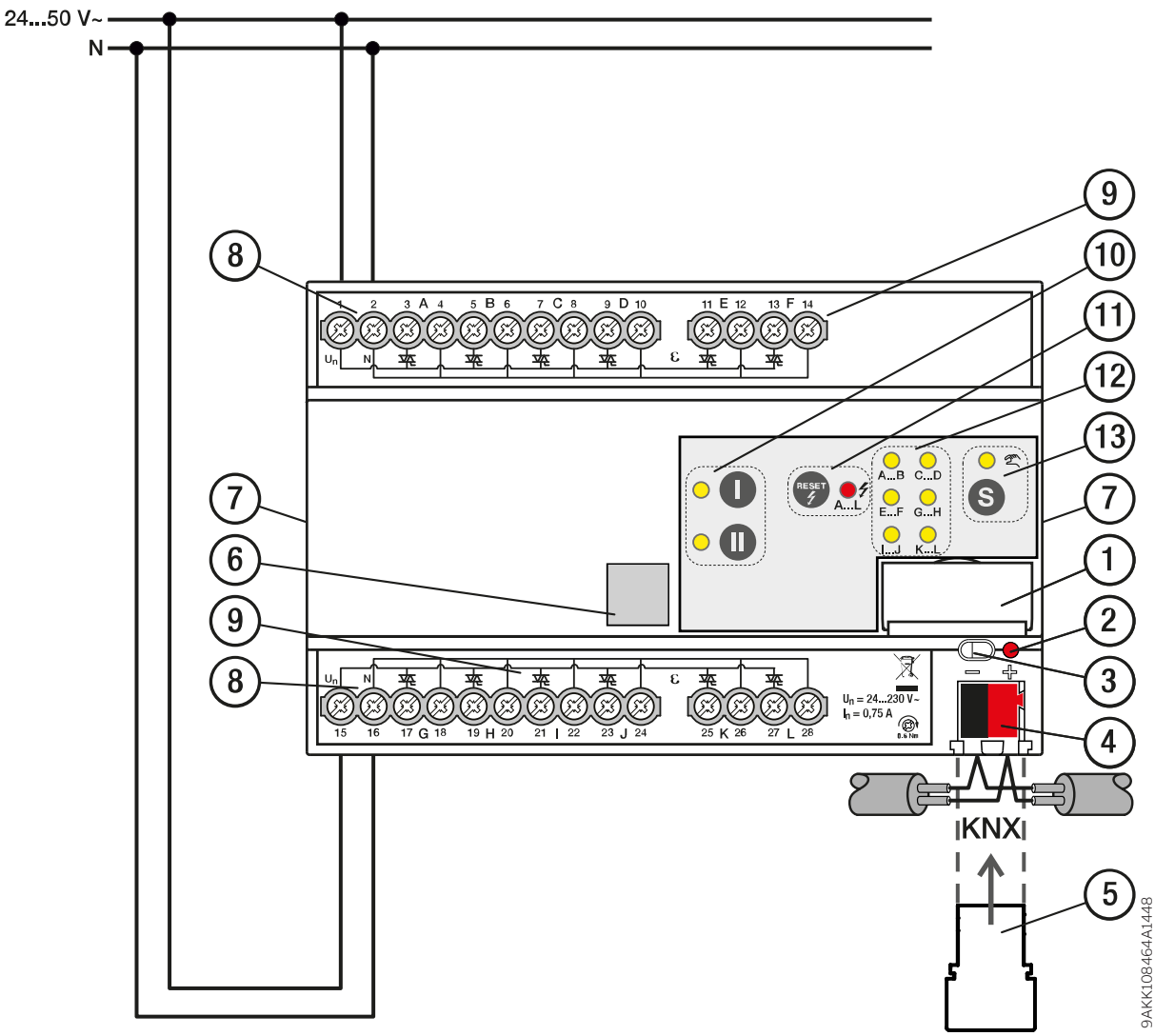


Fig. 9: Connection variant 3: VC/S 12.2.2.2, one 24 … 50 V RCD circuit, SELV

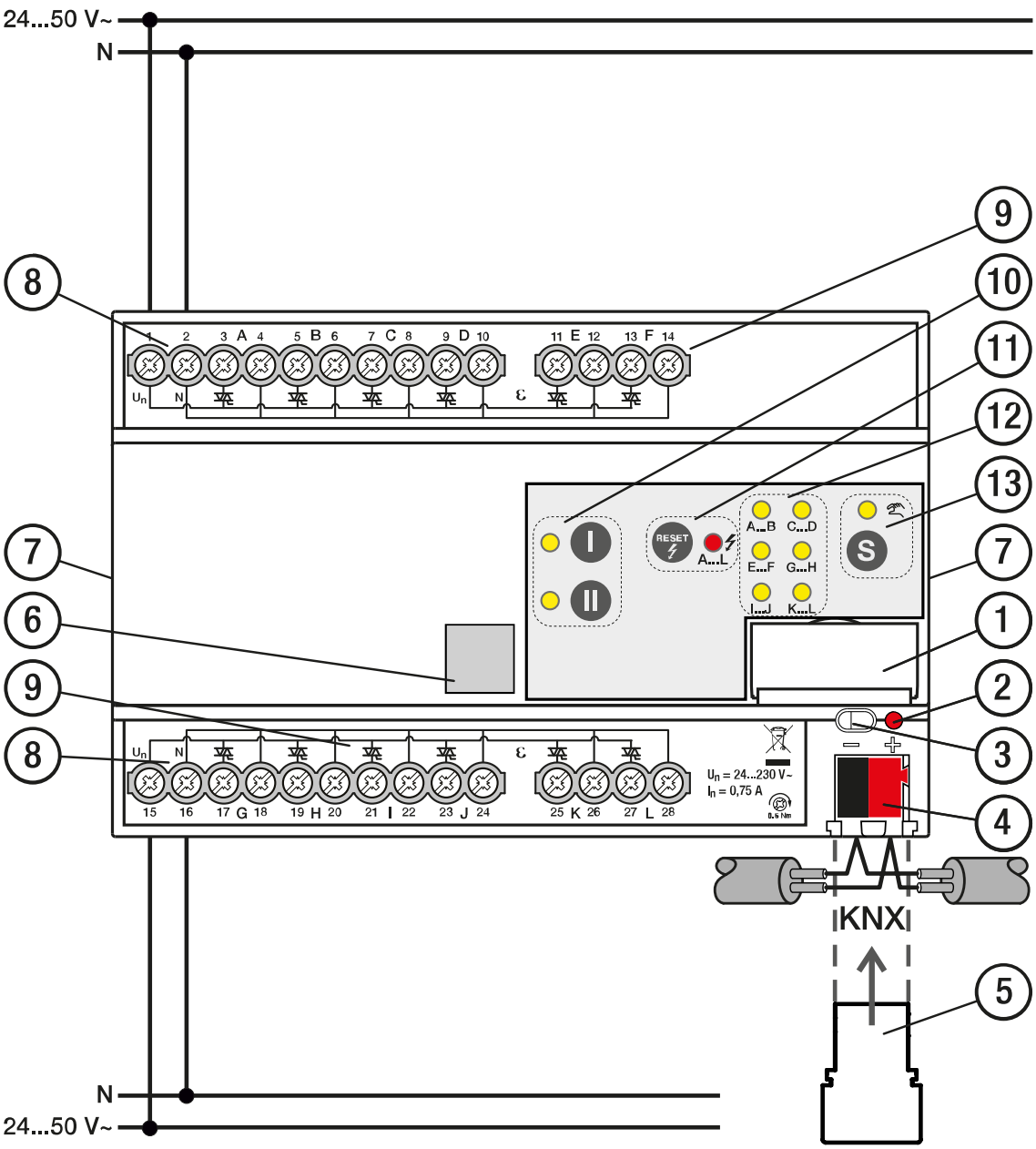


Fig. 10: Connection variant 3: VC/S 12.2.2.2, two 24 … 50 V RCD circuits, SELV

—

Legend

1 label carrier
2 *Programming* LED
3 *Programming* button
4 KNX bus connection terminal
5 Cover cap
6 2D code
7 Device certificate/Identification label (on the side)
8 Supply voltage input
9 Valve output
10 *Valve output* button/LED
11 *Reset* button / *valve output error* LED
12 *Group* LED
13 *S* button / *manual operation* LED

3.5.2.4 Connection variant 4: 51 ... 230 V, Non-SELV

WARNING – Severe injuries due to touch voltage
With SELV and non-SELV (> 50 V) mixed operation at the power supplies, SELV is no longer ensured. Touching can result in currents flowing through the body and causing severe injuries.
- SELV and non-SELV (> 50 V) mixed operation is impermissible.

Note
Several valve drives can be connected in parallel at each output. The possible number of valve drives per output depends on the maximum inrush current of the valve drives used. The maximum permissible rated current of the output pairs (0.25 A per output pair) must not be exceeded.

More information: → Connections, Page 9.

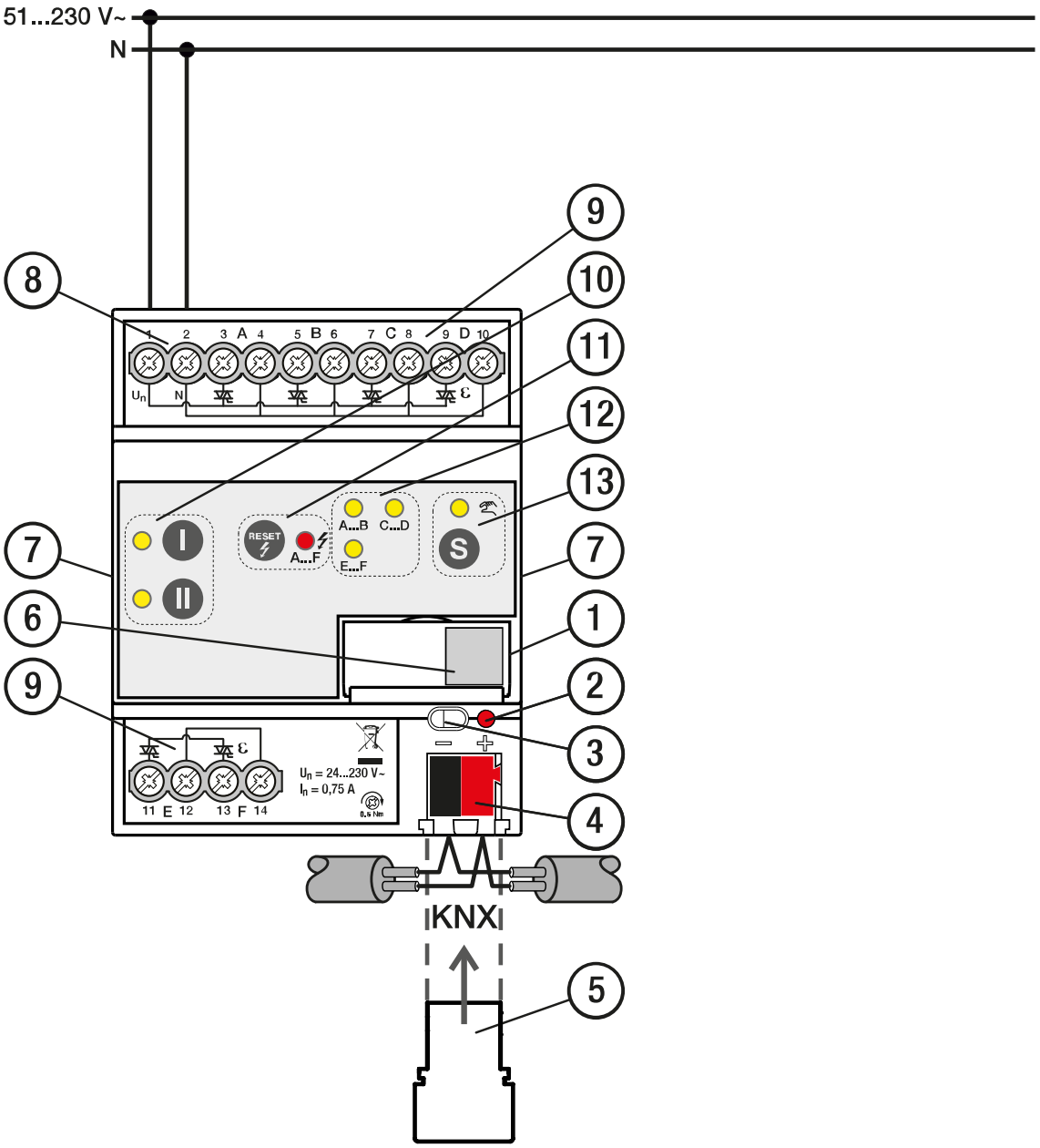


Fig. 11: Connection variant 4: VC/S 6.2.2.2, 51 ... 230 V, Non-SELV

9AKK108464A1449

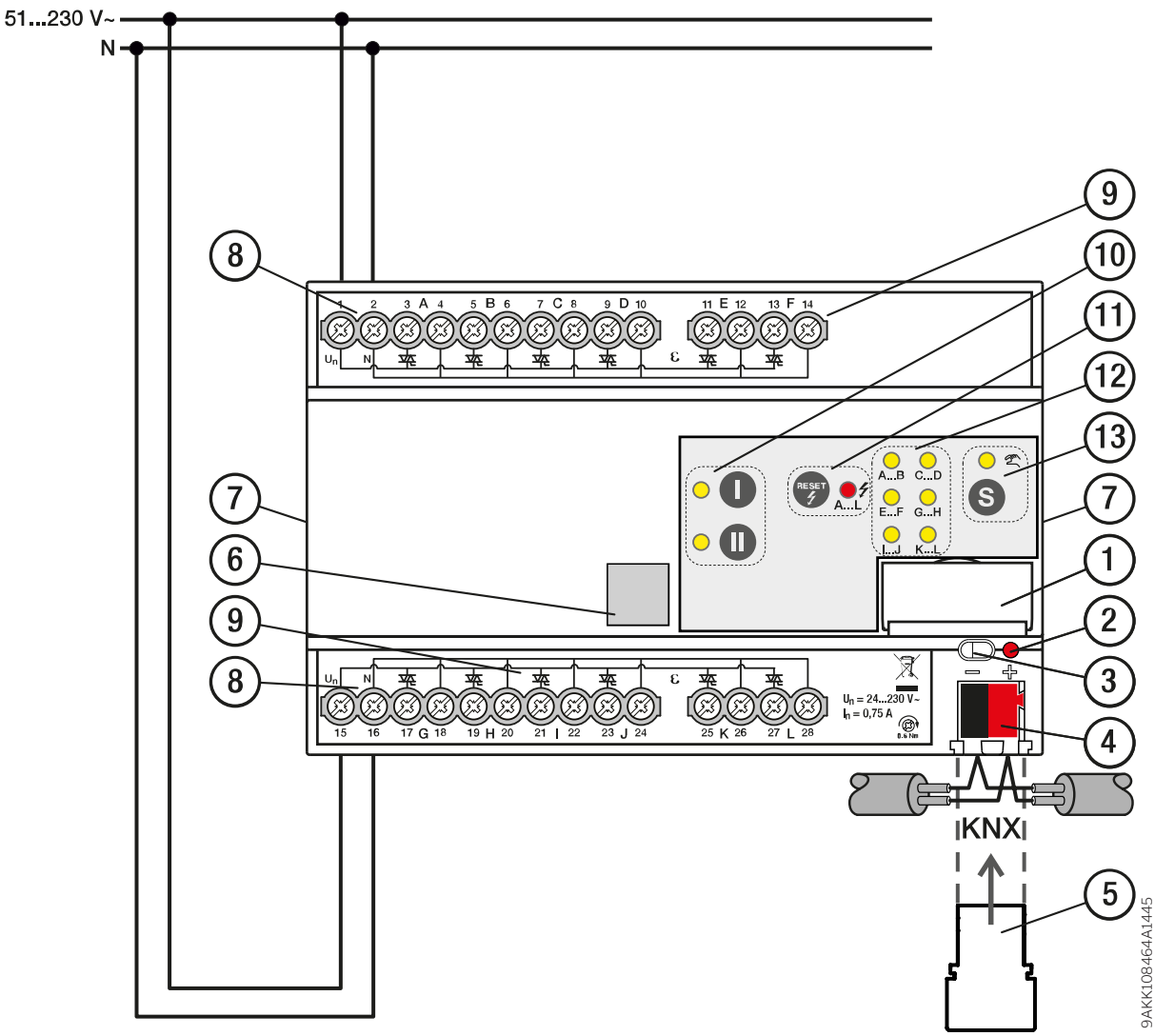


Fig. 12: Connection variant 4: VC/S 12.2.2.2, one 51 ... 230 V RCD circuit, Non-SELV

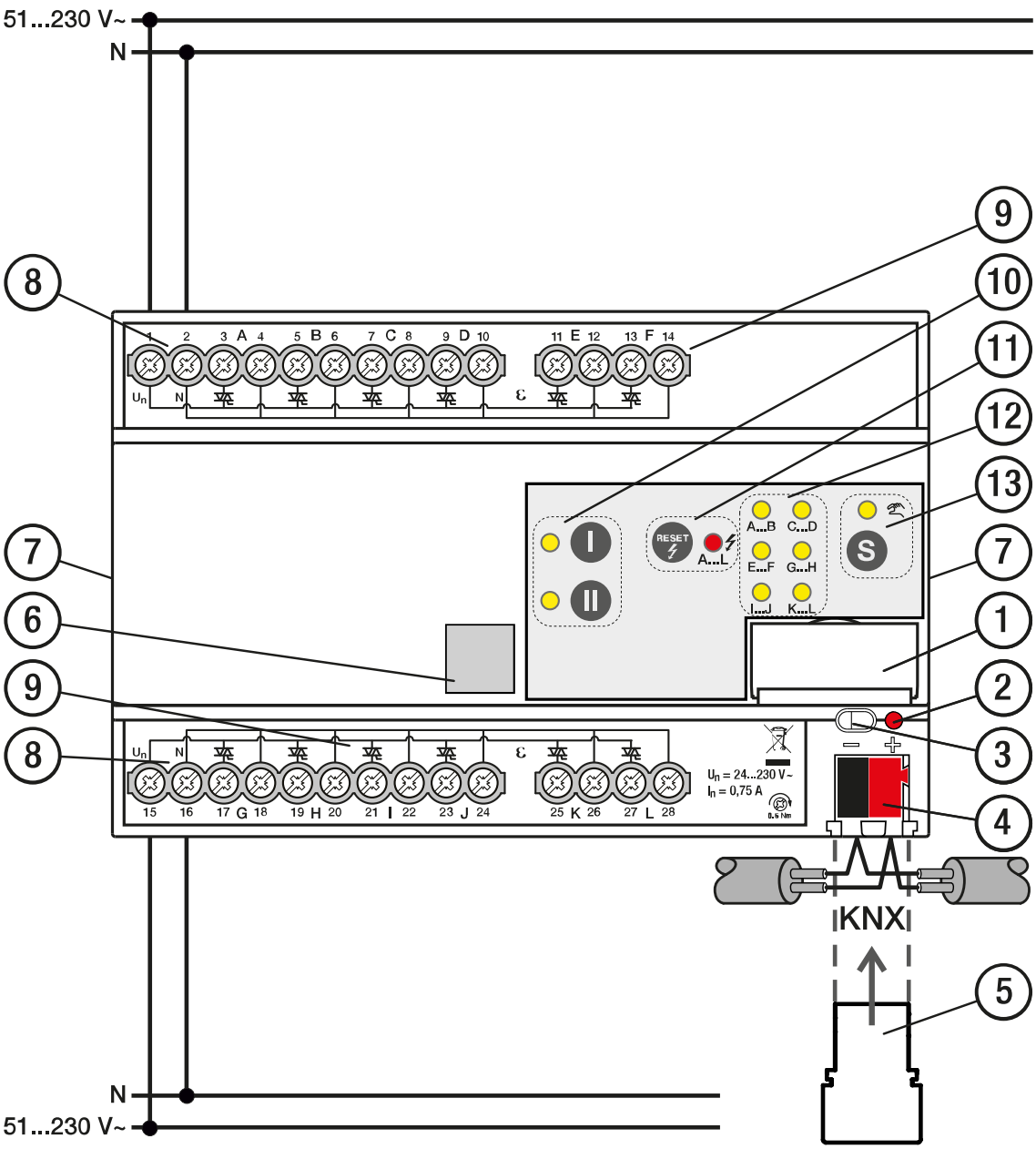


Fig. 13: Connection variant 4: VC/S 12.2.2.2, two 51 ... 230 V RCD circuits, Non-SELV

—

Legend

1 label carrier
2 *Programming* LED
3 *Programming* button
4 KNX bus connection terminal
5 Cover cap
6 2D code
7 Device certificate/Identification label (on the side)
8 Supply voltage input
9 Valve output
10 *Valve output* button/LED
11 *Reset* button / *valve output error* LED
12 *Group* LED
13 *S* button / *manual operation* LED

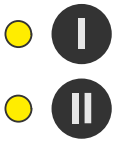
3.5.3 Operating and display elements

> **Note**
> The operating and display elements are shown with variables in the following tables for illustrative purposes only. All elements of the same type function in exactly the same way.

Operating control/LED	Description/function	Display
 *Programming* button/LED	Assignment of the physical address	LED on: Device in programming mode

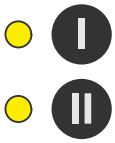
Tab. 6: Operating and display elements

3.5.3.1 Manual mode

Operating control/LED	Description/function	Display
 *S button* / *Manual operation* LED	Short button push < 2 s: Selection of group Button push 2 … 5 s: Changeover to *KNX operation* Long button push > 5 s: Selection of all outputs → Central operation via membrane keypad, Page 166	LED on: *Manual operation* active LED off: *KNX operation* active
 *Output group* LED		LED on: Group selected LED off: Group not selected
 *Reset* button / *valve output error* LED	Long button push > 5 s resets the fault message	LED on: Error on at least one output
 *Valve output* button/LED	Button I: First output of group. Switching of outputs A, C, E, G, I, K (toggle function) Button II: Second output of group. Switching of outputs B, D, F, H, J, L (toggle function)	LED flashing (5 Hz): Error (overload/short circuit) LED on: Control value > 0 % LED off: Control value = 0 %

Tab. 7: Operating and display elements

3.5.3.2 KNX operation

Operating control/LED	Description/function	Display
 *S button* / *Manual operation* LED	Short button push < 2 s: Selection of group Button push 2 … 5 s: Change to *Manual operation*	LED on: *Manual operation* active LED off: *KNX operation* active LED flashing (1 Hz): Device connected to i-bus® Tool, *Manual operation* blocked LED flashing (1 Hz) while button pressed: *Manual operation* not enabled or blocked
 *Output group* LED		LED on: Group selected LED off: Group not selected
 *Reset* button / *valve output error* LED	Button without function	LED on: Error on at least one output
 *Valve output* button/LED	Button without function	LED flashing (5 Hz): Error (overload/short circuit) LED on: Control value > 0 % LED off: Control value = 0 %

Tab. 8: Operating and display elements

3.5.4 Technical data

3.5.4.1 General technical data

		VC/S 6.2.2.2	VC/S 12.2.2.2
Device	Dimensions	90 × 70 × 63.5 mm (H x W x D)	90 × 140 × 63.5 mm (H x W x D)
	Mounting width in space units	4 modules, 17.5 mm each	8 modules, 17.5 mm each
	Weight	0.160 kg	0.292 kg
	Mounting position	Any	Any
	Mounting variant	35 mm mounting rail	35 mm mounting rail
	Design	pro*M*	pro*M*
	Degree of protection	IP 20	IP 20
	Protection class	II	II
	Overvoltage category	III	III
	Pollution degree	2	2
Materials	Housing	Polycarbonate, Makrolon FR6002, halogen free	Polycarbonate, Makrolon FR6002, halogen free
Material note	Fire classification	Flammability V-0	Flammability V-0
Electronics	Rated voltage, bus	30 V DC	30 V DC
	Voltage range, bus	21 … 31 V DC	21 … 31 V DC
	Current consumption, bus	< 12 mA	< 12 mA
	Maximum current, device	0.75 A	1.5 A
	Power loss, device	≤ 4 W	≤ 8 W
	Power loss, bus	≤ 0.25 W	≤ 0.25 W
	KNX safety extra low voltage	SELV	SELV
Connections	Connection type, KNX bus	Plug-in terminal	Plug-in terminal
	Cable diameter, KNX bus	0.6 … 0.8 mm, solid	0.6 … 0.8 mm, solid
	Connection type, outputs	Screw terminal with universal head (PZ 1)	Screw terminal with universal head (PZ 1)
	Pitch	6.35 mm	6.35 mm
	Tightening torque, screw terminals	0.5 … 0.6 Nm	0.5 … 0.6 Nm
	Conductor cross-section, flexible	1 × (0.2 … 4 mm²) / 2 × (0.2 … 2.5 mm²)	1 × (0.2 … 4 mm²) / 2 × (0.2 … 2.5 mm²)
	Conductor cross section, rigid	1 × (0.2 … 6 mm²) / 2 × (0.2 … 2.5 mm²)	1 × (0.2 … 6 mm²) / 2 × (0.2 … 2.5 mm²)
	Conductor cross section with wire end ferrule without plastic sleeve	1 × (0.25 … 4 mm²) / 2 × (0.25 … 0.75 mm²)	1 × (0.25 … 4 mm²) / 2 × (0.25 … 0.75 mm²)
	Conductor cross section with wire end ferrule with plastic sleeve	1 × (0.25 … 2.5 mm²)	1 × (0.25 … 4 mm²)
	Conductor cross section with TWIN wire end ferrule	1 × (0.5 … 2.5 mm²)	1 × (0.5 … 2.5 mm²)
	Dimensions of plastic sleeve for TWIN wire end ferrule (W x D)	≤ 4.4 × 8 mm	≤ 4.4 × 8 mm
	Length, wire end ferrule contact pin	8 mm	8 mm
	Stripping length for KNX terminal	6 mm	6 mm
	Stripping length for load terminal	8 mm	8 mm
Certificates and declarations	CE declaration of conformity	→ 9AKK108464A0999	→ 9AKK108464A1000
Ambient condition	Operation	−5 … +45 °C	−5 … +45 °C
	Transport	−25 … +70 °C	−25 … +70 °C
	Storage	−25 … +55 °C	−25 … +55 °C
	Humidity	≤ 95%	≤ 95%
	Condensation allowed	No	No
	Atmospheric pressure	≥ 80 kPa (corresponds to air pressure at 2,000 m above sea level)	≥ 80 kPa (corresponds to air pressure at 2,000 m above sea level)

3.5.4.2 Valve outputs

		VC/S 6.2.2.2	VC/S 12.2.2.2
Rated values	Number of outputs	6	12
	Non-floating	Yes	Yes
	Rated voltage U_n	230 V AC	230 V AC
	Voltage range	24 … 230 V AC	24 … 230 V AC
	Rated frequency	50/60 Hz	50/60 Hz
	Rated current I_n (per output pair)	0.25 A	0.25 A
	Continuous current at T_u up to 20 °C (per output pair)	0.45 A resistive load	0.45 A resistive load
	Continuous current at T_u up to 45 °C (per output pair)	0.25 A resistive load	0.25 A resistive load
	Inrush current at T_u up to 45°C (per output pair)	≤ 1.3 A (for 10 s)	≤ 1.3 A (for 10 s)
		T_u = ambient temperature	T_u = ambient temperature
	Minimum load (per output)	1 W	1 W
	Overload protection	Yes	Yes
	Short-circuit proof	Yes	Yes

4 Functional overview

4.1 Device functions

The devices have mutually independent semiconductor outputs for activating 2-point valve drives for heating or cooling systems. The outputs can be individually configured and the control values can be obtained direct from an internal controller, or via the bus (ABB i-bus® KNX) from an external controller.

The type of activation, via an internal or external controller, can be specified individually and independently for each output. The settings are made in the *Configuration* parameter window.

Template parameterization can be used to transfer identical settings to other outputs, which significantly reduces configuration work.

4.1.1 Configuring with the internal controller

If using the internal controller, a remote control panel is required to control and operate the room temperature controller. The remote control panel can be an in-room operating control with a remote function (e.g. Busch-Tenton), or a central operating control.

The room temperature controller also requires an external actual temperature. This is measured by a temperature sensor located in the room. The temperature sensor sends its readings to the room temperature controller at regular intervals.

The internal controller processes data (e.g. actual values, setpoints, operating mode changes) received via the bus (ABB i-bus® KNX). The control values of the basic stages are calculated from the received data and transmitted straight to the outputs, which means that no connection via Group Objects is required. In addition, the control values of the basic and additional stages are sent on the bus (ABB i-bus® KNX) via separate Group Objects.

The setpoint temperature of the controller is determined by manual and base setpoint adjustment but also by the following Group Objects (listed in descending order of priority):

- *Operating mode override*
- *Dew point alarm*
- *Window contact*
- *Presence detector*
- *Operating mode (main/secondary)*

Each output can be individually configured with an internal controller. This means that each output can be used to implement individual control strategies, different types of operation and individual activation of valve drives. Alternatively, a single controller can be used for various outputs, which considerably simplifies configuration.

4.1.2 Configuring with an external controller

If using an external controller, the internal controller is not needed. The control values for activating the outputs are specified by an external controller, received via the bus (ABB i-bus® KNX) and transmitted to the outputs.

Each output can be activated individually by different external controllers. This flexibility allows you to seamlessly integrate devices into existing systems and to implement a variety of control strategies.

The control value of the external controller is received as a percentage via the 1-byte Group Object *Control value*.

If a 2-point valve drive is connected to the output and is activated by an open/close signal, the control value can be received as a 1-byte value or via the 1-bit Group Object *Control value* (→ parameter *DPT Group Object "Control value"*).

4.2 Functions

4.2.1 Overview

The ETS application incorporates the following device-specific functions, which can be enabled if required:

→ Safety mode, Page 29

Safety mode ensures that a safe, set control value becomes active if the received control value from the external controller is lost.

→ Temperature acquisition monitoring, Page 29

Monitoring ensures that if the actual temperature on the monitored Group Object fails, the heating or cooling system remains active with an adjustable control value in order to prevent damage due to freezing or overheating.

→ forced operation, Page 30

The function *Forced operation* is used to set the output to a defined state and block it. The function *Forced operation* can be set individually for each output.

→ Function Pump control, Page 30

The function *Pump control* enables up to two individual On/Off telegrams to be sent via the central 1-bit Group Objects *Pump on/off x*.

→ Function Maximum control value, Page 31

The function *Maximum control value* continuously determines the highest control value among those at all the outputs integrated in the function. By adjusting the supply temperature to suit the highest demand, a heating circuit controller (e.g. ABB AC/S 1.x.1) can be used to optimize energy requirements and make heating systems more efficient.

→ Function Summer/winter mode, Page 32

The function *Summer/winter mode* can be used to set different control values for Safety mode (only if using an external controller), forced operation and KNX voltage recovery, based on the time of year.

→ Function Service, Page 31

The function *Service* is used to set the outputs to a defined state and block them. Unlike the function *Forced operation*, the function *Service* applies to all outputs.

→ Function Logic/threshold, Page 33

The function *Logic/threshold* can be used across all devices and independently of other functions. Depending on the device type and variant, up to 16 individually parameterizable Logic or Threshold functions are available; they are enabled in groups of four (→ parameter *Enable groups: Logic/threshold x-y*).

4.2.2 Safety mode

Settings for this are made in the following parameter window:
- Parameter window *Output X:*
  - Parameter window *Monitoring and safety [thermoelectric (PWM)]*

  Or
  - Parameter window *Monitoring and safety [open/close signal]*

Safety mode ensures that a safe, set control value becomes active if the received control value from the external controller is lost.

The device triggers Safety mode when the following conditions are met:
- External controller is used (→ parameter *Output X Controller*)
- Cyclical monitoring is activated (→ parameter *Cyclical monitoring of control value*)
- Error control value receipt

If no value is received on the monitored Group Objects *Control value* (1 byte) or *Control value* (1 bit) during the set time interval (→ parameter *Monitoring time*), the following actions are carried out:
- Group Object *Status Safety mode* is set to "Alarm"
- When using a thermoelectric 2-point valve drive and with the function *Summer/winter mode* deactivated, the control value in the following parameter becomes valid:
  - *Control value in Safety mode*
- When using a thermoelectric 2-point valve drive and with the function *Summer/winter mode* activated, the control values in the following parameters become valid:
  - *Control value in Safety and Summer mode*
  - *Control value in Safety and Winter mode*
- When using a 2-point valve drive that is activated via an open/close signal and with the function *Summer/winter mode* deactivated, the control value in the following parameter becomes valid:
  - *Control value in Safety mode*
- When using a 2-point valve drive that is activated via an open/close signal and with the function *Summer/winter mode* activated, the control values in the following parameters become valid:
  - *Control value in Safety and Summer mode*
  - *Control value in Safety and Winter mode*

The control values apply until a new value is received on the monitored Group Objects.

If the device is in Safety mode when a KNX voltage failure occurs, Safety mode will end on KNX voltage recovery. If no value is received on the monitored Group Objects within the set time interval after KNX voltage recovery, the device switches back to Safety mode.

4.2.3 Temperature acquisition monitoring

Settings for this are made in the following parameter window:
- Parameter window *RTC x* \ Parameter window *Temperature acquisition*

When using the internal controller, the function *Temperature acquisition monitoring* is used to monitor the receipt of an actual temperature on the following Group Object:
- *External actual temperature*

Monitoring ensures that if the actual temperature on the monitored Group Object fails, the heating or cooling system remains active with an adjustable control value in order to prevent damage due to freezing or overheating.

If no value is received on the monitored Group Object during the set time interval (→ parameter *Monitoring time*), the following actions are carried out:

- Group Object *Fault: actual temperature* is set to "Alarm"
- Values in the following parameters become valid:
  - *Type of operation during fault*
  - *Control value during fault*

The internal controller uses the value specified in the parameter *Control value during fault* until a value is received on the Group Object *External actual temperature*. When a value is received, the internal controller changes to standard KNX operation.

4.2.4 forced operation

Settings for this are made in the following parameter window:

- Parameter window *Output X:*
  - Parameter window *Monitoring and safety [thermoelectric (PWM)]*

  Or
  - Parameter window *Monitoring and safety [open/close signal]*

Note
When the function is active, only those commands or functions with a higher priority → Priorities, Page 170 are executed.

The function *Forced operation* is used to set the output to a defined state and block it. The function *Forced operation* can be set individually for each output.

If a forced operation is active, it remains active even after KNX voltage recovery or ETS download.

Forced operation 1 bit

1-bit forced operation is activated or deactivated via the Group Object *Forced operation 1 bit*. When 1-bit forced operation is active, the defined control values or valve positions for 1-bit forced operation are set and the output reacts accordingly.

Forced operation 2 bit

The 2 bit Group Object *Forced operation 2 bit* specifies two states that are set when forced operation is activated. Bit 1 activates or deactivates forced operation. Bit 2 sets the defined state, in this case "OFF" = 0 % (valve closed), "ON" = 100 % (valve open).

Bit 1	Bit 2	State of forced operation
0	0	Forced operation inactive
0	1	Forced operation inactive
1	0	Forced operation active, state "OFF"
1	1	Forced operation active, state "ON"

Tab. 9: Coding of 2-bit forced operation

4.2.5 Function Pump control

Settings for this are made in the following parameter window:

- Parameter window *Device settings* \ Parameter window *Pump control x*

The function *Pump control* enables up to two individual On/Off telegrams to be sent via the central 1-bit Group Objects *Pump on/off x*.

The function *Pump control* is enabled with the parameter *Enable function: Pump control x*. The function *Pump control* applies to all outputs. In the parameter *Output X Pump control x* you can specify individually for each output whether the output should be integrated in the function *Pump control*.

If the device is integrated in a cascade, an external On/Off telegram (→ Cascading, Page 171) can be received via the Group Object *External pump control x*.

The On telegram is sent if one of the following prerequisites is met:
- At least one control value of the integrated outputs exceeds a defined limit value (with hysteresis), → parameters *Limit value* and *Hysteresis*.
- An external On telegram is received via the Group Object *External pump control x*.

The Off telegram is sent if one of the following prerequisites is met:
- All control values of the integrated outputs fall below a defined limit value (with hysteresis), → parameters *Limit value* and *Hysteresis*.
- An external Off telegram is received via the Group Object *External pump control x*.

The parameters *Delay for switching on* and *Switch-off delay* can be used to delay sending the telegram.

- After the set delay for switching on, the On telegram is sent if it is still valid. If an Off telegram is received during the delay for switching on, the On telegram is discarded.
- After the set run-on time, the Off telegram is sent if it is still valid. If an On telegram is received during the run-on time, the Off telegram is discarded.

4.2.6 Function Maximum control value

Settings for this are made in the following parameter window:
- Parameter window *Device settings* \ Parameter window *Maximum control value*

The function *Maximum control value* continuously determines the highest control value among those at all the outputs integrated in the function. By adjusting the supply temperature to suit the highest demand, a heating circuit controller (e.g. ABB AC/S 1.x.1) can be used to optimize energy requirements and make heating systems more efficient.

The function *Maximum control value* is enabled with the parameter *Enable function: Maximum control value*. The function *Maximum control value* applies to all outputs. In the parameter *Output X Maximum control value* you can specify individually for each output whether the output should be integrated in the function *Maximum control value*.

The highest determined control value is sent on the bus (ABB i-bus® KNX) via the Group Object *Status Maximum control value*. The send behavior depends on the setting in the parameter *Send value of Group Object "Status Maximum control value"*.

If the device is integrated in a cascade, an external telegram with the maximum external control value (→ Cascading, Page 171) can be received via the Group Object *Maximum external control value*.

Note
If an output integrated in the function *Maximum control value* is activated by an open/close signal, the determined maximum control value is always 100 %.

4.2.7 Function Service

Settings for this are made in the following parameter window:
- Parameter window *Device settings* \ Parameter window *Service*

Note
When the function is active, only those commands or functions with a higher priority → Priorities, Page 170 are executed.

The function *Service* is used to set the outputs to a defined state and block them. Unlike the function *Forced operation*, the function *Service* applies to all outputs.

In the parameter *Output X Service* you can specify individually for each output whether the output should react to the function *Service*.

The function *Service* is enabled with the parameter *Enable function: Service*. The Group Object *Activate service* activates or ends the function *Service*. In addition, the parameter *End service after* can be used to specify that the function *Service* automatically ends after a parameterizable time. After the function *Service* ends, the refreshed KNX state applies. The reaction of the function *Service* after KNX voltage recovery or ETS download is defined in the parameter *End service after ETS download or KNX voltage recovery*.

The 2 bit Group Object *Activate service* specifies two states that are set when Service is activated. Bit 1 activates or deactivates Service. Bit 2 sets the defined state, in this case "OFF" = 0 % (valve closed), "ON" = 100 % (valve open).

Bit 1	Bit 2	Service state
0	0	Service inactive
0	1	Service inactive
1	0	Service active, state "OFF"
1	1	Service active, state "ON"

Tab. 10: Coding of 2-bit service

The status of the function *Service* (active or inactive) is sent on the bus (ABB i-bus® KNX) via the Group Object *Status Service*.

4.2.8 Function Summer/winter mode

Settings for this are made in the following parameter window:
- Parameter window *Device settings* \ Parameter window *Summer/winter mode*

The function *Summer/winter mode* can be used to set different control values for Safety mode (only if using an external controller), forced operation and KNX voltage recovery, based on the time of year.

The function *Summer/winter mode* applies to all outputs.

The function *Summer/winter mode* is enabled with the parameter *Enable function: Summer/winter mode*. The "summer" or "winter" information is received via the Group Object *Summer/winter mode*.

When the function *Summer/winter mode* is enabled, separate parameters are shown for setting different control values in Summer and Winter mode.

→ Parameter *Set control value after KNX voltage recovery or ETS download*

→ Parameter window *Monitoring and safety [thermoelectric (PWM)]*

→ Parameter window *Monitoring and safety [open/close signal]*

Note
The type of operation on the output (→ parameter *Output X Type of operation*) must match the type of operation on the controller. The settings are not made automatically.

4.2.9 Function Logic/threshold

Settings for this are made in the following parameter window:

- Parameter window *Logic/threshold*
  - Parameter window *Logic/threshold configuration*
  - Parameter window *Logic/Threshold x*

The function *Logic/threshold* can be used across all devices and independently of other functions. Depending on the device type and variant, up to 16 individually parameterizable Logic or Threshold functions are available; they are enabled in groups of four (→ parameter *Enable groups: Logic/threshold x-y*).

4.2.9.1 Function Logic

The function *Logic* calculates a result from two input values. The result depends on the selected logical function and the values in the Input Group Objects.

The available logical functions are:

- AND
- OR
- Exclusive OR
- GATE
- 1 bit Inverter

Two Input Group Objects and one Result Group Object are available in each case for the AND, OR, exclusive OR and GATE logical functions:

- *Connection A*
- *Connection B*
- *Status Result [Logic]*

One Input Group Object and one Result Group Object are available for the 1 bit Inverter:

- *Connection A*
- *Status Result [Logic]*

The result is calculated when a value is received on one of the Input Group Objects.

The result is output on the Group Object *Status Result [Logic]*.

The table below shows the results of the function *Logic* depending on the selected logical function and the values in the Input Group Objects.

Logical function	Connection A	Connection B	Result	Explanation
AND	0	0	0	The result is 1 if each input value is 1.
	0	1	0	
	1	0	0	
	1	1	1	
OR	0	0	0	The result is 1 if at least one of the input values is 1.
	0	1	1	
	1	0	1	
	1	1	1	
Exclusive OR	0	0	0	The result is 1 if an odd number of input values is 1.
	0	1	1	
	1	0	1	
	1	1	0	
GATE	Blocked	0	-	The input value (connection B) is processed only if the GATE is open. The value is ignored if the GATE is blocked.
	Blocked	1	-	
	Open	0	0	
	Open	1	1	
1 bit Inverter	0	-	1	The input value (connection A) is inverted.
	1	-	0	

Tab. 11: Results of the function Logic

4.2.9.2 Function Threshold

The function *Threshold* compares a value received on the threshold input with the thresholds set in the parameters *Upper threshold* and *Lower threshold* and outputs a result.

A result can be defined in the following parameters depending on whether the value on the threshold input is above or below the thresholds:
- *Result if upper threshold is exceeded*
- *Result if lower threshold is dropped below*

One of the "threshold input" Group Objects is used as the threshold input, based on the setting in the parameter *Data point type of Group Object "Threshold input"*. A separate Group Object is available for each data point type.

The following parameters can be used to define how long a threshold must be exceeded before the function outputs a result:
- *Minimum duration of the overshoot*
- *Minimum duration of the undershoot*

The result is output on the Group Object *Status Result [threshold]*.

Monitor range between thresholds

The parameter *Monitor range between thresholds* can be used to define whether the range between the upper and lower thresholds is monitored and an evaluation is output on the following Group Object:
- *Status Input value between thresholds*

The parameter *Minimum dwell time between the thresholds* is used to define how long the value received at the threshold input must be between the thresholds before an evaluation occurs.

Modifying thresholds via the bus

The thresholds set in ETS can be changed via the bus (ABB i-bus® KNX). The setting is made in the parameter *Change thresholds via Group Objects*:

The modified thresholds are received on separate Group Objects via the bus (ABB i-bus® KNX). A separate Group Object is available for each data point type, based on the setting in the parameter *Data point type of Group Object "Threshold input"*.

The parameter *Overwrite thresholds on download* defines whether the thresholds changed via the bus (ABB i-bus® KNX) are overwritten with the thresholds set in ETS during an application download.

4.3 Integration into i-bus® Tool

i-bus® Tool can be used to read the data from connected devices. It can also be used to simulate values and test various functions.

If there is no communication between the devices and i-bus® Tool, the simulated values cannot be sent on the bus.

For more information → parameter *i-bus® Tool access*.

i-bus® Tool can be downloaded free of charge from the company homepage (www.abb.com/knx).

Note
The interface to i-bus® Tool is available from the following software versions:
- Application V1.1 or later

4.3.1 Master-Reset via the i-bus® Tool

The i-bus® Tool function *Master-Reset* can be used to restore a device to the state it was in after the last ETS download. During a Master-Reset, all contacts are opened, all saved values (e.g. travel times, staircase lighting times, counter readings, thresholds) are reset, and the device restarts.

To perform a Master-Reset:
1. Open the "Settings" menu.
2. Enable the function "Master-Reset".
3. Open the "Connect to device" menu.
4. Enter the physical address of the device.
5. Click the *Master-Reset* button.
   ⇨ The Master-Reset is executed; there is no feedback from the device.

Note
A Master-Reset can be done at any time, regardless of the setting in the parameter *i-bus® Tool access*.

4.4 Special operating states

4.4.1 Reaction on KNX voltage failure

KNX voltage failure means the failure of the KNX voltage, e.g. due to a power outage.

In the event of a KNX voltage failure, the outputs are de-energized. Depending on the operating principle of the valve drive, the valves are either opened or closed during a KNX voltage failure.

4.4.2 Reaction after KNX voltage recovery

KNX voltage recovery is the state that exists after the KNX voltage is restored. The device will restart after KNX voltage recovery.

The time set in the following parameter elapses before the device performs an action:
- *Sending and switching delay after KNX voltage recovery*

The parameter *Set control value after KNX voltage recovery or ETS download* can be used to specify an individual control value for each output. The control value applies in the following situations:
- After KNX voltage recovery and after the sending and switching delay.
- After ETS reset.
- After ETS download.

The control value is valid until a new control value is received from the internal controller directly or from the external controller via the bus (ABB i-bus® KNX).

For the internal controller, the parameter *Restore device state after KNX voltage recovery or download* can be used to define whether previous adjustments to the type of operation, operating mode or setpoints are restored after a restart (KNX voltage recovery, ETS reset or ETS download).

Note
To update the Group Objects after KNX voltage recovery, ETS download or ETS reset, the read flags must be set for the corresponding Group Objects of the sending device.

4.4.3 Reaction on ETS reset

ETS reset can be performed in ETS using the Commissioning menu item, in the function *Reset device* (from ETS version 6 *Restart device*).

After an ETS reset, the device restarts and the reaction after a KNX voltage recovery becomes active.

The parameter *Set control value after KNX voltage recovery or ETS download* can be used to specify an individual control value for each output. The control value applies in the following situations:
- After KNX voltage recovery and after the sending and switching delay.
- After ETS reset.
- After ETS download.

The control value is valid until a new control value is received from the internal controller directly or from the external controller via the bus (ABB i-bus® KNX).

For the internal controller, the parameter *Restore device state after KNX voltage recovery or download* can be used to define whether previous adjustments to the type of operation, operating mode or set-points are restored after a restart (KNX voltage recovery, ETS reset or ETS download).

4.4.4 Reaction on ETS download

Note
After the application is uninstalled or the ETS download is canceled, the device will no longer operate and the KNX LED will flash.
- ▸ Repeat the ETS download.

ETS download means loading a modified or updated device application onto the device. During an ETS download, the device will no longer operate and the KNX LED will flash. After the ETS download, the device restarts and the reaction after a KNX voltage recovery becomes active.

The parameter *Set control value after KNX voltage recovery or ETS download* can be used to specify an individual control value for each output. The control value applies in the following situations:
- After KNX voltage recovery and after the sending and switching delay.
- After ETS reset.
- After ETS download.

The control value is valid until a new control value is received from the internal controller directly or from the external controller via the bus (ABB i-bus® KNX).

For the internal controller, the parameter *Restore device state after KNX voltage recovery or download* can be used to define whether previous adjustments to the type of operation, operating mode or set-points are restored after a restart (KNX voltage recovery, ETS reset or ETS download).

For the internal controller, the parameter *Overwrite saved adjustments on download* can be used to define whether the adjustments stored in the device are overwritten during an ETS download with the values specified in ETS.

5 Mounting and installation

5.1 Information about mounting

DANGER – Severe injuries due to touch voltage
Electric feedback from different phase conductors can cause contact voltages and lead to serious injuries.
- Operate the device only in a closed housing.
- Disconnect all phases before working on the electrical connection.

The device can be mounted in any position as required on a 35 mm mounting rail.

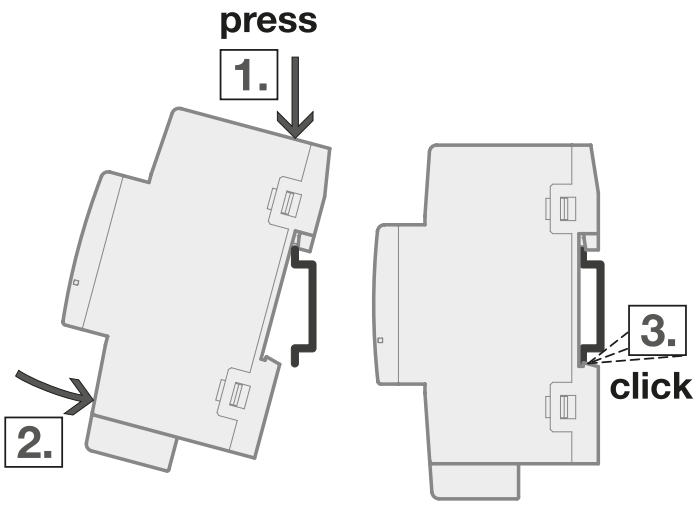
The connection to the bus (ABB i-bus® KNX) is made using the KNX bus connection terminal supplied.

The connections at the inputs or outputs are made via screw terminals
→ terminal designation on the housing.

Note
The maximum permissible current consumption on a KNX line must not be exceeded.
- During planning and installation, ensure that the KNX line is correctly dimensioned. The device has a maximum current consumption of 12 mA.

5.2 Mounting on mounting rail

Note
No additional tools are required for mounting on a mounting rail.



2CDC072013F0015

Fig. 14: Mounting on mounting rail

1. Place the mounting rail holder on the upper edge of the mounting rail and push down.
2. Push the lower part of the device toward the mounting rail until the mounting rail holder engages.
   ⇨ The device is now mounted on the mounting rail.
3. Relieve the pressure on the top of the housing.

6 Commissioning

6.1 Prerequisites for commissioning

A PC with ETS and a connection to the bus (ABB i-bus® KNX), e.g. via a KNX interface, are required to commission the device.

- Required ETS version: 5.7 or higher
- Product-specific device application: installed

Note
See software information on the website → www.abb.com/knx.

Note
In order to commission or program a KNX Data Secure device in Secure mode, the interface used (e.g. USB/S 1.2 or IPS/S 3.x.1) must support "Extended/Long Frames".

6.2 Secure commissioning of KNX Secure devices

Note
ETS version 6 or later is required when using KNX Secure. The use of the latest ETS version is recommended. Using older ETS versions can cause errors during project planning, problems during commissioning (e.g. while replacing devices), or while undertaking diagnostics on group addresses and devices.

Note
In order to commission or program a KNX Data Secure device in Secure mode, the interface used (e.g. USB/S 1.2 or IPS/S 3.x.1) must support "Extended/Long Frames".

To commission the device securely, note the following points:

- A project password must be assigned while importing a KNX Secure device into a KNX project. By assigning the project password, the project is protected against unauthorized access.
  - If a password is not assigned, none of the devices in the project can be operated as KNX Secure devices. This means the security of the whole project will be that of a conventional KNX network (KNX Plain).
  - The project password must be kept in a safe place. Access to the project is not possible without it. Not even the KNX Association or ABB AG will be able to access it.
- A device certificate is required while commissioning a KNX Secure device. This device certificate includes the FDSK (Factory Default Setup Key) and the device's KNX serial number.
  - The device certificate is located on two detachable stickers applied to the device. The stickers should be removed from the device and kept in a safe place.
  - When adding the device from the ETS product catalog, a window opens in ETS, prompting the user to enter the device certificate. The device certificate can be entered using a webcam, a barcode scanner or manually.
  - The device certificates of all KNX Secure devices integrated in the project can be entered in advance in ETS, → Properties/Settings/"Add Device Certificate". Because the device certificate contains the FDSK and the KNX serial number of the device, ETS can automatically manage the assignment of the certificates to the correct device during commissioning.
  - If the device certificate was entered in ETS, the physical address of the device can be assigned via the KNX serial number,
    → Properties/Pending Operations/"Use Device Certificate".
  - The device certificate is only required for initial encryption and authentication of communication between KNX Secure devices and ETS. As it is read, the device certificate is resolved into the FDSK and KNX serial number.
  - During commissioning, ETS assigns a device key (Tool Key) to the device. The device certificate will be required again only if the device is reset to its factory settings (e.g. if the device is to be used as a KNX Secure device in a different system with a different ETS project).

6.2.1 Device certificate

The device certificate is located on two detachable stickers applied to the device. Each sticker contains the following information:

- Device certificate as QR code
- Device certificate as a 36-character combination of numbers and letters

The device certificate is required for securely commissioning the device in a KNX Secure project and contains the following information:

- KNX serial number of the device
- FDSK (Factory Default Setup Key)

For secure commissioning of the device, the device certificate must be entered in ETS (scan the QR code or enter the combination of numbers and letters directly, → Properties/Settings/"Add Device Certificate").

6.3 Commissioning overview

The following factory settings are configured in the state as supplied:

- Physical address of the device: 15.15.255
- Device application: preloaded
- Manual operation: enabled

The device can be programmed only using ETS.

Note

The device application can be re-downloaded if necessary. Downloads may take longer after a device application is uninstalled or when changing applications.

Note
If access to the devices in the project is blocked by a BCU Key, this situation has no effect on this device. Data can still be read and programmed in this device.

6.4 Putting the device into operation

1. Connect the device to the bus (ABB i-bus® KNX).
2. Switch on KNX voltage.
   ⇨ All outputs are de-energized (0 %).

⇨ Device is ready for operation.

Note
Depending on the operating principle of the valve drive, switching on the KNX voltage may cause the valves to open.

6.5 Assignment of the physical address

Triggering assignment of the physical address via ETS:

1. Press the *Programming* button.
   ⇨ Programming mode active. The *Programming* LED lights up.
2. Start programming process in ETS.

⇨ Physical address is assigned. Device executes an ETS reset.

Note
If a KNX Secure device was securely commissioned, the KNX serial number of the device can be used to assign the physical address, → Secure commissioning of KNX Secure devices, Page 38.

6.6 Software/device application

6.6.1 Device applications

The following device applications are available for the devices described in this document:

Device type	Device application	Max. number of group addresses	Max. number of secure group addresses	Max. number of secure partners
VC/S 6.2.2.2	Valve Drive Controller, 6f/ …	2000	1328	500
VC/S 12.2.2.2	Valve Drive Controller, 12f/ …	2000	1328	500

Tab. 12: Device applications

Note
… = current version number of the application.
See software information on the website, → www.abb.com/knx.

6.6.2 Copying, exchanging and converting

The following functions can be performed with the ETS app *ABB Update Copy Convert*:
- *Update*: Changes the device application to a higher or lower version while retaining the current configurations
- *Convert*: Adopts a configuration from an identical or compatible source device
- *Copy channel*: Copies a channel configuration to other channels on a multichannel device
- *Channel exchange*: Exchanges configurations between two channels on a multichannel device
- *Import/export*: Saves and reads device configurations as external files

The ETS app *ABB Update Copy Convert* can be downloaded free of charge from the KNX Shop → www.KNX.org.

6.7 Unloading device or resetting to factory settings (Master-Reset)

6.7.1 Resetting device to factory settings using the Programming button

1. Disconnect the device from the bus (ABB i-bus® KNX).
2. Wait 5 seconds, then press and hold the *Programming* button.
3. Connect the device to the bus (ABB i-bus® KNX).
   ⇨ *Programming* LED flashes at 1 Hz.
4. When the *Programming* LED flashes at 5 Hz, release the *Programming* button.

⇨ Device executes a Master-Reset; *Programming* LED is off.

⇨ Settings and physical addresses are reset to their factory settings.

⇨ The device application is removed.

⇨ The firmware version is retained.

⇨ The device key (Tool Key) assigned by ETS is reset. The device certificate is required for recommissioning if it is not still available in the ETS project from the original commissioning.

Note
The device application can be re-downloaded if necessary. Downloads may take longer after a device application is uninstalled or when changing applications.

6.7.2 Unloading device via ETS

Note
A KNX Secure device can only be unloaded via ETS if the device certificate has been entered in the project and ETS has assigned a device key (Tool Key).

Unloading application
- The physical address is retained.
- The device application is removed.
- The device key assigned by ETS (Tool Key) is retained. The device certificate is not required for reprogramming.

Unloading physical address and application

- Settings and physical addresses are reset to their factory settings.
- The device application is removed.
- The firmware version is retained.
- The device key (Tool Key) assigned by ETS is reset. The device certificate is required for recommissioning if it is not still available in the ETS project from the original commissioning.

Note

The device application can be re-downloaded if necessary. Downloads may take longer after a device application is uninstalled or when changing applications.

7 Parameters

7.1 General

Note
ETS (Engineering Tool Software) is used to parameterize the device.

The following sections describe the device parameters based on the parameter windows. The parameter windows have a dynamic design. Parameters are shown or hidden depending on parameterization and function.

The default values for the parameters are underlined, e.g.:
no (checkbox cleared)
yes (checkbox ticked)

Note
The default values in the device application can vary from the values stated in the product manual depending on the product variant.

Note
The largest and most extensive device in the product family is described below as an example.

7.1.1 Prerequisites for visibility

In the "Prerequisites for visibility" the ETS settings and product variants necessary to display a parameter window/parameter/Group Object are listed. If no "Prerequisites for visibility" are specified, parameter windows/parameters/Group Objects are always shown or the prerequisites are given by the higher-level parameter window.

The "Prerequisites for visibility" are structured as follows:
- Parameter windows: all necessary prerequisites
- Parameters: Settings in other parameter windows, higher-level parameters, product variant required
- Group Objects: all necessary prerequisites

7.2 Parameter windows

7.2.1 Parameter windows Configuration

The following settings can be made in this parameter window:

- Specify valve type and basic configuration
- Enable controller and specify basic configuration
- Enable functions *Logic* and *Threshold*
- Enable device-specific functions

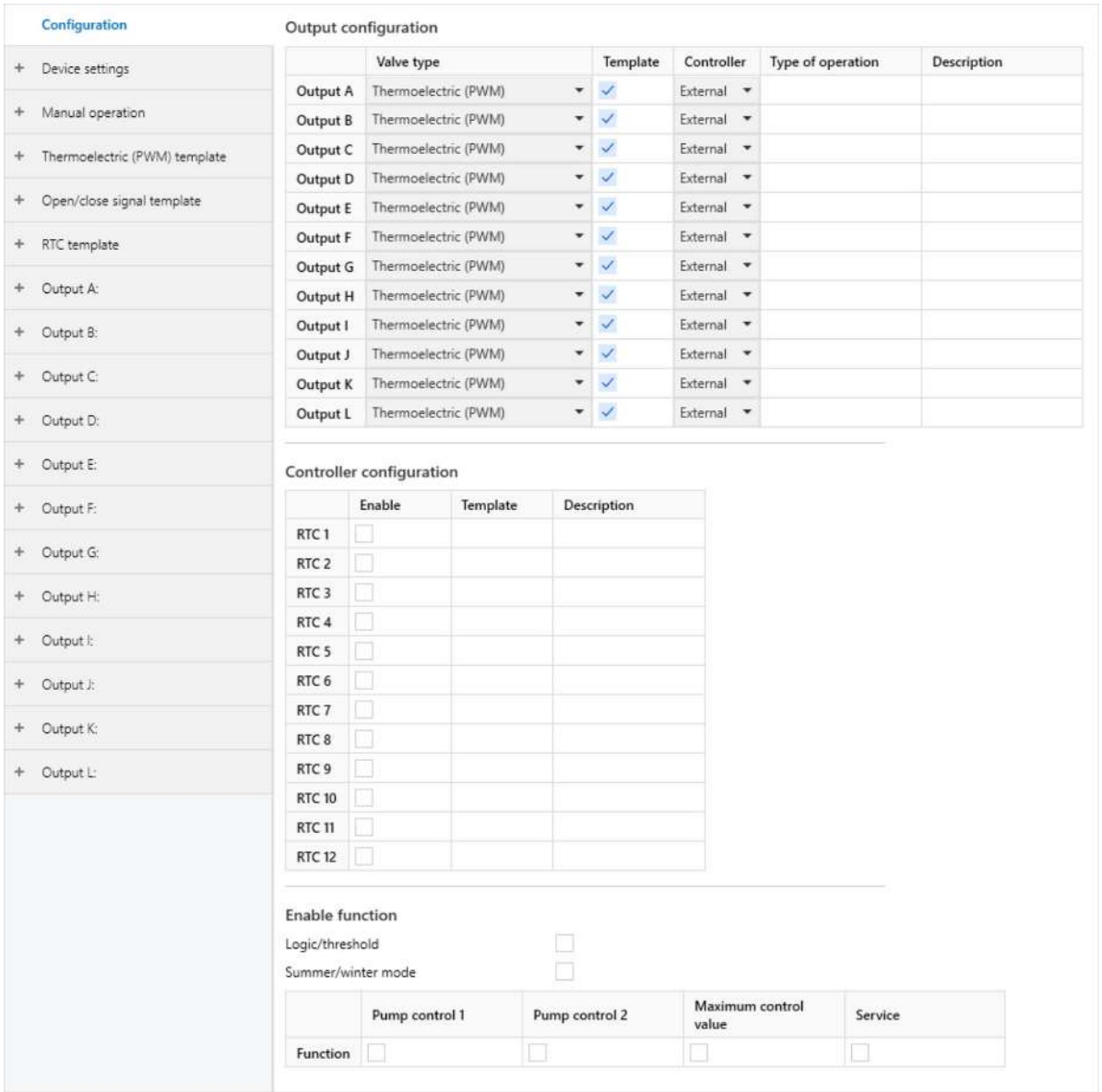


Fig. 15: Configuration parameter window

This parameter window includes the following parameters:

7.2.1.1 Output X Valve type

This parameter is used to set which type of valve is connected to the output.

Option	
Thermoelectric (PWM)	The following dependent parameter windows are displayed: • *Output X:* • *Thermoelectric (PWM)* • *Monitoring and safety [thermoelectric (PWM)]* The following dependent parameters are displayed: • *Output X Template* • *Output X Controller* • *Output X description*
Open/close signal	The following dependent parameter windows are displayed: • *Output X:* • *Open/close signal* • *Monitoring and safety [open/close signal]* The following dependent parameters are displayed: • *Output X Template* • *Output X Controller* • *Output X description*
Deactivated	The output is deactivated.

7.2.1.2 Output X Template

This parameter is used to define whether the settings for the output are adopted from the template.

Option	
No	The parameters can be set individually.
Yes	The settings for the parameters are adopted from the template. The parameterization can only be done in the template; the parameters are not visible in any other parameter windows.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*

7.2.1.3 Output X Controller

This parameter is used to define the controller configuration that is used for the output.

Option	
RTC x	The device-internal controller configuration x is used. The control values of the basic-stages *Heating/Cooling* (depending on the setting in the parameter *Output X Type of operation*) are output on the output. In addition, the control values of all stages can be sent on the bus (ABB i-bus® KNX) via separate Group Objects.
External	An external controller is used. The control value of the output is calculated by the external controller and received via a Group Object. Configuration (type of operation, type of heating/cooling system, etc.) must be performed in the external controller. Depending on the setting in the parameter *Output X Valve type*, one of the following Group Objects will be shown: • *Control value* (1 byte) • *Control value* (1 bit)

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*

7.2.1.4 Output X Type of operation

This parameter is used to define the type of operation on the output.

Note

The output only reacts to the control values for the type of operation defined. The type of operation on the output must match the type of operation on the controller. The settings are not made automatically.

Option	
Heating	The output is configured for the type of operation *Heating*. The output only outputs the control values for the type of operation *Heating*.
Cooling	The output is configured for the type of operation *Cooling*. The output only outputs the control values for the type of operation *Cooling*.
Heating and cooling	The output is configured for the types of operation *Heating* and *Cooling*. The output outputs the control values for both types of operation.

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ all options except *Deactivated*
  - Parameter *Output X Controller* \ Option *RTC x*

7.2.1.5 Output X description

This parameter is used to define an individual description for a channel, an input or an output. The description is displayed at the following points:

- In the name of the corresponding parameter window
- In the name of the corresponding Group Objects

Option	
Free text entry	Maximum 24 ASCII characters; the maximum number of characters may vary for other character formats.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*

7.2.1.6 Enable RTC x

Depending on the device variant, this parameter enables up to 12 mutually independent room temperature controllers.

Option	
No	The room temperature controller is not enabled.
Yes	The following dependent parameter windows are displayed: • *RTC x* The following dependent parameters are displayed: • *RTC x Template* • *RTC x Description* The following dependent Group Objects are displayed: • *External actual temperature* • *Current setpoint (main/secondary)* • *Request setpoint adjustment (main/secondary)* • *Operating mode (main/secondary)* • *Current operating mode (main/secondary)*

7.2.1.7 RTC x Template

This parameter is used to define whether the settings for the room temperature controller are adopted from the template.

Option	
No	The parameters can be set individually.
Yes	The settings for the parameters are adopted from the template. The parameterization can only be done in the template; the parameters are not visible in any other parameter windows.

Prerequisites for visibility
Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*

7.2.1.8 RTC x Description

This parameter is used to specify an individual description for the room temperature controller. The description is displayed in the following places:
- In the name of the corresponding parameter window
- In the name of the corresponding Group Objects

Option	
Free text entry	Maximum 24 ASCII characters; the maximum number of characters may vary for other character formats.

Prerequisites for visibility
Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*

7.2.1.9 Enable function: Logic/threshold

This parameter enables the functions *Logic* and *Threshold*.

More information: → Function Logic/threshold, Page 33.

Option	
No	The functions *Logic* and *Threshold* are not enabled.
Yes	The following dependent parameter windows are displayed: • *Logic/threshold* • *Logic/threshold configuration*

7.2.1.10 Enable function: Summer/winter mode

This parameter enables the function *Summer/winter mode*.

More information: → Function Summer/winter mode, Page 32.

Option	
No	The function is not enabled.
Yes	The following dependent parameter windows are displayed: • *Device settings* \ *Summer/winter mode* The following dependent Group Objects are displayed: • *Summer/winter mode*

7.2.1.11 Enable function: Pump control x

This parameter enables the function *Pump control x*.

More information: → Function Pump control, Page 30.

Option	
No	The function is not enabled.
Yes	The following dependent parameter windows are displayed: • *Device settings* \ *Pump control x* The following dependent Group Objects are displayed: • *Pump on/off x*

7.2.1.12 Output X Pump control x

This parameter is used to activate the use of the function *Pump control x* for the output.

More information: → Function Pump control, Page 30.

Option
No
Yes

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Pump control x* \ Option *Yes*

7.2.1.13 Enable function: Maximum control value

This parameter enables the function *Maximum control value*.

More information: → Function Maximum control value, Page 31.

Option	
No	The function is not enabled.
Yes	The following dependent parameter windows are displayed: • *Device settings* \ *Maximum control value* The following dependent Group Objects are displayed: • *Status Maximum control value*

7.2.1.14 Output X Maximum control value

This parameter is used to activate the use of the function *Maximum control value* for the output.

More information: → Function Maximum control value, Page 31.

> **Note**
> If an output integrated in the function *Maximum control value* is activated by an open/close signal, the determined maximum control value is always 100 %.

Option
No
Yes

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Maximum control value* \ Option *Yes*

7.2.1.15 Enable function: Service

This parameter enables the function *Service*.

More information: → Function Service, Page 31.

Option	
No	The function is not enabled.
Yes	The following dependent parameter windows are displayed: • *Device settings* \ *Service* The following dependent Group Objects are displayed: • *Activate service* • *Status Service*

7.2.1.16 Output X Service

This parameter is used to activate the use of the function *Service* for the output.

More information: → Function Service, Page 31.

Option
No
Yes

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Service* \ Option *Yes*

7.2.2 Parameter windows Device settings

7.2.2.1 Parameter windows Device settings

The following settings can be made in this parameter window:

- Set sending and switching delay
- Set telegram rate limit
- Enable central and device-specific Group Objects

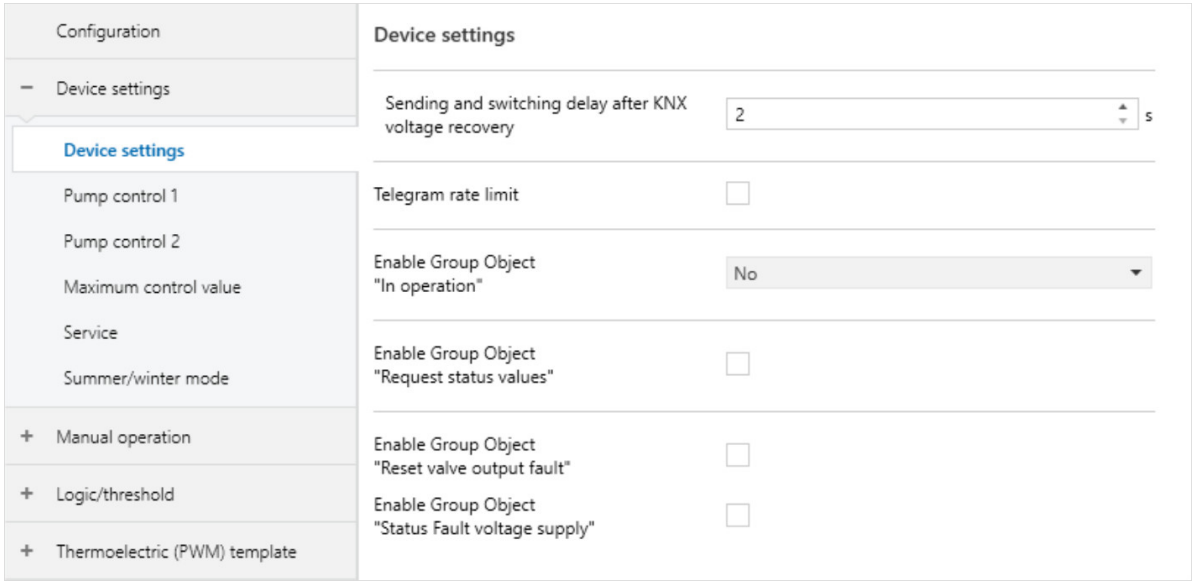


Fig. 16: Device Settings parameter window

This parameter window includes the following parameters:

→ Sending and switching delay after KNX voltage recovery, Page 50
→ Telegram rate limit, Page 51
→ Maximum number of sent telegrams, Page 51
→ In period, Page 51
→ i-bus® Tool access, Page 52
→ Enable Group Object "In operation", Page 52
→ Sending cycle, Page 52
→ Enable Group Object "Request status values", Page 52
→ Enable Group Object "Reset valve output fault", Page 53
→ Enable Group Object "Status Fault supply voltage", Page 53
→ Value of Group Object "Status Fault supply voltage", Page 53

7.2.2.1.1 Sending and switching delay after KNX voltage recovery

This parameter is used to define the sending and switching delay after KNX voltage recovery.

More information: → Sending or switching delay, Page 174.

Note
After KNX voltage recovery, the device waits for the sending delay time to elapse before sending telegrams on the bus (ABB i-bus® KNX).

Option
2 ... 60 s

7.2.2.1.2 Telegram rate limit

This parameter is used to define whether the number of telegrams sent by the device will be limited. The fewer telegrams sent per unit of time, the lower the bus load will be.

More information: → Telegram rate limit, Page 175.

Option	
No	The number of telegrams is not limited.
Yes	The number of telegrams is limited. The following dependent parameters are displayed: • *Maximum number of sent telegrams* • *In period*

7.2.2.1.3 Maximum number of sent telegrams

This parameter is used to define the number of telegrams sent within a period that can be set.

The period is defined in the parameter *In period*.

More information: → Telegram rate limit, Page 175.

Option
0 … 20 … 100

Prerequisites for visibility

- Parameter window *Device settings* \ Parameter *Telegram rate limit* \ Option *Yes*

7.2.2.1.4 In period

This parameter is used to define the period during which the device sends telegrams. The telegrams are sent as quickly as possible at the start of a period.

More information: → Telegram rate limit, Page 175.

Note
The telegram rate limit is deactivated when the value 0 is selected.

Option
0 … 1 … 59 s

Prerequisites for visibility

- Parameter window *Device settings* \ Parameter *Telegram rate limit* \ Option *Yes*

7.2.2.1.5 i-bus® Tool access

This parameter is used to define whether the device can be accessed via i-bus® Tool.

More information: → Integration into i-bus® Tool, Page 34.

Note
The interface to i-bus® Tool is available from the following software versions:
- Application V1.1 or later

Option	
No access	Access via the i-bus® Tool is deactivated. A Master-Reset via the i-bus® Tool is possible.
Read only	Values can be displayed via the i-bus® Tool. A Master-Reset via the i-bus® Tool is possible.
Full access	Values can be displayed and changed via the i-bus® Tool. A Master-Reset via the i-bus® Tool is possible.

7.2.2.1.6 Enable Group Object "In operation"

This parameter enables the Group Object *In operation*.

Option	
No	The Group Object is not enabled.
Yes, send value 0 cyclically	The Group Object is enabled and cyclically sends the value 0. The following dependent parameters are displayed: • *Sending cycle* The following dependent Group Objects are displayed: • *In operation*
Yes, send value 1 cyclically	The Group Object is enabled and cyclically sends the value 1. The following dependent parameters are displayed: • *Sending cycle* The following dependent Group Objects are displayed: • *In operation*

7.2.2.1.7 Sending cycle

This parameter is used to define the cycle in which the Group Object *In operation* sends a telegram.

Option
00:00:01 … 00:10:00 … 18:12:15 hh:mm:ss

Prerequisites for visibility
- Parameter window *Device settings* \ Parameter *Enable Group Object "In operation"* \ Option *Yes, send value 0 cyclically* / *Yes, send value 1 cyclically*

7.2.2.1.8 Enable Group Object "Request status values"

This parameter enables the following Group Object:
- *Request status values*

All status messages of the device can be requested using the Group Object *Request status values* and sent on the bus (ABB i-bus® KNX).

Option	
No	The Group Object is not enabled.
Yes	The Group Object is enabled. The following dependent Group Objects are displayed: • *Request status values*

7.2.2.1.9 Enable Group Object "Reset valve output fault"

This parameter enables the following Group Object:
- *Reset valve output fault*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Reset valve output fault*

7.2.2.1.10 Enable Group Object "Status Fault supply voltage"

This parameter enables the following Group Object:
- *Status Fault supply voltage*

Note
The Group Object *Status Fault supply voltage* reports a fault if there is no supply voltage (VC/S 6.2.2.2: terminal pair 1/2, VC/S 12.2.2.2: terminal pairs 1/2 and 15/16). If the upper outputs (valve outputs A ... F) or the lower outputs (valve outputs G ... L) are not in use on the VC/S 12.2.2.2 and the corresponding supply voltage is not connected, the corresponding outputs should be deactivated (→ parameter *Output X Valve type*) to avoid unwanted fault messages.

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Status Fault supply voltage*

7.2.2.1.11 Value of Group Object "Status Fault supply voltage"

This parameter is used to define the telegram value that the Group Object *Status Fault supply voltage* sends.

Option
1: Fault, 0: No fault
0: Fault, 1: No fault

Prerequisites for visibility
- Parameter window *Device settings* \ Parameter window *Device settings* \ Parameter *Enable Group Object "Status Fault supply voltage"* \ Option *Yes*

7.2.2.2 Parameter windows Pump control x

The following settings can be made in this parameter window:
- Set the function *Pump control*

> **Note**
> The device can be used to individually activate up to two pumps (or devices that function in the same way, e.g. heat pumps)

More information: → Function Pump control, Page 30.

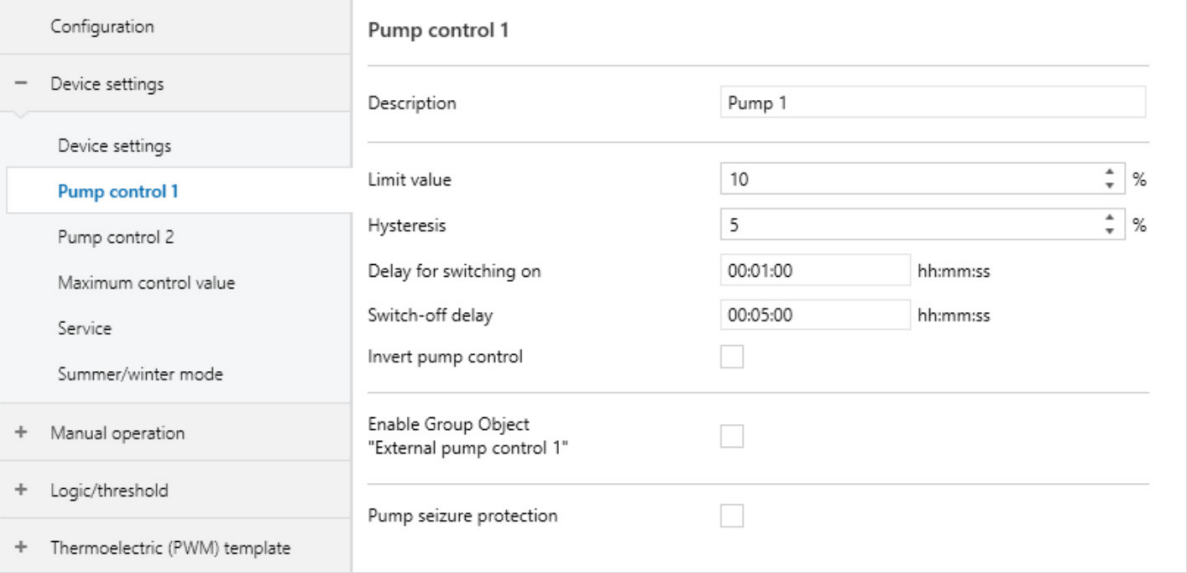


Fig. 17: Parameter window Pump control x

This parameter window includes the following parameters:
→ Description, Page 54
→ Limit value, Page 55
→ Hysteresis, Page 55
→ Delay for switching on, Page 55
→ Switch-off delay, Page 55
→ Invert pump control, Page 55
→ Enable Group Object "External pump control x", Page 55
→ Pump seizure protection, Page 56
  → Pump seizure protection duration, Page 56
  → Pump seizure protection cycle, Page 56

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Pump control x* \ Option *Yes*

7.2.2.2.1 Description

This parameter is used to specify an individual description for the following Group Object:
- *Pump on/off x*

The description is shown in the name of the corresponding Group Object.

Option	
Free text entry	Maximum 24 ASCII characters; the maximum number of characters may vary for other character formats.

7.2.2.2.2 Limit value

This parameter is used to define a limit value for sending the On/Off telegram.

More information: → Function Pump control, Page 30.

Option
1 ... 5 ... 100 %

7.2.2.2.3 Hysteresis

This parameter is used to define a double-sided hysteresis for the limit value (→ parameter *Limit value*).

More information:
→ Hysteresis, Page 170
→ Function Pump control, Page 30

Option
1 ... 5 ... 20 %

7.2.2.2.4 Delay for switching on

This parameter is used to define a delay for switching on, for sending the On telegram.

More information: → Function Pump control, Page 30.

Option
00:00:00 ... 00:01:00 ... 18:12:15 hh:mm:ss

7.2.2.2.5 Switch-off delay

This parameter is used to define the run-on time (delay for switching off) for sending the Off telegram.

More information: → Function Pump control, Page 30.

Option
00:00:00 ... 00:05:00 ... 18:12:15 hh:mm:ss

7.2.2.2.6 Invert pump control

This parameter is used to define the telegram value that the Group Object *Pump on/off x* sends.

Option	
No	1 = On, 0 = Off
Yes	0 = On, 1 = Off

7.2.2.2.7 Enable Group Object "External pump control x"

This parameter enables the following Group Object:
- *External pump control x*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *External pump control x*

7.2.2.2.8 Pump seizure protection

This parameter is used to activate pump seizure protection.

More information: → Pump seizure protection, Page 171.

Option	
No	Pump seizure protection is not activated.
Yes	The following dependent parameters are displayed: • *Pump seizure protection duration* • *Pump seizure protection cycle*

7.2.2.2.9 Pump seizure protection duration

This parameter is used to define the pump seizure protection duration.

More information: → Pump seizure protection, Page 171.

Option
1 ... 4 ... 59 min

Prerequisites for visibility
Parameter window *Device settings* \ Parameter window *Pump control x* \ Parameter *Pump seizure protection* \ Option *Yes*

7.2.2.2.10 Pump seizure protection cycle

This parameter is used to define the pump seizure protection cycle.

More information: → Pump seizure protection, Page 171.

Option
1 ... 4 ... 52 weeks

Prerequisites for visibility
Parameter window *Device settings* \ Parameter window *Pump control x* \ Parameter *Pump seizure protection* \ Option *Yes*

7.2.2.3 Parameter windows Maximum control value

The following settings can be made in this parameter window:
- Set the function *Maximum control value*

More information: → Function Maximum control value, Page 31.

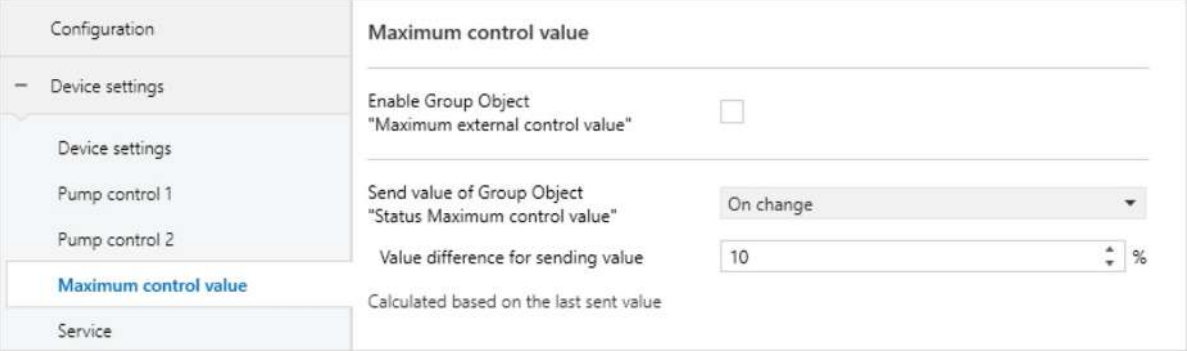


Fig. 18: Parameter window Maximum control value

This parameter window includes the following parameters:
→ Enable Group Object "Maximum external control value", Page 57
→ Send value of Group Object "Status Maximum control value", Page 57
→ Sending cycle, Page 58
→ Value is sent from a change of, Page 58

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Maximum control value* \ Option *Yes*

7.2.2.3.1 Enable Group Object "Maximum external control value"

This parameter enables the following Group Object:
- *Maximum external control value*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Maximum external control value*

7.2.2.3.2 Send value of Group Object "Status Maximum control value"

This parameter is used to define when the value of the following Group Object is sent on the bus (ABB i-bus® KNX):
- *Status Maximum control value*

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent if there is a change. The following dependent parameters are displayed: • *Value is sent from a change of*
Cyclically	The value is sent cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent. The following dependent parameters are displayed: • *Sending cycle*
On change or cyclically	The value is sent on change or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change. The following dependent parameters are displayed: • *Value is sent from a change of* • *Sending cycle*
On request	The value is sent on request.
On change or on request	The value is sent on change or on request. The following dependent parameters are displayed: • *Value is sent from a change of*
On request or cyclically	The value is sent on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on request. The following dependent parameters are displayed: • *Sending cycle*
On change, on request or cyclically	The value is sent on change, on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change or on request. The following dependent parameters are displayed: • *Value is sent from a change of* • *Sending cycle*

7.2.2.3.3 Sending cycle

This parameter is used to define the cycle in which the Group Object value is sent.

Note
The possible options and default values depend on the higher-level parameter.

Option
00:00:30 … 01:00:00 … 24:00:00 hh:mm:ss
00:01 … 00:05 … 23:59 hh:mm
1 … 15 … 240 min

Prerequisites for visibility

- The parameter appears at various points in the application. The visibility is dependent on the application and the higher-level parameter.

7.2.2.3.4 Value is sent from a change of

This parameter is used to define the minimum change in the input value (based on the last value sent) required for the output value to be sent on the bus (ABB i-bus® KNX).

Note
The possible options and default values depend on the higher-level parameter.

Option
1 … 5 … 100 %

Prerequisites for visibility

- The visibility is dependent on the application and the higher-level parameter.

7.2.2.4 Parameter windows Service

The following settings can be made in this parameter window:
- Set the function *Service*

More information: → Function Service, Page 31.

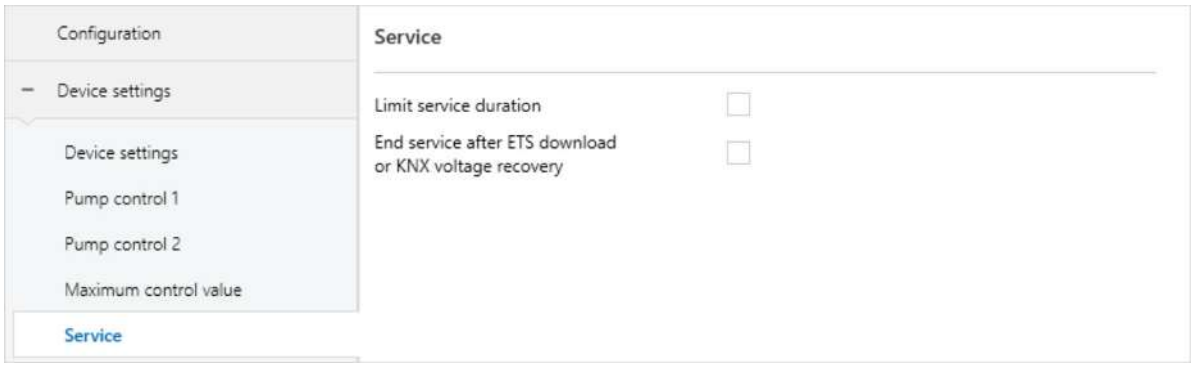


Fig. 19: Parameter window Service

This parameter window includes the following parameters:
→ Limit service duration, Page 59
→ End service after, Page 59
→ End service after ETS download or KNX voltage recovery, Page 59

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Service* \ Option *Yes*

7.2.2.4.1 Limit service duration

This parameter is used to define whether the service duration is limited and ends automatically.

Option	
No	The service duration is unlimited. The service can be ended only via Group Object *Activate service*.
Yes	The service ends automatically after a set time. The following dependent parameters are displayed: • *End service after*

7.2.2.4.2 End service after

This parameter is used to define the time after which the service automatically ends.

Option
00:00:30 … 01:00:00 … 99:59:59 hh:mm:ss

7.2.2.4.3 End service after ETS download or KNX voltage recovery

This parameter is used to define whether the function *Service* ends after KNX voltage recovery or ETS download.

Option
No
Yes

7.2.2.5 Parameter windows Summer/winter mode

The following settings can be made in this parameter window:
- Set the function *Summer/winter mode*

More information: → Function Summer/winter mode, Page 32.

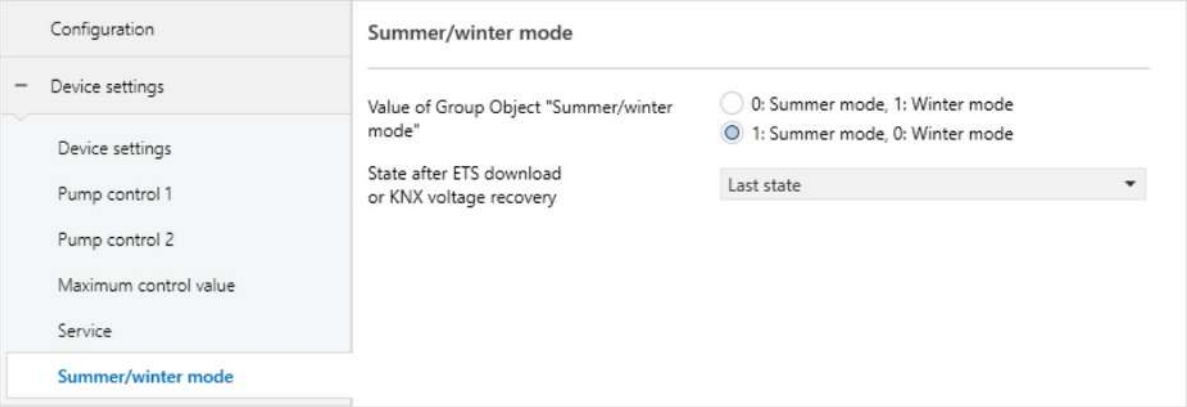


Fig. 20: Parameter window Summer/winter mode

This parameter window includes the following parameters:
→ Value of Group Object "Summer/winter mode", Page 60
→ State after ETS download or KNX voltage recovery, Page 60

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*

7.2.2.5.1 Value of Group Object "Summer/winter mode"

This parameter is used to define the telegram value from the Group Object *Summer/winter mode* that activates Summer or Winter mode.

Option	
0: Summer mode, 1: Winter mode	
1: Summer mode, 0: Winter mode	

7.2.2.5.2 State after ETS download or KNX voltage recovery

This parameter is used to define the state of the function *Summer/winter mode* after ETS download or KNX voltage recovery.

Option	
Last state	The last known state (Summer or Winter mode) is set.
Winter mode	Winter mode is set.
Summer mode	Summer mode is set.

7.2.3 Parameter windows Manual operation

The following settings can be made in this parameter window:
- Enable operating state *Manual operation*
- Automatically reset the device to operating state *KNX operation*

More information: → Manual operation, Page 166.

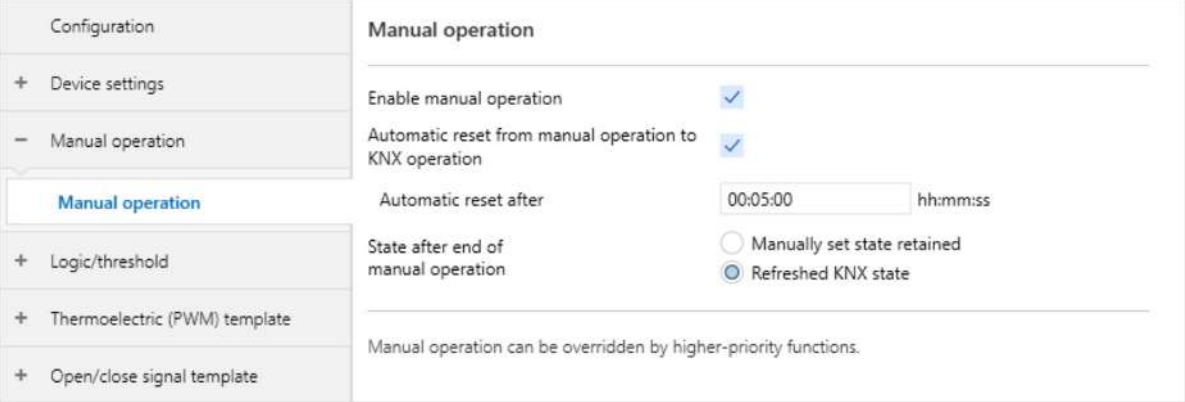


Fig. 21: Parameter window Manual operation

This parameter window includes the following parameters:

→ Enable manual operation, Page 61
→ Automatic reset from manual operation to KNX operation, Page 61
→ Automatic reset after, Page 62
→ State after deactivating manual operation, Page 62

7.2.3.1 Enable manual operation

This parameter is used to enable the manual operation of the device.

Option	
No	The manual operation of the device is not enabled.
Yes	The following dependent parameters are displayed: • *Automatic reset from manual operation to KNX operation* • *State after deactivating manual operation* The following dependent Group Objects are displayed: • *Enable/Block Manual operation* • *Deactivate manual operation* • *Status Manual operation*

7.2.3.2 Automatic reset from manual operation to KNX operation

This parameter is used to define whether the device is reset from the operating state *Manual operation* to the operating state *KNX operation* after an adjustable time.

Option	
No	Automatic reset is deactivated.
Yes	The following dependent parameters are displayed: • *Automatic reset after*

Prerequisites for visibility
- Parameter window *Manual operation* \ Parameter *Enable manual operation* \ Option *Yes*

7.2.3.3 Automatic reset after

This parameter is used to define the time after which the device is automatically reset to the operating state *KNX operation*.

After the *Manual operation* button is pressed, the device remains in the operating state *Manual operation* until the button is pressed again or the set time expires.

Option
00:00:30 … 00:05:00 … 18:12:15 hh:mm:ss

Prerequisites for visibility

- Parameter window *Manual operation* \ Parameter *Automatic reset from manual operation to KNX operation* \ Option *Yes*

7.2.3.4 State after deactivating manual operation

This parameter is used to define the state of the outputs after deactivating manual operation.

Option	
Manually set state retained	The manually set state remains active.
Refreshed KNX state	The manually set state is overwritten. The refreshed KNX state is used, → Refreshed KNX state, Page 174.

Prerequisites for visibility

- Parameter window *Manual operation* \ Parameter *Enable manual operation* \ Option *Yes*

7.2.4 Parameter windows Logic/threshold

The functions *Logic* and *Threshold* can be set individually for each output in the subordinate parameter windows.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*

7.2.4.1 Parameter windows Logic/threshold configuration

The following settings can be made in this parameter window:
- Enable functions *Logic* and *Threshold* in groups of four

The functions *Logic* and *Threshold* can be used as standalone functions.

More information: → Function Logic/threshold, Page 33.

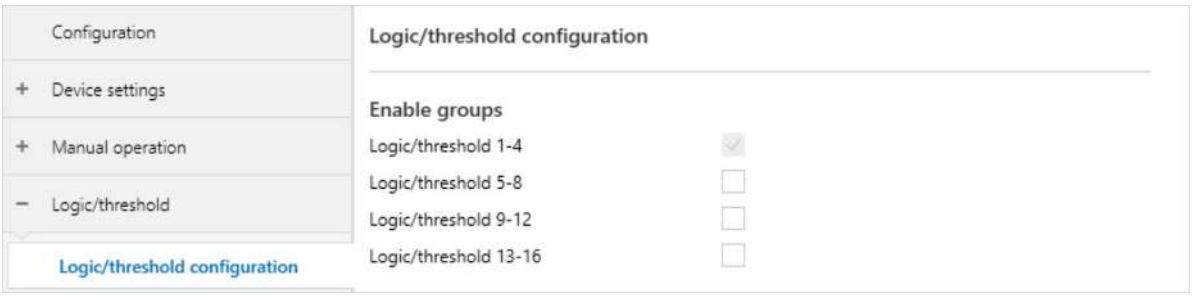


Fig. 22: Parameter window Logic/threshold configuration

This parameter window includes the following parameters:
→ Enable groups: Logic/threshold x-y, Page 64

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*

7.2.4.1.1 Enable groups: Logic/threshold x-y

This parameter enables the functions *Logic* and *Threshold* in groups of four.

The functions *Logic* and *Threshold* are configured in the parameter window *Logic/Threshold x*.

Option	
No	The logic/threshold groups are not enabled
Yes	The following dependent parameter windows are displayed: • *Logic/Threshold x*

7.2.4.2 Parameter windows Logic/Threshold x

The following settings can be made in this parameter window:
- Parameterize function *Logic*
- Parameterize function *Threshold*

The functions *Logic* and *Threshold* can be used independent of other functions. The results of the functions *Logic* and *Threshold* can be sent on the bus (ABB i-bus® KNX).

More information: → Function Logic/threshold, Page 33.

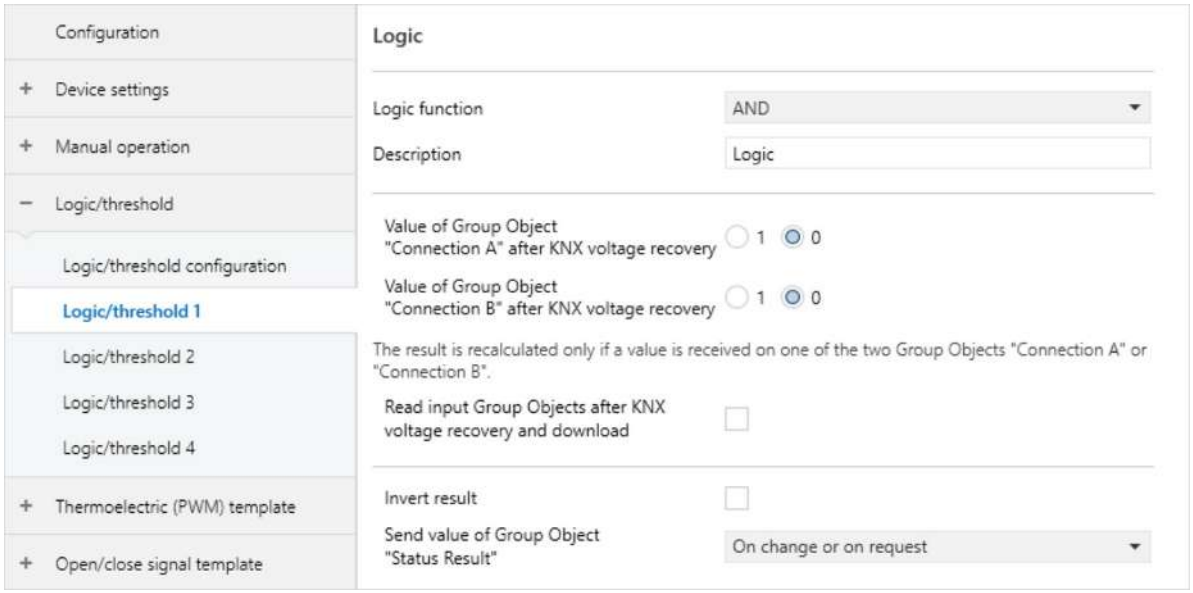


Fig. 23: Parameter window Logic/threshold x

This parameter window includes the following parameters:

→ Logic function, Page 66
- → GATE blocks if Group Object "Connection A" equals, Page 68
- → Value of Group Object "Connection A" after KNX voltage recovery, Page 68
- → Value of Group Object "Connection B" after KNX voltage recovery, Page 69
- → Read input Group Objects after KNX voltage recovery or download, Page 69
- → Invert result, Page 69
- → Send value of Group Object "Status Result", Page 69
- → Data point type of Group Object "Threshold input", Page 70
- → Upper threshold, Page 70
- → Lower threshold, Page 71
- → Change thresholds via Group Objects, Page 72
- → Change thresholds via i-bus® Tool, Page 72
  - → Overwrite thresholds on download, Page 73
- → Result if upper threshold is exceeded, Page 73
- → Minimum duration of the overshoot, Page 73
- → Monitor range between thresholds, Page 73
  - → Minimum dwell time between the thresholds, Page 74
- → Result if lower threshold is dropped below, Page 74
- → Minimum duration of the undershoot, Page 74
- → Read input Group Objects after KNX voltage recovery or download, Page 74
- → Send value of Group Objects "Status Result", "Status Input value between thresholds", Page 75

Prerequisites for visibility
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*

7.2.4.2.1 Logic function

This parameter is used to define whether one of the logic functions or the threshold function is used.

More information: → Function Logic/threshold, Page 33.

Option	
None	The logic function is not used.
AND	The logic function *AND* is used. The result is 1 if each input value is 1. The following dependent parameters are displayed: • *Description* • *Value of Group Object "Connection A" after KNX voltage recovery* • *Value of Group Object "Connection B" after KNX voltage recovery* • *Read input Group Objects after KNX voltage recovery or download* • *Invert result* • *Send value of Group Object "Status Result"* The following dependent Group Objects are displayed: • *Connection A* • *Connection B* • *Status Result [Logic]*
OR	The logic function *OR* is used. The result is 1 if at least one of the input values is 1. The following dependent parameters are displayed: • *Description* • *Value of Group Object "Connection A" after KNX voltage recovery* • *Value of Group Object "Connection B" after KNX voltage recovery* • *Read input Group Objects after KNX voltage recovery or download* • *Invert result* • *Send value of Group Object "Status Result"* The following dependent Group Objects are displayed: • *Connection A* • *Connection B* • *Status Result [Logic]*
Exclusive OR	The logic function *exclusive OR* is used. The result is 1 if an odd number of input values is 1. The following dependent parameters are displayed: • *Description* • *Value of Group Object "Connection A" after KNX voltage recovery* • *Value of Group Object "Connection B" after KNX voltage recovery* • *Read input Group Objects after KNX voltage recovery or download* • *Invert result* • *Send value of Group Object "Status Result"* The following dependent Group Objects are displayed: • *Connection A* • *Connection B* • *Status Result [Logic]*
GATE	The logic function *GATE* is used. If the GATE is open (connection A), the most recent value sent to the input (connection B) remains as the result. If the GATE is blocked (connection A), the value that the result had before the block is retained. After enabling, the result corresponds to the value of the input (connection B). The following dependent parameters are displayed: • *Description* • *GATE blocks if Group Object "Connection A" equals* • *Value of Group Object "Connection A" after KNX voltage recovery* • *Value of Group Object "Connection B" after KNX voltage recovery* • *Read input Group Objects after KNX voltage recovery or download* • *Invert result* • *Send value of Group Object "Status Result"* The following dependent Group Objects are displayed: • *Connection A* • *Connection B* • *Status Result [Logic]*
1 bit Inverter	The logic function *1 bit Inverter* is used. If the value 1 is present at the input, the result = 0. If the value 0 is present at the input, the result = 1. The following dependent parameters are displayed: • *Description* • *Read input Group Objects after KNX voltage recovery or download* • *Send value of Group Object "Status Result"* The following dependent Group Objects are displayed: • *Connection A* • *Status Result [Logic]*

Option	
Threshold	The function *Threshold* is used. The following dependent parameters are displayed: • *Data point type of Group Object "Threshold input"* • *Upper threshold* • *Lower threshold* • *Change thresholds via Group Objects* • *Change thresholds via i-bus® Tool* • *Result if upper threshold is exceeded* • *Minimum duration of the overshoot* • *Monitor range between thresholds* • *Result if lower threshold is dropped below* • *Minimum duration of the undershoot* • *Read input Group Objects after KNX voltage recovery or download* • *Send value of Group Objects "Status Result", "Status Input value between thresholds"*

7.2.4.2.2 Description

This parameter is used to specify a description of the Group Objects for the function. The description is shown in the name of the corresponding Group Objects.

Option	
Free text entry	Maximum 24 ASCII characters; the maximum number of characters may vary for other character formats.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ all options except *None*

7.2.4.2.3 GATE blocks if Group Object "Connection A" equals

This parameter is used to define the telegram value on the Group Object *Connection A* for which the GATE is blocked. If the GATE is blocked, telegrams received on the Group Object *Connection B* are ignored.

Option
1
0

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *GATE*

7.2.4.2.4 Value of Group Object "Connection A" after KNX voltage recovery

This parameter is used to define the value that is written to the *Connection A* Group Object after KNX voltage recovery.

Option	
1	The value 1 is written to the Group Object and the result of the function *Logic* is calculated.
0	The value 0 is written to the Group Object and the result of the function *Logic* is calculated.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE*

7.2.4.2.5 Value of Group Object "Connection B" after KNX voltage recovery

This parameter is used to define the value that is written to the *Connection B* Group Object after KNX voltage recovery.

Option	
1	The value 1 is written to the Group Object and the result of the function *Logic* is calculated.
0	The value 0 is written to the Group Object and the result of the function *Logic* is calculated.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE*

7.2.4.2.6 Read input Group Objects after KNX voltage recovery or download

This parameter is used to define whether the following input Group Objects are read after KNX voltage recovery or ETS download:

- Connection A
- Connection B

Note
To update the Group Objects after KNX voltage recovery, ETS download or ETS reset, the read flags must be set for the corresponding Group Objects of the sending device.

Option	
No	The input Group Objects are not read.
Yes	The input Group Objects are read. The result of the function *Logic* is re-calculated.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE* / *1 bit Inverter*

7.2.4.2.7 Invert result

This parameter is used to define whether the result of the function *Logic* is output inverted.

Option
No
Yes

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE*

7.2.4.2.8 Send value of Group Object "Status Result"

This parameter is used to define when the value of the following Group Object is sent on the bus (ABB i-bus® KNX):

- *Status Result [Logic]*

Note
With the 1-bit inverter, this parameter is permanently set to the option *After receiving input value or on request*.

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.
After receiving input value	The value is sent on the input Group Objects after reception of a telegram. The result is recalculated due to the reception of a telegram on the input Group Objects; the result does not necessarily need to change.
After receiving input value or on request	The value is sent on the input Group Objects after reception of a telegram or on request. The result is recalculated due to the reception of a telegram on the input Group Objects; the result does not necessarily need to change.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE*

7.2.4.2.9 Data point type of Group Object "Threshold input"

This parameter is used to define the data point type that is received via the Group Object "Threshold input" and evaluated.

Option	
Percent (DPT 5.001)	The following dependent Group Objects are displayed: • *Threshold input*
Counter pulses (DPT 5.010)	The following dependent Group Objects are displayed: • *Threshold input*
Counter pulses (DPT 7.001)	The following dependent Group Objects are displayed: • *Threshold input*
Temperature (DPT 9.001)	The following dependent Group Objects are displayed: • *Threshold input*
Lux (DPT 9.004)	The following dependent Group Objects are displayed: • *Threshold input*
Wind (DPT 9.005)	The following dependent Group Objects are displayed: • *Threshold input*
mA (DPT 9.021)	The following dependent Group Objects are displayed: • *Threshold input*
A (DPT 14.019)	The following dependent Group Objects are displayed: • *Threshold input*
W (DPT 14.056)	The following dependent Group Objects are displayed: • *Threshold input*
kW (DPT 9.024)	The following dependent Group Objects are displayed: • *Threshold input*
Wh (DPT 13.010)	The following dependent Group Objects are displayed: • *Threshold input*
kWh (DPT 13.013)	The following dependent Group Objects are displayed: • *Threshold input*

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.10 Upper threshold

This parameter is used to define the upper threshold. Default values and units depend on the option selected in the *Data point type of Group Object "Threshold input"* parameter.

Option	
0 ... 50 ... 100 %	Upper threshold on selection of DPT 5.001.
0 ... 200 ... 255	Upper threshold on selection of DPT 5.010.
0 ... 40000 ... 65535	Upper threshold on selection of DPT 7.001.
-100 ... 22 ... 250°C	Upper threshold on selection of DPT 9.001.
0 ... 400 ... 100,000 lux	Upper threshold on selection of DPT 9.004.
0 ... 8 ... 100 m/s	Upper threshold on selection of DPT 9.005.
0 ... 16000 ... 240000mA	Upper threshold on selection of DPT 9.021.
0 ... 16 ... 24A	Upper threshold on selection of DPT 14.019.
0 ... 4000 ... 10000 W	Upper threshold on selection of DPT 14.056.
0 ... 3 ... 10 kW	Upper threshold on selection of DPT 9.024.
0 ... 100000 ... 2147483647 Wh	Upper threshold on selection of DPT 13.010.
0 ... 70 ... 2147483647 kWh	Upper threshold on selection of DPT 13.013.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.11 Lower threshold

This parameter is used to define the lower threshold. Default values and units depend on the option selected in the *Data point type of Group Object "Threshold input"* parameter.

Option	
0 ... 20 ... 100 %	Lower threshold on selection of DPT 5.001.
0 ... 100 ... 255	Lower threshold on selection of DPT 5.010.
0 ... 10000 ... 65535	Lower threshold on selection of DPT 7.001.
-100 ... 18 ... 250°C	Lower threshold on selection of DPT 9.001.
0 ... 100 ... 100,000 lux	Lower threshold on selection of DPT 9.004.
0 ... 4 ... 100 m/s	Lower threshold on selection of DPT 9.005.
0 ... 1000 ... 240000mA	Lower threshold on selection of DPT 9.021.
0 ... 1 ... 24A	Lower threshold on selection of DPT 14.019.
0 ... 40 ... 10000 W	Lower threshold on selection of DPT 14.056.
0 ... 1 ... 10 kW	Lower threshold on selection of DPT 9.024.
0 ... 1000 ... 2147483647 Wh	Lower threshold on selection of DPT 13.010.
0 ... 25 ... 2147483647 kWh	Lower threshold on selection of DPT 13.013.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.12 Change thresholds via Group Objects

This parameter is used to define whether the thresholds set in ETS can be changed via the corresponding Group Objects.

Option	
No	The values cannot be changed via Group Objects.
Yes	The values can be changed via Group Objects. The Group Objects shown are dependent on the setting in the parameter *Data point type of Group Object "Threshold input"*. The following dependent parameters are displayed: • *Overwrite thresholds on download* The following dependent Group Objects are displayed: • *Change upper threshold* (DPT 13.010) *Change lower threshold* (DPT 13.010) • *Change upper threshold* (DPT 13.013) *Change lower threshold* (DPT 13.013) • *Change upper threshold* (DPT 14.019) *Change lower threshold* (DPT 14.019) • *Change upper threshold* (DPT 14.056) *Change lower threshold* (DPT 14.056) • *Change upper threshold* (DPT 5.001) *Change lower threshold* (DPT 5.001) • *Change upper threshold* (DPT 5.010) *Change lower threshold* (DPT 5.010) • *Change upper threshold* (DPT 7.001) *Change lower threshold* (DPT 7.001) • *Change upper threshold* (DPT 9.001) *Change lower threshold* (DPT 9.001) • *Change upper threshold* (DPT 9.004) *Change lower threshold* (DPT 9.004) • *Change upper threshold* (DPT 9.005) *Change lower threshold* (DPT 9.005) • *Change upper threshold* (DPT 9.021) *Change lower threshold* (DPT 9.021) • *Change upper threshold* (DPT 9.024) *Change lower threshold* (DPT 9.024)

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.13 Change thresholds via i-bus® Tool

This parameter is used to define whether the thresholds set in ETS can be changed via i-bus® Tool.

Note

The interface to i-bus® Tool is available from the following software versions:

- Application V1.1 or later

Option	
No	The values cannot be changed via i-bus® Tool.
Yes	The following dependent parameters are displayed: • *Overwrite thresholds on download*

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.14 Overwrite thresholds on download

This parameter is used to define whether the thresholds are overwritten during the ETS download of the application to the device.

Option	
No	The existing values in the device are not overwritten.
Yes	The existing values in the device are overwritten with the values defined in ETS.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*
    Or
  - Parameter *Change thresholds via i-bus® Tool* \ Option *Yes*

7.2.4.2.15 Result if upper threshold is exceeded

This parameter is used to define the result of the function *Threshold* when the value received at the threshold input exceeds the upper threshold.

The result is output on the following Group Object:

- *Status Result [threshold]*

Option	
Unchanged	The result of the function *Threshold* remains unchanged.
1	The result of the function *Threshold* is 1.
0	The result of the function *Threshold* is 0.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.16 Minimum duration of the overshoot

This parameter is used to define how long the value received at the threshold input must exceed the threshold before the result of the function *Threshold* is updated.

Option
00:00:00 … 18:12:15 hh:mm:ss

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.17 Monitor range between thresholds

This parameter is used to define whether the range between the thresholds is monitored and evaluated.

Option	
No	The range between the thresholds is not monitored and evaluated.
Yes	The following dependent parameters are displayed: • *Minimum dwell time between the thresholds* The following dependent Group Objects are displayed: • *Status Input value between thresholds*

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.18 Minimum dwell time between the thresholds

This parameter is used to define how long the value received at the threshold input must be between the thresholds before an evaluation occurs.

Option
00:00:00 … 18:12:15 hh:mm:ss

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Monitor range between thresholds* \ Option *Yes*

7.2.4.2.19 Result if lower threshold is dropped below

This parameter is used to define the result of the function *Threshold* when the value received at the threshold input falls below the lower threshold.

The result is output on the following Group Object:

- *Status Result [threshold]*

Option	
Unchanged	The result of the function *Threshold* remains unchanged.
1	The result of the function *Threshold* is 1.
0	The result of the function *Threshold* is 0.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.20 Minimum duration of the undershoot

This parameter is used to define how long the value received at the threshold input must undershoot the threshold before the result of the function *Threshold* is updated.

Option
00:00:00 … 18:12:15 hh:mm:ss

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.21 Read input Group Objects after KNX voltage recovery or download

This parameter is used to define whether the "Threshold input" Group Objects are read after KNX voltage recovery or ETS download, → parameter *Data point type of Group Object "Threshold input"*.

Note
To update the Group Objects after KNX voltage recovery, ETS download or ETS reset, the read flags must be set for the corresponding Group Objects of the sending device.

Option	
No	The input Group Objects are not read.
Yes	The input Group Objects are read. The result of the function *Threshold* is re-calculated.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.4.2.22 Send value of Group Objects "Status Result", "Status Input value between thresholds"

This parameter is used to define when the values of the following Group Objects are sent on the bus (ABB i-bus® KNX):

- *Status Input value between thresholds*
- *Status Result [threshold]*

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.
After receiving input value	The value is sent on the input Group Objects after reception of a telegram. The result is recalculated due to the reception of a telegram on the input Group Objects; the result does not necessarily need to change.
After receiving input value or on request	The value is sent on the input Group Objects after reception of a telegram or on request. The result is recalculated due to the reception of a telegram on the input Group Objects; the result does not necessarily need to change.

Prerequisites for visibility

- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

7.2.5 Parameter windows Thermoelectric (PWM) template

In this parameter window, the parameters can be set for all outputs.

The parameterization options in the template and in the parameter windows for the outputs are identical.

Note
For each output, you can decide whether to use the parameterization from the template (→ parameter *Output X Template*). When using parameterization from the template, the parameters are not visible in the parameter window for each output.
The individual setting for an output is made in the respective parameter window.

7.2.6 Parameter windows Open/close signal template

In this parameter window, the parameters can be set for all outputs.

The parameterization options in the template and in the parameter windows for the outputs are identical.

Note
For each output, you can decide whether to use the parameterization from the template (→ parameter *Output X Template*). When using parameterization from the template, the parameters are not visible in the parameter window for each output.
The individual setting for an output is made in the respective parameter window.

7.2.7 Parameter windows RTC template

In this parameter window, the parameters can be set for all outputs.

The parameterization options in the template and in the parameter windows for the outputs are identical.

Note
For each output, you can decide whether to use the parameterization from the template (→ parameter *RTC x Template*). When using parameterization from the template, the parameters are not visible in the parameter window for each output.
The individual setting for an output is made in the respective parameter window.

7.2.8 Parameter windows Output X:

Note
An individual description can be added to the name of the parameter window, → parameter *Output X description*.

7.2.8.1 Parameter windows Thermoelectric (PWM)

Note
If several outputs are to be set to the same values, parameterization can be performed in the template. When using parameterization from the template, the parameters are not visible in this parameter window.

The following settings can be made in this parameter window:
- Set the valve drive
- Enable the status Group Objects for the output
- Configure the valve purge
- Enable manual valve override

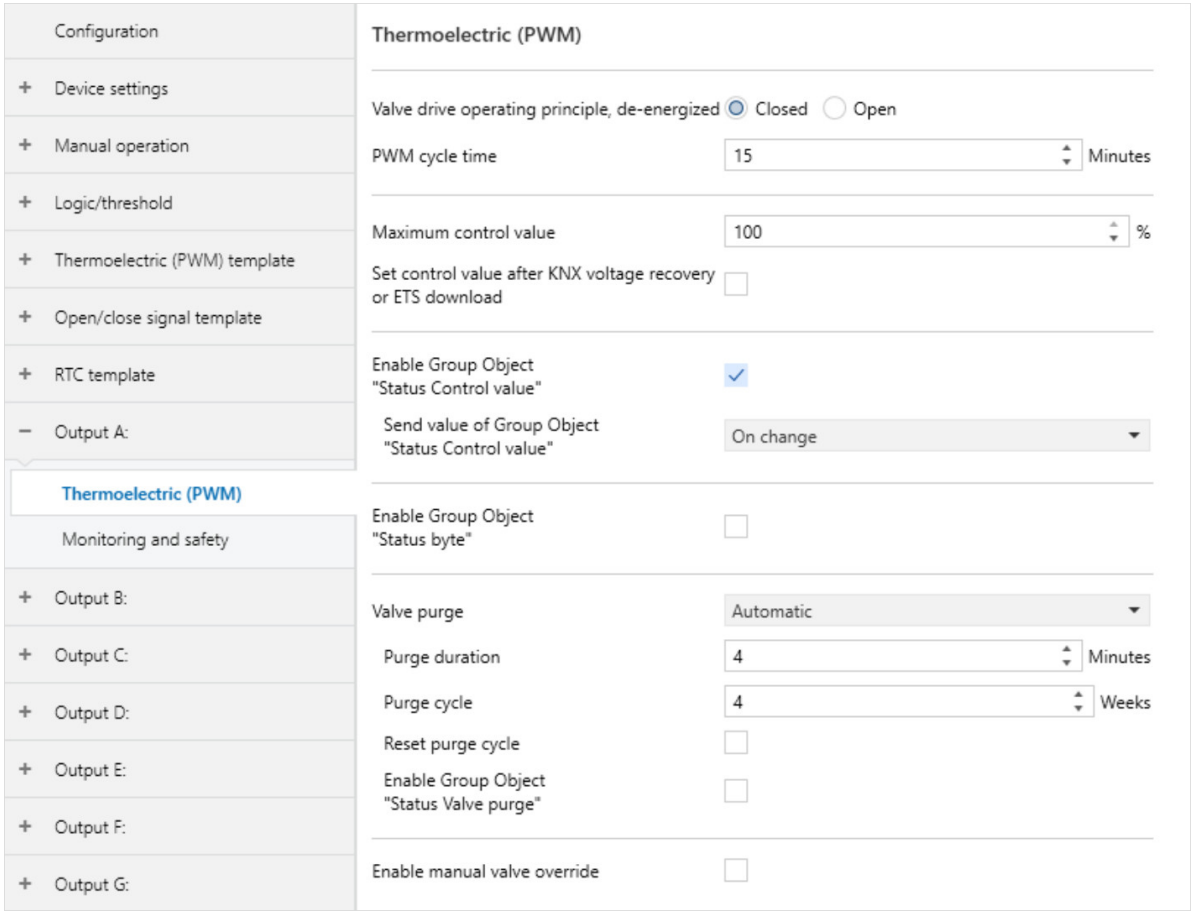


Fig. 24: Parameter window Thermoelectric (PWM)

This parameter window includes the following parameters:

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*

7.2.8.1.1 Valve drive operating principle, de-energized

This parameter is used to define the operating principle of the valve drive connected.

Option	
Closed	The valve is closed if no current flows through the valve drive. The valve is opened if current flows through the valve drive.
Open	The valve is opened if no current flows through the valve drive. The valve is closed if current flows through the valve drive.

7.2.8.1.2 PWM cycle time

This parameter is used to define the cycle time for the pulse width modulation on the valve output.

Note
The defined value must be greater than the opening/closing time of the valve drive.
The opening/closing time is listed in the technical data for the valve drive and corresponds to the total run time.

Option
1 … 15 … 255 min

7.2.8.1.3 Maximum control value

This parameter is used to define the maximum control value. The control is not allowed to exceed the maximum control value, even if the internal controller calculates a higher control value or an external controller specifies a higher control value.

Option
0 … 100 %

7.2.8.1.4 Set control value after KNX voltage recovery or ETS download

This parameter is used to define whether a control value is set after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option	
No	No control value is set after KNX voltage recovery or ETS download.
Yes	The following dependent parameters are displayed: • *Control value* • *Control value, Summer mode* • *Control value, Winter mode*

7.2.8.1.5 Control value

This parameter is used to define the control value after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option
0 …20…100 %

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*
  - Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Set control value after KNX voltage recovery or ETS download* \ Option *Yes*

7.2.8.1.6 Control value, Summer mode

This parameter is used to define the control value in Summer mode after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option
0 …10…100 %

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*
  - Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Set control value after KNX voltage recovery or ETS download* \ Option *Yes*

7.2.8.1.7 Control value, Winter mode

This parameter is used to define the control value in Winter mode after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option
0 … 50 … 100 %

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*
  - Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Set control value after KNX voltage recovery or ETS download* \ Option *Yes*

7.2.8.1.8 Enable Group Object "Status Control value"

This parameter enables one of the following Group Objects:
- *Status Control value* (1 bit)
- *Status Control value* (1 byte)

The configuration of the output determines which Group Object is enabled, → parameter *Output X Valve type*.

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Status Control value* (1 bit) • *Status Control value* (1 byte)

7.2.8.1.9 Send value of Group Object "Status Control value"

This parameter is used to define when the values of the following Group Objects are sent on the bus (ABB i-bus® KNX):
- *Status Control value* (1 bit)
- *Status Control value* (1 byte)

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
Cyclically	The value is sent cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent. The following dependent parameters are displayed: • *Sending cycle*
On change or cyclically	The value is sent on change or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change. The following dependent parameters are displayed: • *Sending cycle*
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.
On request or cyclically	The value is sent on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on request. The following dependent parameters are displayed: • *Sending cycle*
On change, on request or cyclically	The value is sent on change, on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change or on request. The following dependent parameters are displayed: • *Sending cycle*

Prerequisites for visibility

Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* / *Open/close signal* \ Parameter *Enable Group Object "Status Control value"* \ Option *Yes*

7.2.8.1.10 Sending cycle

This parameter is used to define the cycle in which the Group Object value is sent.

Note

The possible options and default values depend on the higher-level parameter.

Option
00:00:30 … 01:00:00 … 24:00:00 hh:mm:ss
00:01 … 00:05 … 23:59 hh:mm
1 … 15 … 240 min

Prerequisites for visibility

- The parameter appears at various points in the application. The visibility is dependent on the application and the higher-level parameter.

7.2.8.1.11 Enable Group Object "Status byte"

This parameter enables the following Group Object:

- *Status byte*

Option	
No	The Group Object is not enabled.
Yes	The Group Object is enabled; the value of the Group Object is sent on request. The following dependent parameters are displayed: • *Send value of Group Object "Status byte" on change* The following dependent Group Objects are displayed: • *Status byte*

7.2.8.1.12 Send value of Group Object "Status byte" on change

This parameter is used to define whether the value of the following Group Object is sent on change:
- *Status byte*

Option	
No	The value is sent only on request.
Yes	The value is sent on change and on request.

Prerequisites for visibility
Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* / *Open/close signal* \ Parameter *Enable Group Object "Status byte"* \ Option *Yes*

7.2.8.1.13 valve purge

This parameter is used to define how the valve purge is activated.

More information: → valve purge, Page 171.

Option	
Deactivated	Valve purge is deactivated.
Automatic	Valve purge takes place automatically in a set cycle. The following dependent parameters are displayed: • *Purge duration* • *Purge cycle* • *Reset purge cycle* • *Enable Group Object "Status Valve purge"*
Via Group Object	The valve purge can be triggered via a Group Object. The following dependent parameters are displayed: • *Purge duration* • *Enable Group Object "Status Valve purge"* The following dependent Group Objects are displayed: • *Activate valve purge*
Automatic or via Group Object	Valve purge takes place automatically in a set cycle. Valve purge can be triggered via a Group Object as well. The following dependent parameters are displayed: • *Purge duration* • *Purge cycle* • *Reset purge cycle* • *Enable Group Object "Status Valve purge"* The following dependent Group Objects are displayed: • *Activate valve purge*

7.2.8.1.14 Purge duration

This parameter is used to define the duration of the valve purge.

> **Note**
> The defined value must be greater than the opening/closing time of the valve drive.
> The opening/closing time is listed in the technical data for the valve drive and corresponds to the total run time.

Option
1 … 4 … 120 min

Prerequisites for visibility
- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *valve purge* \ all options except *Deactivated*

7.2.8.1.15 Purge cycle

This parameter is used to define the cycle for the automatic valve purge.

Option
1 … 4 … 52 weeks

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *valve purge* \ Option *Automatic / Automatic or via Group Object*

7.2.8.1.16 Reset purge cycle

This parameter is used to define whether the purge cycle is reset if it exceeds a parameterizable control value.

Option	
No	The purge cycle is only reset after a valve purge.
Yes	The following dependent parameters are displayed: • *Reset purge cycle from control value greater than*

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *valve purge* \ Option *Automatic / Automatic or via Group Object*

7.2.8.1.17 Reset purge cycle from control value greater than

This parameter is used to define the control value as of which the purge cycle is reset. If the value exceeds the set value, the purge cycle is reset. If the value falls below the set value, the purge cycle starts.

Option
0 … 50 … 100 %

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Reset purge cycle* \ Option *Yes*

7.2.8.1.18 Enable Group Object "Status Valve purge"

This parameter enables the following Group Object:

- *Status Valve purge*

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Send value of Group Object "Status Valve purge"* The following dependent Group Objects are displayed: • *Status Valve purge*

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *valve purge* \ all options except *Deactivated*

7.2.8.1.19 Send value of Group Object "Status Valve purge"

This parameter is used to define when the value of the following Group Object is sent on the bus (ABB i-bus® KNX):

- *Status Valve purge*

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Enable Group Object "Status Valve purge"* \ Option *Yes*

7.2.8.1.20 Enable manual valve override

This parameter enables manual valve override.

More information: → Manual valve override, Page 171.

Note
The value of Group Object *Control value override* becomes active only when manual valve override has been activated via Group Object *Activate/block control value override*.

Option	
No	Manual valve override cannot be enabled via a Group Object.
Yes	Manual valve override is enabled and can be activated or blocked via a Group Object. • *Control value override* • *Activate/block control value override*

7.2.8.2 Parameter windows Open/close signal

Note
If several outputs are to be set to the same values, parameterization can be performed in the template. When using parameterization from the template, the parameters are not visible in this parameter window.

The following settings can be made in this parameter window:
- Set the valve drive
- Enable the status Group Objects for the output
- Configure the valve purge
- Enable manual valve override

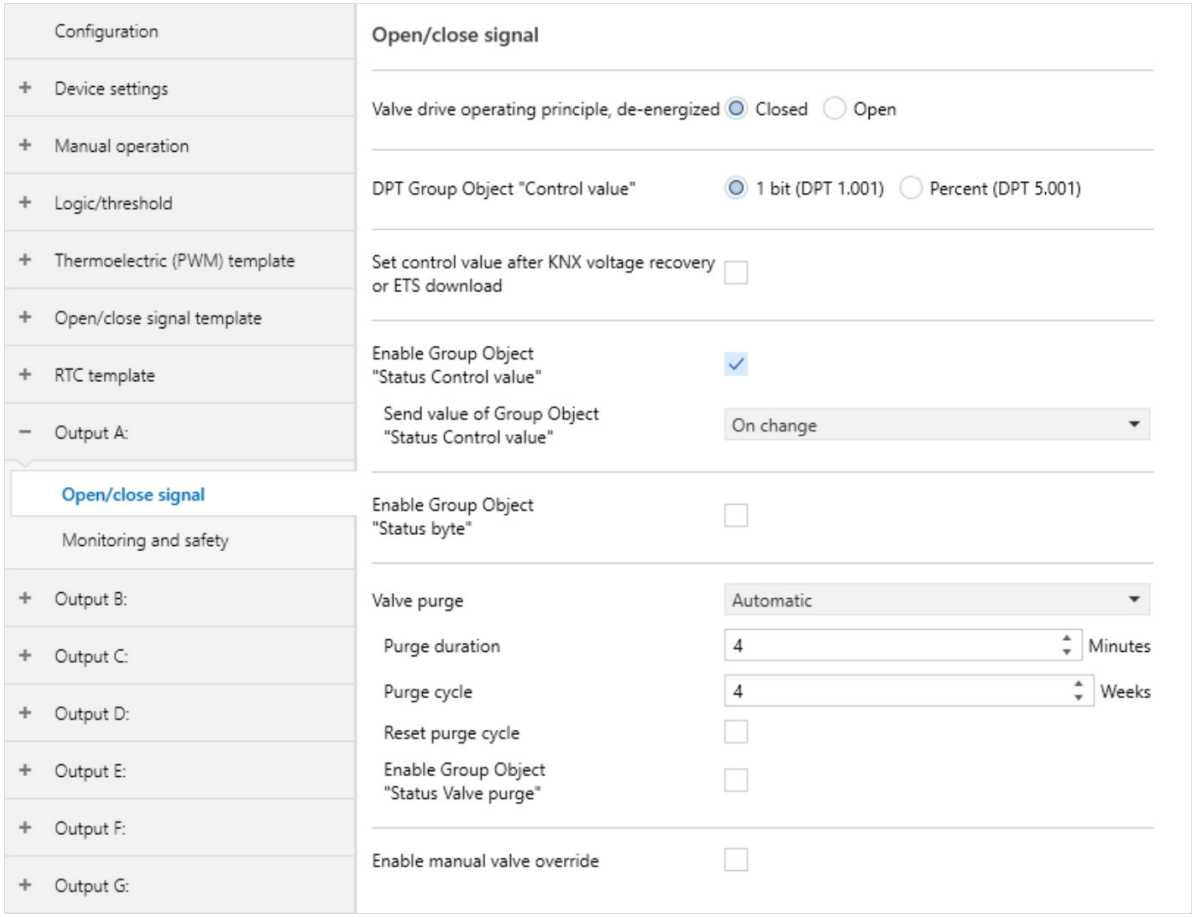


Fig. 25: Parameter window Open/close signal

This parameter window includes the following parameters:

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Output X Valve type* \ Option *Open/close signal*

7.2.8.2.1 Valve drive operating principle, de-energized

This parameter is used to define the operating principle of the valve drive connected.

Option	
Closed	The valve is closed if no current flows through the valve drive. The valve is opened if current flows through the valve drive.
Open	The valve is opened if no current flows through the valve drive. The valve is closed if current flows through the valve drive.

7.2.8.2.2 DPT Group Object "Control value"

This parameter is used to define the data point type (DPT) of the external control value.

Option	
1 bit (DPT 1.001)	The following dependent Group Objects are displayed: • *Control value*
Percent (DPT 5.001)	The following dependent Group Objects are displayed: • *Control value*

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Open/close signal*
  - Parameter *Output X Controller* \ Option *External*

7.2.8.2.3 Open if control value greater than

This parameter is used to define the control value as of which the valve is opened.

Option
0 … 99 %

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *DPT Group Object "Control value"* \ Option *Percent (DPT 5.001)*

7.2.8.2.4 Hysteresis

This parameter is used to define a double-sided hysteresis to prevent the valve from constantly opening and closing due to minimal fluctuations in the control value (→ parameter *Open if control value greater than*).

More information: → Hysteresis, Page 170.

Option
1 … 5 … 20 %

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Open/close signal*
  - Parameter *DPT Group Object "Control value"* \ Option *Percent (DPT 5.001)*
  - Parameter *Open if control value greater than* \ Option *> 1 %*

7.2.8.2.5 Set control value after KNX voltage recovery or ETS download

This parameter is used to define whether a control value is set after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option	
No	No control value is set after KNX voltage recovery or ETS download.
Yes	The following dependent parameters are displayed: • *Control value* • *Control value, Summer mode* • *Control value, Winter mode*

7.2.8.2.6 Control value

This parameter is used to define the control value after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Open/close signal*
  - Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X* \ Parameter window *Open/close signal* \ Parameter *Set control value after KNX voltage recovery or ETS download* \ Option *Yes*

7.2.8.2.7 Control value, Summer mode

This parameter is used to define the control value in Summer mode after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Open/close signal*
  - Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X* \ Parameter window *Open/close signal* \ Parameter *Set control value after KNX voltage recovery or ETS download* \ Option *Yes*

7.2.8.2.8 Control value, Winter mode

This parameter is used to define the control value in Winter mode after KNX voltage recovery or ETS download. The control value is valid until a new value is received via the bus (ABB i-bus® KNX).

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Open/close signal*
  - Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X* \ Parameter window *Open/close signal* \ Parameter *Set control value after KNX voltage recovery or ETS download* \ Option *Yes*

7.2.8.2.9 Enable Group Object "Status Control value"

This parameter enables one of the following Group Objects:

- *Status Control value* (1 bit)
- *Status Control value* (1 byte)

The configuration of the output determines which Group Object is enabled, → parameter *Output X Valve type*.

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Status Control value* (1 bit) • *Status Control value* (1 byte)

7.2.8.2.10 Send value of Group Object "Status Control value"

This parameter is used to define when the values of the following Group Objects are sent on the bus (ABB i-bus® KNX):

- *Status Control value* (1 bit)
- *Status Control value* (1 byte)

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
Cyclically	The value is sent cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent. The following dependent parameters are displayed: • *Sending cycle*
On change or cyclically	The value is sent on change or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change. The following dependent parameters are displayed: • *Sending cycle*
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.
On request or cyclically	The value is sent on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on request. The following dependent parameters are displayed: • *Sending cycle*
On change, on request or cyclically	The value is sent on change, on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change or on request. The following dependent parameters are displayed: • *Sending cycle*

Prerequisites for visibility

Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* / *Open/close signal* \ Parameter *Enable Group Object "Status Control value"* \ Option *Yes*

7.2.8.2.11 Sending cycle

This parameter is used to define the cycle in which the Group Object value is sent.

Note

The possible options and default values depend on the higher-level parameter.

Option
00:00:30 … 01:00:00 … 24:00:00 hh:mm:ss
00:01 … 00:05 … 23:59 hh:mm
1 … 15 … 240 min

Prerequisites for visibility

- The parameter appears at various points in the application. The visibility is dependent on the application and the higher-level parameter.

7.2.8.2.12 Enable Group Object "Status byte"

This parameter enables the following Group Object:

- *Status byte*

Option	
No	The Group Object is not enabled.
Yes	The Group Object is enabled; the value of the Group Object is sent on request. The following dependent parameters are displayed: • *Send value of Group Object "Status byte" on change* The following dependent Group Objects are displayed: • *Status byte*

7.2.8.2.13 Send value of Group Object "Status byte" on change

This parameter is used to define whether the value of the following Group Object is sent on change:
- *Status byte*

Option	
No	The value is sent only on request.
Yes	The value is sent on change and on request.

Prerequisites for visibility
Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* / *Open/close signal* \ Parameter *Enable Group Object "Status byte"* \ Option *Yes*

7.2.8.2.14 valve purge

This parameter is used to define how the valve purge is activated.

More information: → valve purge, Page 171.

Option	
Deactivated	Valve purge is deactivated.
Automatic	Valve purge takes place automatically in a set cycle. The following dependent parameters are displayed: • *Purge duration* • *Purge cycle* • *Reset purge cycle* • *Enable Group Object "Status Valve purge"*
Via Group Object	The valve purge can be triggered via a Group Object. The following dependent parameters are displayed: • *Purge duration* • *Enable Group Object "Status Valve purge"* The following dependent Group Objects are displayed: • *Activate valve purge*
Automatic or via Group Object	Valve purge takes place automatically in a set cycle. Valve purge can be triggered via a Group Object as well. The following dependent parameters are displayed: • *Purge duration* • *Purge cycle* • *Reset purge cycle* • *Enable Group Object "Status Valve purge"* The following dependent Group Objects are displayed: • *Activate valve purge*

7.2.8.2.15 Purge duration

This parameter is used to define the duration of the valve purge.

> **Note**
> The defined value must be greater than the opening/closing time of the valve drive.
> The opening/closing time is listed in the technical data for the valve drive and corresponds to the total run time.

Option
1 … 4 … 120 min

Prerequisites for visibility
- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *valve purge* \ all options except *Deactivated*

7.2.8.2.16 Purge cycle

This parameter is used to define the cycle for the automatic valve purge.

More information: → valve purge, Page 171.

Option
1 … 4 … 52 weeks

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *valve purge* \ Option *Automatic* / *Automatic or via Group Object*

7.2.8.2.17 Reset purge cycle

This parameter is used to define whether the purge cycle is reset after each time the valve has been fully opened.

Option	
No	The purge cycle is only reset after a valve purge.
Yes	The purge cycle is reset after each time the valve has been fully opened.

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *valve purge* \ Option *Automatic* / *Automatic or via Group Object*

7.2.8.2.18 Enable Group Object "Status Valve purge"

This parameter enables the following Group Object:

- *Status Valve purge*

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Send value of Group Object "Status Valve purge"* The following dependent Group Objects are displayed: • *Status Valve purge*

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *valve purge* \ all options except *Deactivated*

7.2.8.2.19 Send value of Group Object "Status Valve purge"

This parameter is used to define when the value of the following Group Object is sent on the bus (ABB i-bus® KNX):

- *Status Valve purge*

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *Enable Group Object "Status Valve purge"* \ Option *Yes*

7.2.8.2.20 Enable manual valve override

This parameter enables manual valve override.

More information: → Manual valve override, Page 171.

> **Note**
> The value of Group Object *Control value override* becomes active only when manual valve override has been activated via Group Object *Activate/block control value override*.

Option	
No	Manual valve override cannot be enabled via a Group Object.
Yes	Manual valve override is enabled and can be activated or blocked via a Group Object. • *Control value override* • *Activate/block control value override*

7.2.8.3 Parameter windows Monitoring and safety [thermoelectric (PWM)]

Note
If several outputs are to be set to the same values, parameterization can be performed in the template. When using parameterization from the template, the parameters are not visible in this parameter window.

The following settings can be made in this parameter window:
- Activate cyclical monitoring and forced operation
- Specify control values for Safety mode and forced operation
- Activate valve fault monitoring

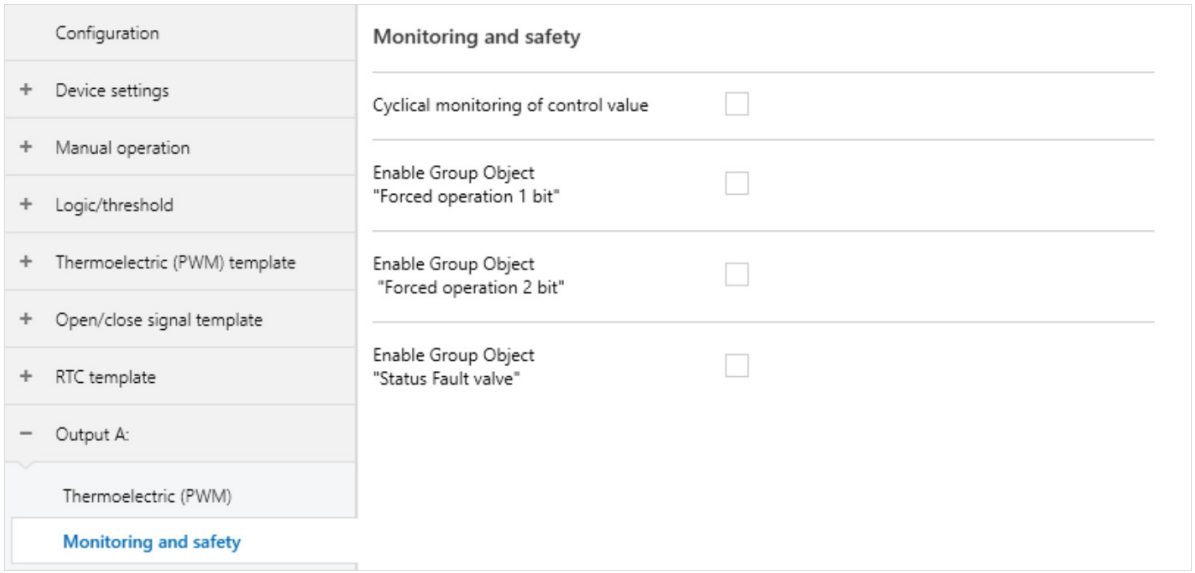


Fig. 26: Parameter window Monitoring and safety (thermoelectric)

This parameter window includes the following parameters:

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*

7.2.8.3.1 Cyclical monitoring of control value

This parameter is used to activate cyclical monitoring of the external control value.

More information: → Safety mode, Page 29.

Option	
No	Cyclical monitoring is not activated.
Yes	The following dependent parameters are displayed: • *Monitoring time* • *Control value in Safety mode* • *Control value in Safety and Summer mode* • *Control value in Safety and Winter mode* The following dependent Group Objects are displayed: • *Status Safety mode*

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*
  - Parameter *Output X Controller* \ Option *External*

7.2.8.3.2 Monitoring time

This parameter is used to define the period during which a value must be received on monitored Group Objects.

More information: → Cyclical monitoring, Page 175.

Note
The monitoring time should be at least quadruple the cyclical sending time of the sending device. As a result, the set reactions or the alarms will not be triggered immediately if a telegram is missing, e.g. due to high bus load.

Option
00:03 … 01:00 … 12:00 hh:mm

Prerequisites for visibility

- The visibility is dependent on the application and the higher-level parameter.

7.2.8.3.3 Control value in Safety mode

This parameter is used to define the control value. The control value is valid for as long as Safety mode is active.

Option
0 …20…100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Cyclical monitoring of control value* \ Option *Yes*

7.2.8.3.4 Control value in Safety and Summer mode

This parameter is used to define the control value in Summer mode. The control value is valid for as long as Safety mode is active.

Option
0 …10…100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Cyclical monitoring of control value* \ Option *Yes*

7.2.8.3.5 Control value in Safety and Winter mode

This parameter is used to define the control value in Winter mode. The control value is valid for as long as Safety mode is active.

Option
0 … 50 … 100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Cyclical monitoring of control value* \ Option *Yes*

7.2.8.3.6 Enable Group Object "Forced operation 1 bit"

This parameter enables the following Group Object:

- *Forced operation 1 bit*

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Control value* • *Control value, Summer mode* • *Control value, Winter mode* The following dependent Group Objects are displayed: • *Forced operation 1 bit*

7.2.8.3.7 Control value

This parameter is used to define the control value. The control value is valid for as long as 1-bit forced operation is active.

Option
0 …20…100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*

7.2.8.3.8 Control value, Summer mode

This parameter is used to define the control value in Summer mode. The control value is valid for as long as 1-bit forced operation is active.

Option
0 …10…100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*

7.2.8.3.9 Control value, Winter mode

This parameter is used to define the control value in Winter mode. The control value is valid for as long as 1-bit forced operation is active.

Option
0 … 50 … 100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*

7.2.8.3.10 Enable Group Object "Forced operation 2 bit"

This parameter enables the following Group Object:

- *Forced operation 2 bit*

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Control value "OFF"* • *Control value "ON"* • *Control value "OFF", Summer mode* • *Control value "OFF", Winter mode* • *Control value "ON", summer mode* • *Control value "ON", winter mode* The following dependent Group Objects are displayed: • *Forced operation 2 bit*

7.2.8.3.11 Control value "OFF"

This parameter is used to specify the control value "OFF". The control value is valid for as long as 2-bit forced operation is active.

Option
0 … 100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*

7.2.8.3.12 Control value "OFF", Summer mode

This parameter is used to specify the control value "OFF" in Summer mode. The control value is valid for as long as 2-bit forced operation is active.

Option
0 ... 100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*

7.2.8.3.13 Control value "OFF", Winter mode

This parameter is used to specify the control value "OFF" in Winter mode. The control value is valid for as long as 2-bit forced operation is active.

Option
0 ... 100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*

7.2.8.3.14 Control value "ON"

This parameter is used to specify the control value "ON". The control value is valid for as long as 2-bit forced operation is active.

Option
0 ... 50 ... 100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*

7.2.8.3.15 Control value "ON", summer mode

This parameter is used to specify the control value "ON" in Summer mode. The control value is valid for as long as 2-bit forced operation is active.

Option
0 ...20...100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*

7.2.8.3.16 Control value "ON", winter mode

This parameter is used to specify the control value "ON" in Winter mode. The control value is valid for as long as 2-bit forced operation is active.

Option
0 … 50 … 100 %

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*

7.2.8.3.17 Enable Group Object "Status Fault valve"

This parameter enables the following Group Object:

- *Status Fault valve*

Note

The Group Object *Status Fault valve* reports a fault if at least one of the following errors occurs on the output pair (two adjacent outputs, e.g. A/B, C/D, etc.):

- Overload/short circuit lasting > 10 s
- No supply voltage (VC/S 6.2.2.2: terminal pair 1/2, VC/S 12.2.2.2: terminal pairs 1/2 and 15/16)

If the upper outputs (valve outputs A ... F) or the lower outputs (valve outputs G ... L) are not in use on the VC/S 12.2.2.2 and the corresponding supply voltage is not connected, the corresponding outputs should be deactivated (→ parameter *Output X Valve type*) to avoid unwanted fault messages.

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Value of Group Object "Status Fault valve"* • *Send value of Group Object "Status Fault valve"* The following dependent Group Objects are displayed: • *Status Fault valve*

7.2.8.3.18 Value of Group Object "Status Fault valve"

This parameter is used to define the telegram value that the Group Object *Status Fault valve* sends.

Option
1: Fault, 0: No fault
0: Fault, 1: No fault

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Status Fault valve"* \ Option *Yes*

7.2.8.3.19 Send value of Group Object "Status Fault valve"

This parameter is used to define when the value of the following Group Object is sent on the bus (ABB i-bus® KNX):

- *Status Fault valve*

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
Cyclically	The value is sent cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent. The following dependent parameters are displayed: • *Sending cycle*
On change or cyclically	The value is sent on change or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change. The following dependent parameters are displayed: • *Sending cycle*
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.
On request or cyclically	The value is sent on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on request. The following dependent parameters are displayed: • *Sending cycle*
On change, on request or cyclically	The value is sent on change, on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change or on request. The following dependent parameters are displayed: • *Sending cycle*

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Status Fault valve"* \ Option *Yes*

7.2.8.3.20 Sending cycle

This parameter is used to define the cycle in which the Group Object value is sent.

Note
The possible options and default values depend on the higher-level parameter.

Option
00:00:30 … 01:00:00 … 24:00:00 hh:mm:ss
00:01 … 00:05 … 23:59 hh:mm
1 … 15 … 240 min

Prerequisites for visibility

- The parameter appears at various points in the application. The visibility is dependent on the application and the higher-level parameter.

7.2.8.4 Parameter windows Monitoring and safety [open/close signal]

> **Note**
> If several outputs are to be set to the same values, parameterization can be performed in the template. When using parameterization from the template, the parameters are not visible in this parameter window.

The following settings can be made in this parameter window:
- Activate cyclical monitoring and forced operation
- Specify valve positions for Safety mode and forced operation
- Activate valve fault monitoring

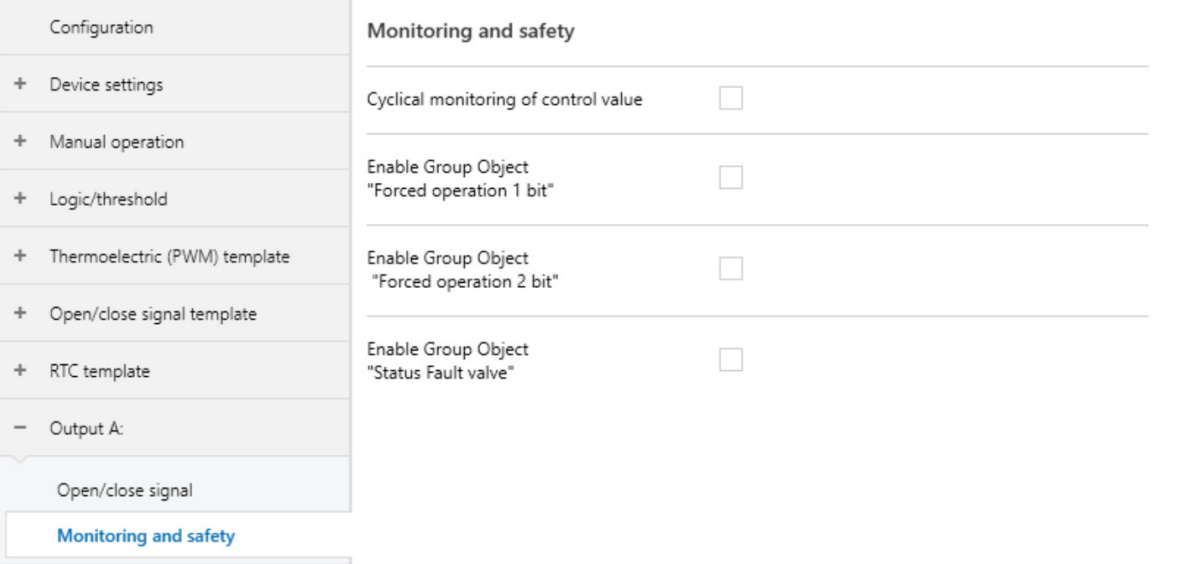


Fig. 27: Parameter window Monitoring and safety (open/close signal)

This parameter window includes the following parameters:

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ Option *Open/close signal*

7.2.8.4.1 Cyclical monitoring of "Control value"

This parameter is used to activate cyclical monitoring of the external control value.

More information: → Safety mode, Page 29.

Option	
No	Cyclical monitoring is not activated.
Yes	The following dependent parameters are displayed: • *Monitoring time* • *Control value in Safety mode* • *Control value in Safety and Summer mode* • *Control value in Safety and Winter mode* The following dependent Group Objects are displayed: • *Status Safety mode*

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Open/close signal*
  - Parameter *Output X Controller* \ Option *External*

7.2.8.4.2 Monitoring time

This parameter is used to define the period during which a value must be received on monitored Group Objects.

More information: → Cyclical monitoring, Page 175.

Note
The monitoring time should be at least quadruple the cyclical sending time of the sending device. As a result, the set reactions or the alarms will not be triggered immediately if a telegram is missing, e.g. due to high bus load.

Option
00:03 … 01:00 … 12:00 hh:mm

Prerequisites for visibility

- The visibility is dependent on the application and the higher-level parameter.

7.2.8.4.3 Control value in Safety mode

This parameter is used to define the control value. The control value is valid for as long as Safety mode is active.

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Cyclical monitoring of "Control value"* \ Option *Yes*

7.2.8.4.4 Control value in Safety and Summer mode

This parameter is used to define the control value in Summer mode. The control value is valid for as long as Safety mode is active.

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Cyclical monitoring of "Control value"* \ Option *Yes*

7.2.8.4.5 Control value in Safety and Winter mode

This parameter is used to define the control value in Winter mode. The control value is valid for as long as Safety mode is active.

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Cyclical monitoring of "Control value"* \ Option *Yes*

7.2.8.4.6 Enable Group Object "Forced operation 1 bit"

This parameter enables the following Group Object:

- *Forced operation 1 bit*

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Control value* • *Control value, Summer mode* • *Control value, Winter mode* The following dependent Group Objects are displayed: • *Forced operation 1 bit*

7.2.8.4.7 Control value

This parameter is used to define the control value. The control value is valid for as long as 1-bit forced operation is active.

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*

7.2.8.4.8 Control value, Summer mode

This parameter is used to define the control value in Summer mode. The control value is valid for as long as 1-bit forced operation is active.

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*

7.2.8.4.9 Control value, Winter mode

This parameter is used to define the control value in Winter mode. The control value is valid for as long as 1-bit forced operation is active.

Option	
0 %	The control value 0 % applies.
100 %	The control value 100 % applies.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*

7.2.8.4.10 Enable Group Object "Forced operation 2 bit"

This parameter enables the following Group Object:

- *Forced operation 2 bit*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Forced operation 2 bit*

7.2.8.4.11 Enable Group Object "Status Fault valve"

This parameter enables the following Group Object:
- *Status Fault valve*

> **Note**
> The Group Object *Status Fault valve* reports a fault if at least one of the following errors occurs on the output pair (two adjacent outputs, e.g. A/B, C/D, etc.):
> - Overload/short circuit lasting > 10 s
> - No supply voltage (VC/S 6.2.2.2: terminal pair 1/2, VC/S 12.2.2.2: terminal pairs 1/2 and 15/16)
>
> If the upper outputs (valve outputs A ... F) or the lower outputs (valve outputs G ... L) are not in use on the VC/S 12.2.2.2 and the corresponding supply voltage is not connected, the corresponding outputs should be deactivated (→ parameter *Output X Valve type*) to avoid unwanted fault messages.

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Value of Group Object "Status Fault valve"* • *Send value of Group Object "Status Fault valve"* The following dependent Group Objects are displayed: • *Status Fault valve*

7.2.8.4.12 Value of Group Object "Status Fault valve"

This parameter is used to define the telegram value that the Group Object *Status Fault valve* sends.

Option
1: Fault, 0: No fault
0: Fault, 1: No fault

Prerequisites for visibility
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Status Fault valve"* \ Option *Yes*

7.2.8.4.13 Send value of Group Object "Status Fault valve"

This parameter is used to define when the value of the following Group Object is sent on the bus (ABB i-bus® KNX):
- *Status Fault valve*

Option	
No, update only	The value is updated but is not sent.
On change	The value is sent on change.
Cyclically	The value is sent cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent. The following dependent parameters are displayed: • *Sending cycle*
On change or cyclically	The value is sent on change or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change. The following dependent parameters are displayed: • *Sending cycle*
On request	The value is sent on request.
On change or on request	The value is sent on change or on request.
On request or cyclically	The value is sent on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on request. The following dependent parameters are displayed: • *Sending cycle*
On change, on request or cyclically	The value is sent on change, on request or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change or on request. The following dependent parameters are displayed: • *Sending cycle*

Prerequisites for visibility

- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Status Fault valve"* \ Option *Yes*

7.2.8.4.14 Sending cycle

This parameter is used to define the cycle in which the Group Object value is sent.

Note
The possible options and default values depend on the higher-level parameter.

Option
00:00:30 … 01:00:00 … 24:00:00 hh:mm:ss
00:01 … 00:05 … 23:59 hh:mm
1 … 15 … 240 min

Prerequisites for visibility

- The parameter appears at various points in the application. The visibility is dependent on the application and the higher-level parameter.

7.2.9 Parameter windows RTC x

Note
An individual description can be added to the name of the parameter window, → parameter *RTC x Description*.

Note
If several outputs are to be set to the same values, parameterization can be performed in the template. When using parameterization from the template, the parameters are not visible in this parameter window.

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*

7.2.9.1 Parameter windows Basic settings

The following settings can be made in this parameter window:

- Set the use of the basic and additional stages
- Set the reaction of the controller

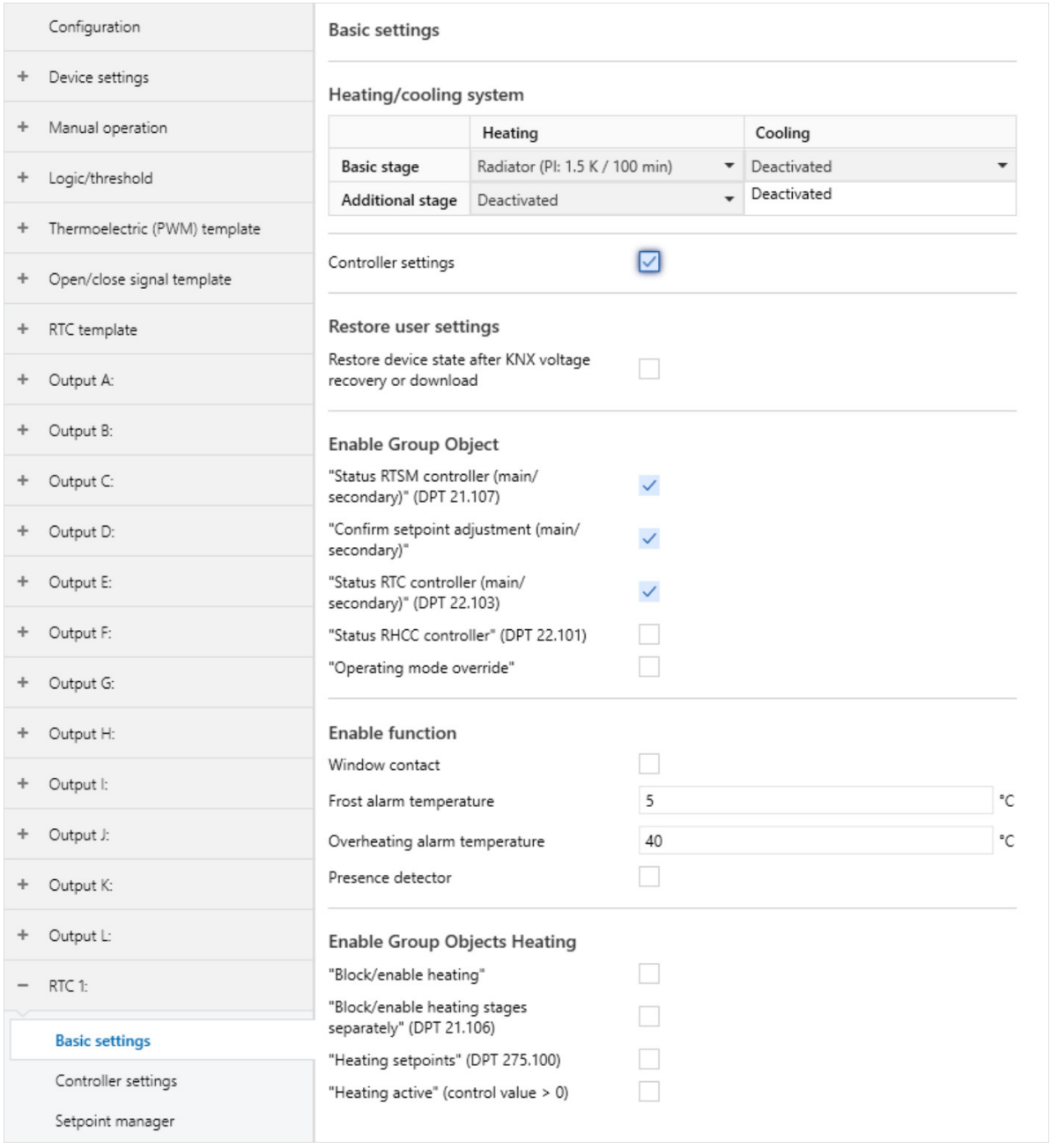


Fig. 28: Parameter window Basic Settings

This parameter window includes the following parameters:

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Enable RTC x* \ Option *Yes*
  - Parameter *RTC x Template* \ Option *No*

7.2.9.1.1 Basic-stage heating

This parameter is used to define how basic-stage heating is used. The controller is preset based on the selected option.

Option	
Deactivated	Basic-stage heating is deactivated.
Radiator (PI: 1.5 K / 100 min)	Basic-stage heating is set for the use of a radiator. The following dependent parameter windows are displayed: • *Setpoint manager* • *Setpoint adjustment* The following dependent parameters are displayed: • *Additional-stage heating* • *Enable Group Object "Block/enable heating"* • *Enable Group Object "Block/enable heating stages separately" (DPT 21.106)* • *Enable Group Object "Heating active" (control value > 0)* The following dependent Group Objects are displayed: • *Control value basic-stage heating*
Area heating (PI: 4 K / 200 min)	Basic-stage heating is set for the use of area heating. The following dependent parameter windows are displayed: • *Setpoint manager* • *Setpoint adjustment* The following dependent parameters are displayed: • *Additional-stage heating* • *Enable Group Object "Block/enable heating"* • *Enable Group Object "Block/enable heating stages separately" (DPT 21.106)* • *Enable Group Object "Heating active" (control value > 0)* The following dependent Group Objects are displayed: • *Control value basic-stage heating*
Free configuration	Basic-stage heating can be configured as required. The following dependent parameter windows are displayed: • *Setpoint manager* • *Setpoint adjustment* The following dependent parameters are displayed: • *Additional-stage heating* • *Enable Group Object "Block/enable heating"* • *Enable Group Object "Block/enable heating stages separately" (DPT 21.106)* • *Enable Group Object "Heating active" (control value > 0)* The following dependent Group Objects are displayed: • *Control value basic-stage heating*

7.2.9.1.2 Additional-stage heating

This parameter is used to define how the additional-stage heating is used. The controller is preset based on the selected option.

Option	
Deactivated	Additional-stage heating is deactivated.
Radiator (PI: 1.5 K / 100 min)	Additional-stage heating is set for the use of a radiator. The following dependent parameters are displayed: • *Additional-stage setpoint difference* The following dependent Group Objects are displayed: • *Control value additional-stage heating*
Area heating (PI: 4 K / 200 min)	Additional-stage heating is set for the use of area heating. The following dependent parameters are displayed: • *Additional-stage setpoint difference* The following dependent Group Objects are displayed: • *Control value additional-stage heating*
Electric heater (2-point)	Additional-stage heating is set for the use of an electric heater. The following dependent parameters are displayed: • *Additional-stage setpoint difference* • *Hysteresis setpoint difference* The following dependent Group Objects are displayed: • *Control value additional-stage heating*
Free configuration	The additional-stage heating can be configured as required. The following dependent parameters are displayed: • *Additional-stage setpoint difference* The following dependent Group Objects are displayed: • *Control value additional-stage heating*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*

7.2.9.1.3 Basic-stage cooling

This parameter is used to define how basic-stage cooling is used. The controller is preset based on the selected option.

Option	
Deactivated	Basic-stage cooling is deactivated.
Cooling ceiling (PI: 5 K / 240 min)	Basic-stage cooling is set for the use of a cooling ceiling. The following dependent parameter windows are displayed: • *Setpoint manager* • *Setpoint adjustment* The following dependent parameters are displayed: • *Additional-stage cooling* • *Enable Group Object "Block/enable cooling"* • *Enable Group Object "Block/enable cooling stages separately" (DPT 21.106)* • *Enable Group Object "Cooling active" (control value > 0)* • *Enable Group Object "Dew point alarm"* The following dependent Group Objects are displayed: • *Control value basic-stage cooling*
Free configuration	Basic-stage cooling can be configured as required. The following dependent parameter windows are displayed: • *Setpoint manager* • *Setpoint adjustment* The following dependent parameters are displayed: • *Additional-stage cooling* • *Enable Group Object "Block/enable cooling"* • *Enable Group Object "Block/enable cooling stages separately" (DPT 21.106)* • *Enable Group Object "Cooling active" (control value > 0)* • *Enable Group Object "Dew point alarm"* The following dependent Group Objects are displayed: • *Control value basic-stage cooling*

7.2.9.1.4 Additional-stage cooling

This parameter is used to define how the additional-stage cooling is used. The controller is preset based on the selected option.

Option	
Deactivated	Additional-stage cooling is deactivated.
Cooling ceiling (PI: 5 K / 240 min)	Additional-stage cooling is set for the use of a cooling ceiling. The following dependent parameters are displayed: • *Additional-stage setpoint difference* The following dependent Group Objects are displayed: • *Control value additional-stage cooling*
Additional fan (2-point)	Additional-stage cooling is set for the use of a fan. The following dependent parameters are displayed: • *Additional-stage setpoint difference* • *Hysteresis setpoint difference* The following dependent Group Objects are displayed: • *Control value additional-stage cooling*
Free configuration	Additional-stage cooling can be configured as required. The following dependent parameters are displayed: • *Additional-stage setpoint difference* The following dependent Group Objects are displayed: • *Control value additional-stage cooling*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.5 Controller settings

This parameter unhides the parameter window *Controller settings*.

If at least one basic or additional stage is freely configurable, this parameter is permanently set to the option *Yes* and the parameter window *Controller settings* is always shown.

Option	
No	The parameter window is not shown.
Yes	The following dependent parameter windows are displayed: • *Controller settings*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
    Or
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.6 Type of operation after KNX voltage recovery or download

This parameter is used to define which type of operation is activated after KNX voltage recovery, ETS reset or ETS download.

Option
Heating
Cooling

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.7 Type of heating/cooling system

This parameter is used to define the type of heating/cooling system used. The selection affects the changeover behavior between types of operation (*Heating*/*Cooling*).

Option	
4-pipe	The activated heating and cooling devices are in a 4-pipe system.
2-pipe	The activated heating and cooling devices are in a 2-pipe system. The parameter *Heating/cooling changeover* is permanently set to the option *Via Group Object*.

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.8 Heating/cooling changeover

This parameter is used to define how the change between operating modes takes place.

> **Note**
> When using a 2-pipe system, this parameter is permanently set to the option *Via Group Object* (→ parameter *Type of heating/cooling system*).

Option	
Via Group Object	The change between the types of operation takes place via Group Object *Heating/cooling changeover*. The following dependent Group Objects are displayed: • *Heating/cooling changeover*
Automatic	The change between the types of operation takes place automatically depending on the difference between the actual and setpoint temperature. The following dependent parameters are displayed: • *Hysteresis for heating/cooling changeover*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.9 Restore device state after KNX voltage recovery or download

This parameter is used to define whether previous adjustments to the type of operation, operating mode and setpoints are restored after a restart (KNX voltage recovery, ETS reset or ETS download).

Option	
No	The adjustments are discarded.
Yes	The adjustments are retained. The following dependent parameters are displayed: • *Overwrite saved adjustments on download*

7.2.9.1.10 Overwrite saved adjustments on download

This parameter is used to define whether the adjustments stored in the device are overwritten during an ETS download with the values specified in ETS.

Option
No
Yes

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Restore device state after KNX voltage recovery or download* \ Option *Yes*

7.2.9.1.11 Enable Group Object "Status RTSM controller (main/secondary)" (DPT 21.107)

This parameter enables the following Group Object:

- *Status RTSM Controller (main/secondary)*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Status RTSM Controller (main/secondary)*

7.2.9.1.12 Enable Group Object "Confirm setpoint adjustment (main/secondary)"

This parameter enables the following Group Object:
- *Confirm setpoint adjustment (main/secondary)*

Option	
No	
Yes	The following dependent Group Objects are displayed: • *Confirm setpoint adjustment (main/secondary)*

7.2.9.1.13 Enable Group Object "Status RTC (main/secondary)" (DPT 22.103)

This parameter enables Group Object *Status RTC Controller (main/secondary)* as well as the frost and overheating alarm.

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Frost alarm temperature* • *Overheating alarm temperature* The following dependent Group Objects are displayed: • *Status RTC Controller (main/secondary)*

7.2.9.1.14 Enable Group Object "Status RHCC" (DPT 22.101)

This parameter enables Group Object *Status RHCC* as well as the frost and overheating alarm.

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Frost alarm temperature* • *Overheating alarm temperature* The following dependent Group Objects are displayed: • *Status RHCC*

7.2.9.1.15 Enable Group Object "Status Heating/cooling changeover"

This parameter enables the following Group Object:
- *Status Heating/cooling changeover*

Option	
No	
Yes	The following dependent Group Objects are displayed: • *Status Heating/cooling changeover*

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.16 Enable Group Object "Operating mode override"

This parameter enables the following Group Object:
- *Operating mode override*

Option	
No	
Yes	The following dependent Group Objects are displayed: • *Operating mode override*

7.2.9.1.17 Enable function: Window contact

This parameter enables the following Group Object:
- *Window contact*

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *Delay time* The following dependent Group Objects are displayed: • *Window contact*

7.2.9.1.18 Delay time

This parameter is used to define the delay time for the window contact.

Note
To prevent split units from reacting to the window being opened briefly, it is possible to delay processing of the "window open" telegram. If the window remains open throughout the delay time, the "window open" telegram is processed. The delay time restarts each time the window is opened.

Option
00:00:00 … 00:00:30 … 01:00:00 hh:mm:ss

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable function: Window contact* \ Option *Yes*

7.2.9.1.19 Frost alarm temperature

This parameter is used to define the limit temperature for the frost alarm. The frost alarm becomes active if the room temperature falls below the set value.

Note
The frost alarm is used to report critical temperatures in the room and does not affect temperature control. The Building Protection setpoint temperatures from the control ensure that critical temperatures in the room are prevented. Active frost alarms are reported via the following Group Objects:
- *Status RTC Controller (main/secondary)* (bit 3)
- *Status RHCC* (bit 13)

Option
-10 … 5 … 20 °C

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Enable Group Object "Status RTC (main/secondary)" (DPT 22.103)* \ Option *Yes*
    Or
  - Parameter *Enable Group Object "Status RHCC" (DPT 22.101)* \ Options *Yes*

7.2.9.1.20 Overheating alarm temperature

This parameter is used to define the limit temperature for the overheating alarm. The overheating alarm becomes active if the room temperature exceeds the set value.

Note
The overheating alarm is used to report critical temperatures in the room and does not affect temperature control. The Building Protection setpoint temperatures from the control ensure that critical temperatures in the room are prevented. Active overheating alarms are reported via the following Group Objects:
- *Status RTC Controller (main/secondary)* (bit 4)
- *Status RHCC* (bit 14)

Option
-10 … 40 … 60 °C

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Enable Group Object "Status RTC (main/secondary)" (DPT 22.103)* \ Option *Yes*
    Or
  - Parameter *Enable Group Object "Status RHCC" (DPT 22.101)* \ Options *Yes*

7.2.9.1.21 Enable function: Presence detector

This parameter enables the following Group Object:
- *Presence detector*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Presence detector*

7.2.9.1.22 Enable function: Controller synchronization (DPT 20.105)

This parameter enables controller synchronization. Enabling is only necessary if several controllers are used in a control zone and all controllers are required to operate in the same type of operation (*Heating* or *Cooling*).

Option	
No	Controller synchronization is not enabled.
Yes	The following dependent parameters are displayed: • *Synchronization settings*

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.23 Synchronization settings

This parameter is used to define whether the device is operated as a master or slave.

Option	
Master	The device is operated as a master and specifies the type of operation to the slave. The following dependent Group Objects are displayed: • *Controller synchronization of output*
Slave	The device is operated as a slave and receives the type of operation from the master. The following dependent Group Objects are displayed: • *Controller synchronization of input*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable function: Controller synchronization (DPT 20.105)* \ Option *Yes*

7.2.9.1.24 Enable Group Object "Block/enable heating"

This parameter enables the following Group Object:

- *Block/enable heating*

Option	
No	The Group Object is not enabled. The type of operation *Heating* cannot be blocked.
Yes	The type of operation *Heating* can be blocked via the enabled Group Object. If the type of operation *Heating* is blocked, the heating stages cannot be activated and the control value 0 % is sent to the outputs or on the bus (ABB i-bus® KNX). The following dependent Group Objects are displayed: • *Block/enable heating*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*

7.2.9.1.25 Enable Group Object "Block/enable heating stages separately" (DPT 21.106)

This parameter enables the following Group Object:

- *Block/enable heating stages separately*

Option	
No	The Group Object is not enabled. All heating stages are enabled, all control values are sent, and the output outputs the control value for the basic stage.
Yes	The heating stages can be separately blocked or enabled via the enabled Group Object. The following dependent Group Objects are displayed: • *Block/enable heating stages separately*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*

7.2.9.1.26 Enable Group Object "Heating setpoints" (DPT 275.100)

This parameter enables the following Group Object:
- *Heating setpoints*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Heating setpoints*

Prerequisites for visibility
- Parameter window *RTC x*
  - Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*

7.2.9.1.27 Enable Group Object "Heating active" (control value > 0)

This parameter enables the following Group Object:
- *Heating active*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Heating active*

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*

7.2.9.1.28 Enable Group Object "Block/enable cooling"

This parameter enables the following Group Object:
- *Block/enable cooling*

Option	
No	The Group Object is not enabled. The type of operation *Cooling* cannot be blocked.
Yes	The type of operation *Cooling* can be blocked via the enabled Group Object. If the type of operation *Cooling* is blocked, the cooling stages cannot be activated and the control value 0 % is sent to the outputs or on the bus (ABB i-bus® KNX). The following dependent Group Objects are displayed: • *Block/enable cooling*

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.29 Enable Group Object "Block/enable cooling stages separately" (DPT 21.106)

This parameter enables the following Group Object:
- *Block/enable cooling stages separately*

Option	
No	The Group Object is not enabled. All cooling stages are enabled, all control values are sent, and the output outputs the control value for the basic stage.
Yes	The cooling stages can be separately blocked or enabled via the enabled Group Object. The following dependent Group Objects are displayed: • *Block/enable cooling stages separately*

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.30 Enable Group Object "Cooling setpoints" (DPT 275.100)

This parameter enables the following Group Object:
- *Cooling setpoints*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Cooling setpoints*

Prerequisites for visibility
- Parameter window *RTC x*
  - Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*

7.2.9.1.31 Enable Group Object "Cooling active" (control value > 0)

This parameter enables the following Group Object:
- *Cooling active*

Option	
No	The Group Object is not enabled.
Yes	The following dependent Group Objects are displayed: • *Cooling active*

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.32 Enable Group Object "Dew point alarm"

This parameter enables the following Group Object:
- *Dew point alarm*

Option	
No	The Group Object is not enabled.
Yes	The following dependent parameters are displayed: • *When dew point alarm is active* The following dependent Group Objects are displayed: • *Dew point alarm*

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.1.33 When dew point alarm is active

This parameter is used to define the reaction to an active dew point alarm.

Option	
Activate Building Protection cooling	The operating mode *Building Protection cooling* is activated to prevent dew from forming on refrigerant lines (→ parameter *Building Protection cooling setpoint*). The operating mode *Building Protection cooling* is valid until the dew point alarm is reset.
Deactivate cooling	The type of operation *Cooling* is deactivated to prevent dew from forming on refrigerant lines. The type of operation *Cooling* is deactivated until the dew point alarm is reset.

Prerequisites for visibility
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Dew point alarm"* \ Option *Yes*

7.2.9.2 Parameter windows Controller settings

The following settings can be made in this parameter window:

- Set control parameters (when the basic or additional stages are freely configured)
- Enable status Group Objects
- Specify the send behavior of control values
- Set basic load
- Specify temperature limitations

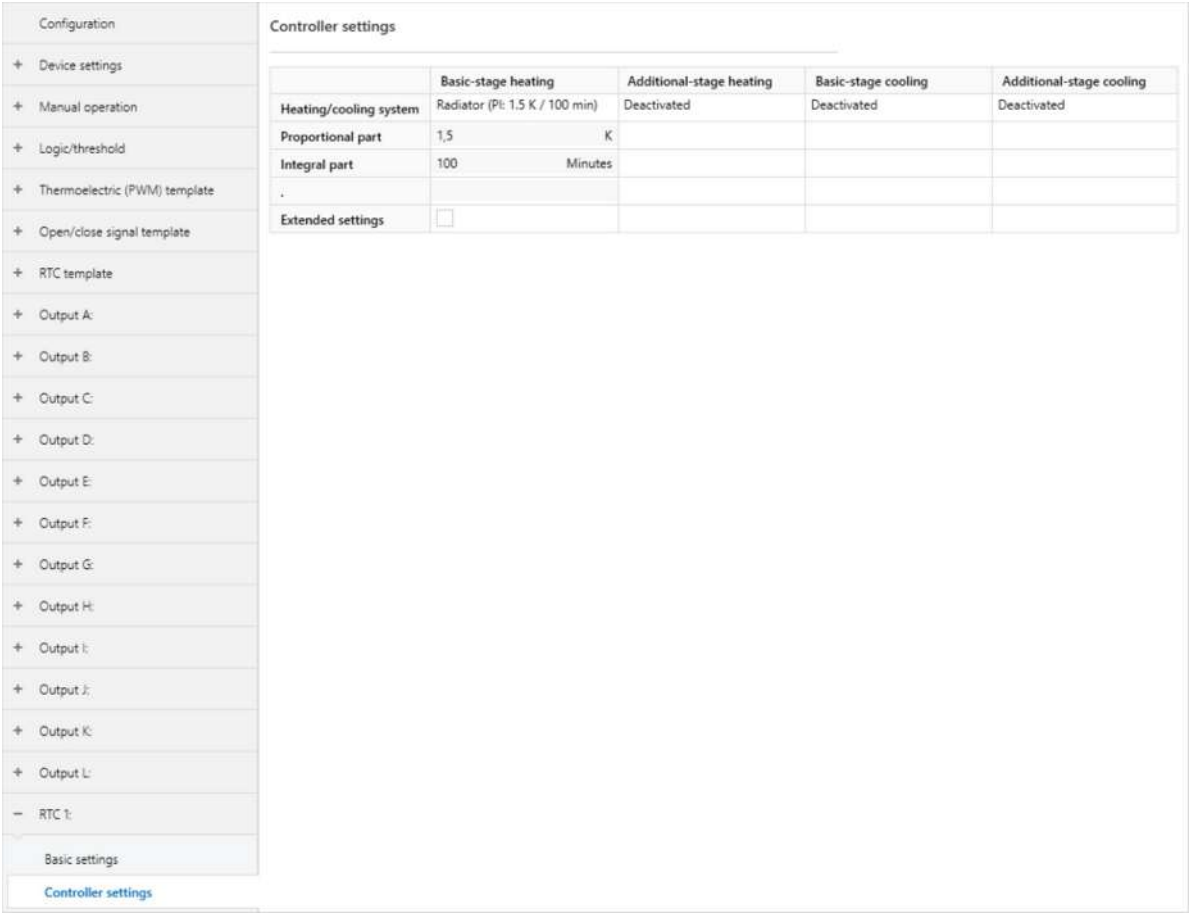


Fig. 29: Parameter window Controller settings

This parameter window includes the following parameters:

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Enable RTC x* \ Option *Yes*
  - Parameter *RTC x Template* \ Option *No*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Controller settings* \ Option *Yes*

7.2.9.2.1 Heating/cooling system

This parameter permanently adopts the settings for the basic and additional stages from the parameter window *Basic settings*.

Option	
Fixed	The setting is taken from the specifications in other parameters and cannot be changed here.

7.2.9.2.2 Proportional part

This parameter is used to permanently preset the proportional part of the PI control according to the use of the basic and additional stages. If the corresponding basic or additional stage is freely configurable, the value can be changed.

Note
The default value depends on the operating mode (Heating or Cooling).

Option
0 … 25 K

7.2.9.2.3 Integral part

This parameter is used to permanently preset the integral part of the PI control according to the use of the basic and additional stages. If the corresponding basic or additional stage is freely configurable, the value can be changed.

Option
0 … 100 … 240 min

7.2.9.2.4 Additional-stage setpoint difference

This parameter determines how far from the setpoint temperature the additional stage remains active.

Option
0 … 2 … 15 K

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Additional-stage heating* \ all options except *Deactivated*
    Or
  - Parameter *Additional-stage cooling* \ all options except *Deactivated*

7.2.9.2.5 Extended settings

This parameter is used to display the extended settings for the parameter window.

Note
The modified settings for the dependent parameters are only valid if the dependent parameters are shown.

Option	
No	The extended settings are not shown. The corresponding parameters are used with the default values. Changes to the default values are discarded.
Yes	The following dependent parameters are displayed: • *Send control value after a change of* • *Send control value cyclically* • *Maximum control value* • *Basic load* • *Temperature limitation*

7.2.9.2.6 Send control value after a change of

This parameter is used to define the minimum change in the control value (based on the last value sent) required for the output value to be sent on the bus (ABB i-bus® KNX).

Option
2 %
5 %
10 %

7.2.9.2.7 Send control value cyclically

This parameter is used to define whether the control value is sent cyclically.

Option	
No	The control value is sent only on change.
Yes	The following dependent parameters are displayed: • *Sending cycle* • *Send inactive control value cyclically*

7.2.9.2.8 Sending cycle

This parameter is used to define the cycle for sending the control value.

Option
0 … 15 … 60 min

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Controller settings* \ Parameter *Send control value cyclically* \ Option *Yes*

7.2.9.2.9 Send inactive control value cyclically

This parameter is used to define whether the control value for the inactive type of operation is sent cyclically.

Note
Cyclical sending increases the bus load. The control value of the inactive type of operation should only be sent cyclically if monitoring of cyclical sending between two devices is absolutely necessary.

Note
In systems that only have one control value input for heating and cooling, the control value Group Objects for heating and cooling must be connected to the same input Group Object. If the *Yes* option is selected in this parameter, the control values for the active and inactive type of operation overwrite each other.

Option
No
Yes

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Controller settings* \ Parameter *Send control value cyclically* \ Option *Yes*

7.2.9.2.10 Maximum control value

This parameter is used to define the maximum control value. The control is not allowed to exceed the maximum control value, even if the controller calculates a higher control value.

Option
0 … 100 %

7.2.9.2.11 Basic load

This parameter is used to define the use of a basic load.

The basic load is used to specify a minimum control value. The control is not allowed to drop below the basic load, even if the controller calculates a lower control value.

> **Note**
> The basic load can be set individually for each heating and cooling stage.
> The basic load is always activated jointly for all stages, but it is applicable only to the active type of operation (*Heating* or *Cooling*). The basic load of the inactive type of operation is not taken into account when the type of operation is changed.

Option	
Inactive	The basic load is not used.
Always active	The basic load is always active. The following dependent parameters are displayed: • *Minimum control value [Basic load]*
Via Group Object	The basic load is activated or deactivated via a Group Object. The following dependent parameters are displayed: • *Minimum control value [Basic load]* The following dependent Group Objects are displayed: • *Basic load*

7.2.9.2.12 Minimum control value [Basic load]

This parameter is used to define the minimum control value (basic load) for the controller.

Option
0 ... 100 %

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Controller settings* \ Parameter *Basic load* \ all options except *Inactive*

7.2.9.2.13 Hysteresis setpoint difference

This parameter is used to define the double-sided hysteresis (fluctuation range of the 2-point controller around the setpoint) of the PI control.

More information: → Hysteresis, Page 170.

Option
0.5 ... 1 ... 4 K

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Additional-stage heating* \ Option *Electric heater (2-point)*
    Or
  - Parameter *Additional-stage cooling* \ Option *Additional fan (2-point)*
- Parameter window *RTC x* \ Parameter window *Controller settings* \ Parameter *Extended settings* \ Option *Yes*

7.2.9.2.14 Temperature limitation

This parameter is used to define whether the temperature limitation is activated.

Option	
No	The temperature limitation is not activated.
Yes	The following dependent parameters are displayed: • *Limit temperature* • *Single-sided hysteresis* The following dependent Group Objects are displayed: • *Basic-stage heating limit temperature* • *Additional-stage heating limit temperature* • *Basic-stage cooling limit temperature* • *Additional-stage cooling limit temperature*

7.2.9.2.15 Limit temperature

This parameter is used to define the limit temperature for the types of operation. When the defined limit temperature is reached, the controller sets the control value to 0.

Note
The default value depends on the operating mode (Heating or Cooling).

Option
5 … 90 °C

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Controller settings* \ Parameter *Temperature limitation* \ Option *Yes*

7.2.9.2.16 Single-sided hysteresis

This parameter is used to define the single-sided hysteresis for the limit temperature. The hysteresis specifies how far the temperature must drop below the limit temperature (*Heating*) or exceed the limit temperature (*Cooling*) before the controller becomes active again.

More information: → Hysteresis, Page 170.

Option
0.5 … 5 … 25 K

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Controller settings* \ Parameter *Temperature limitation* \ Option *Yes*

7.2.9.3 Parameter windows Setpoint manager

The following settings can be made in this parameter window:

- Define the base setpoint
- Enable the Standby and Economy operating modes
- Specify setpoint temperatures
- Enable Setpoint Group Objects

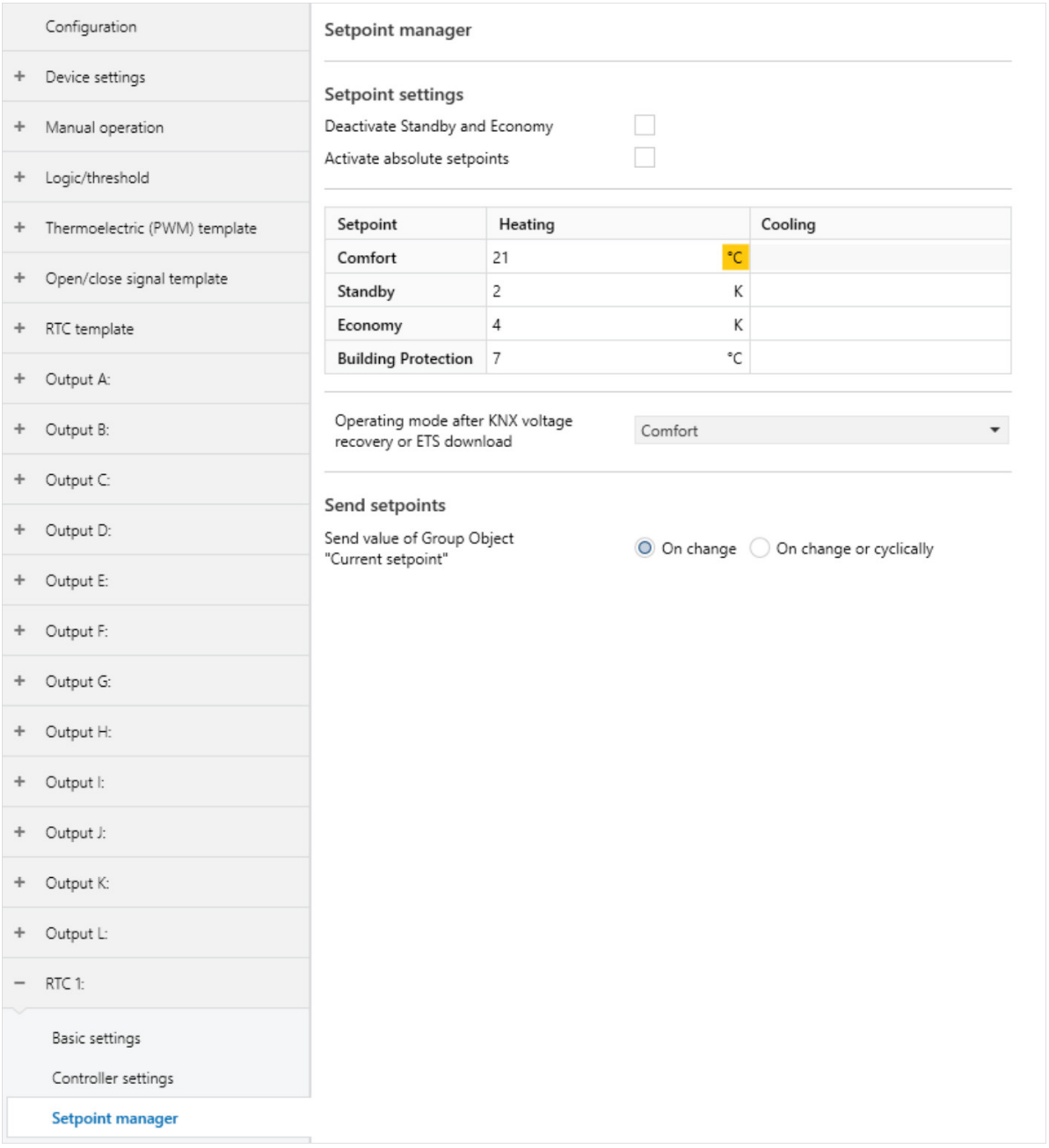


Fig. 30: Parameter window Setpoint manager

This parameter window includes the following parameters:

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Enable RTC x* \ Option *Yes*
  - Parameter *RTC x Template* \ Option *No*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*

  Or
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.3.1 Comfort heating setpoint = Comfort cooling setpoint

This parameter is used to define whether a common setpoint temperature is used for *Comfort heating* and *Comfort cooling*.

Option	
No	Two different setpoint temperatures (setpoints) can be set for *Comfort heating* and *Comfort cooling*. The following dependent parameters are displayed: • *Comfort cooling setpoint* • *Base setpoint is*
Yes	A common setpoint temperature is used for *Comfort heating* and *Comfort cooling*. The following dependent parameters are displayed: • *Hysteresis for heating/cooling changeover*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.3.2 Hysteresis for heating/cooling changeover

This parameter is used to define the hysteresis for automatic changeover between heating and cooling in *Comfort* operating mode if a common setpoint is used for *Comfort heating* and *Comfort cooling*.

Note
Automatic changeover between heating and cooling only occurs if the parameter *Heating/cooling changeover* is set to the option *Automatic*.

	Type of operation
Actual temperature > (setpoint + hysteresis)	Cooling
Actual temperature < (setpoint – hysteresis)	Heating

Tab. 13: Changing over heating/cooling

Option
0.5 … 1.0 … 10.0 °C

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Heating/cooling changeover* \ Option *Automatic*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Comfort heating set-point = Comfort cooling setpoint* \ Option *Yes*

7.2.9.3.3 Deactivate Standby and Economy

This parameter is used to deactivate the Standby and Economy operating modes.

Option	
No	The following dependent parameters are displayed: • *Standby heating setpoint* • *Economy heating setpoint* • *Standby cooling setpoint* • *Economy cooling setpoint* • *Operating mode after KNX voltage recovery or ETS download* • *Enable Group Object "Heating setpoints" (DPT 275.100)* • *Enable Group Object "Cooling setpoints" (DPT 275.100)* The following dependent Group Objects are displayed: • *Operating mode (main/secondary)*
Yes	The operating modes Standby and Economy are deactivated; received values for Standby or Economy are discarded. Control is possible only in the operating modes Comfort and Building Protection.

7.2.9.3.4 Activate absolute setpoints

This parameter is used to define whether the Standby and Economy setpoints are shown as absolute values.

Option	
No	The Standby and Economy setpoints are defined as a relative temperature difference from the Comfort setpoint. When adjusting the base setpoint or when shifting setpoints, the Standby and Economy setpoints are also adjusted while maintaining the defined temperature difference.
Yes	The Standby and Economy setpoints are defined as absolute temperature values. When adjusting the base setpoint or when shifting setpoints, no adjustment is made to the Standby, Economy and Building Protection setpoints. The setpoints for all operating modes and types of operation can be separately adjusted via Group Objects, → parameter *Enable Setpoint Group Objects*.

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Heating/cooling changeover* \ Option *Via Group Object*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*

7.2.9.3.5 Enable Setpoint Group Objects

This parameter enables the individual setpoint Group Objects for all operating modes and types of operation.

Option	
No	The Group Objects will not be enabled.
Yes	The following dependent Group Objects are displayed: • *Comfort heating setpoint* • *Standby heating setpoint* • *Economy heating setpoint* • *Building Protection heating setpoint* • *Comfort cooling setpoint* • *Standby cooling setpoint* • *Economy cooling setpoint* • *Building Protection cooling setpoint*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Activate absolute setpoints* \ Option *Yes*

7.2.9.3.6 Comfort heating setpoint

This parameter is used to define the setpoint temperature for the *Comfort heating* operating mode.

Option
5 … 21 … 40 °C

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*

7.2.9.3.7 Comfort cooling setpoint

This parameter is used to define the setpoint temperature for the *Comfort cooling* operating mode.

Option
10 … 25 … 40 °C

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.3.8 Standby heating setpoint

This parameter is used to define the setpoint temperature for the *Standby heating* operating mode.

Note
The possible options and the default values depend on the selection made in the parameter *Activate absolute setpoints*.

Option
5 … 19 … 40 °C
0.5 … 2 … 25 K

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic Settings* \ Parameter Basic-stage heating \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Enable Standby and Economy* \ Option *No*

7.2.9.3.9 Standby cooling setpoint

This parameter is used to define the setpoint temperature for the *Standby cooling* operating mode.

Note
The possible options and the default values depend on the selection made in the parameter *Activate absolute setpoints*.

Option
5 … 27 … 40 °C
0.5 … 2 … 25 K

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Enable Standby and Economy* \ Option *No*

7.2.9.3.10 Economy heating setpoint

This parameter is used to define the setpoint temperature for the *Economy heating* operating mode.

Note
The possible options and the default values depend on the selection made in the parameter *Activate absolute setpoints*.

Option
5 … 17 … 40 °C
0.5 … 4 … 25 K

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic Settings* \ Parameter Basic-stage heating \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Enable Standby and Economy* \ Option *No*

7.2.9.3.11 Economy cooling setpoint

This parameter is used to define the setpoint temperature for the *Economy cooling* operating mode.

Note
The possible options and the default values depend on the selection made in the parameter *Activate absolute setpoints*.

Option
5 … 29 … 40 °C
0.5 … 4 … 25 K

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Enable Standby and Economy* \ Option *No*

7.2.9.3.12 Building Protection heating setpoint

This parameter is used to define the setpoint temperature for the *Building Protection Heating* operating mode.

Option
5 … 7 … 40 °C

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*

7.2.9.3.13 Building Protection cooling setpoint

This parameter is used to define the setpoint temperature for the *Building Protection Cooling* operating mode.

Option
5 … 35 … 45 °C

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.3.14 Operating mode after KNX voltage recovery or ETS download

This parameter is used to define which operating mode is activated after KNX voltage recovery or ETS download. The operating mode remains active until a new operating mode is set.

Note
The operating mode should be defined during the planning phase. If the operating mode is defined incorrectly, this might reduce comfort or increase energy consumption.

Option
Comfort
Standby
Economy
Building Protection

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*

7.2.9.3.15 Base setpoint is

This parameter is used to define which value corresponds to the base setpoint.

Note
If only the *Heating* operating mode or *Cooling* operating mode is configured, the base setpoint corresponds to the respective *Comfort* setpoint.

If the parameter *Comfort heating setpoint = Comfort cooling setpoint* is set to the option *Yes*, this parameter is hidden and permanently set to the option *Comfort heating setpoint*.

Option	
Comfort heating setpoint	The base setpoint corresponds to the setpoint for Comfort heating. If the base setpoint is changed via the Group Object *Base setpoint Comfort heating*, the setpoint for Comfort cooling shifts as well. The relative distances between the two Comfort values remain unchanged. The following dependent Group Objects are displayed: • *Base setpoint Comfort heating*
Comfort cooling setpoint	The base setpoint corresponds to the setpoint for Comfort cooling. If the base setpoint is changed via the Group Object *Base setpoint Comfort cooling*, the setpoint for Comfort heating shifts as well. The relative distances between the two Comfort values remain unchanged. The following dependent Group Objects are displayed: • *Base setpoint Comfort cooling*

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Comfort heating setpoint = Comfort cooling setpoint* \ Option *No*

7.2.9.3.16 Send value of Group Object "Current setpoint"

This parameter is used to define when the setpoint currently valid is sent via the Group Object *Current setpoint (main/secondary)*.

Option	
On change	The value is sent on change.
On change or cyclically	The value is sent on change or cyclically. The cycle time can be set. The cycle time is restarted each time a value is sent, including if the value is sent on change. The following dependent parameters are displayed: • *Sending cycle*

7.2.9.3.17 Sending cycle

This parameter is used to define the cycle in which the Group Object value is sent.

Note
The possible options and default values depend on the higher-level parameter.

Option
00:00:30 … 01:00:00 … 24:00:00 hh:mm:ss
00:01 … 00:05 … 23:59 hh:mm
1 … 15 … 240 min

Prerequisites for visibility

- The parameter appears at various points in the application. The visibility is dependent on the application and the higher-level parameter.

7.2.9.4 Parameter windows Setpoint adjustment

The following settings can be made in this parameter window:

- Define minimum and maximum setpoint adjustment
- Define setpoint adjustment reset

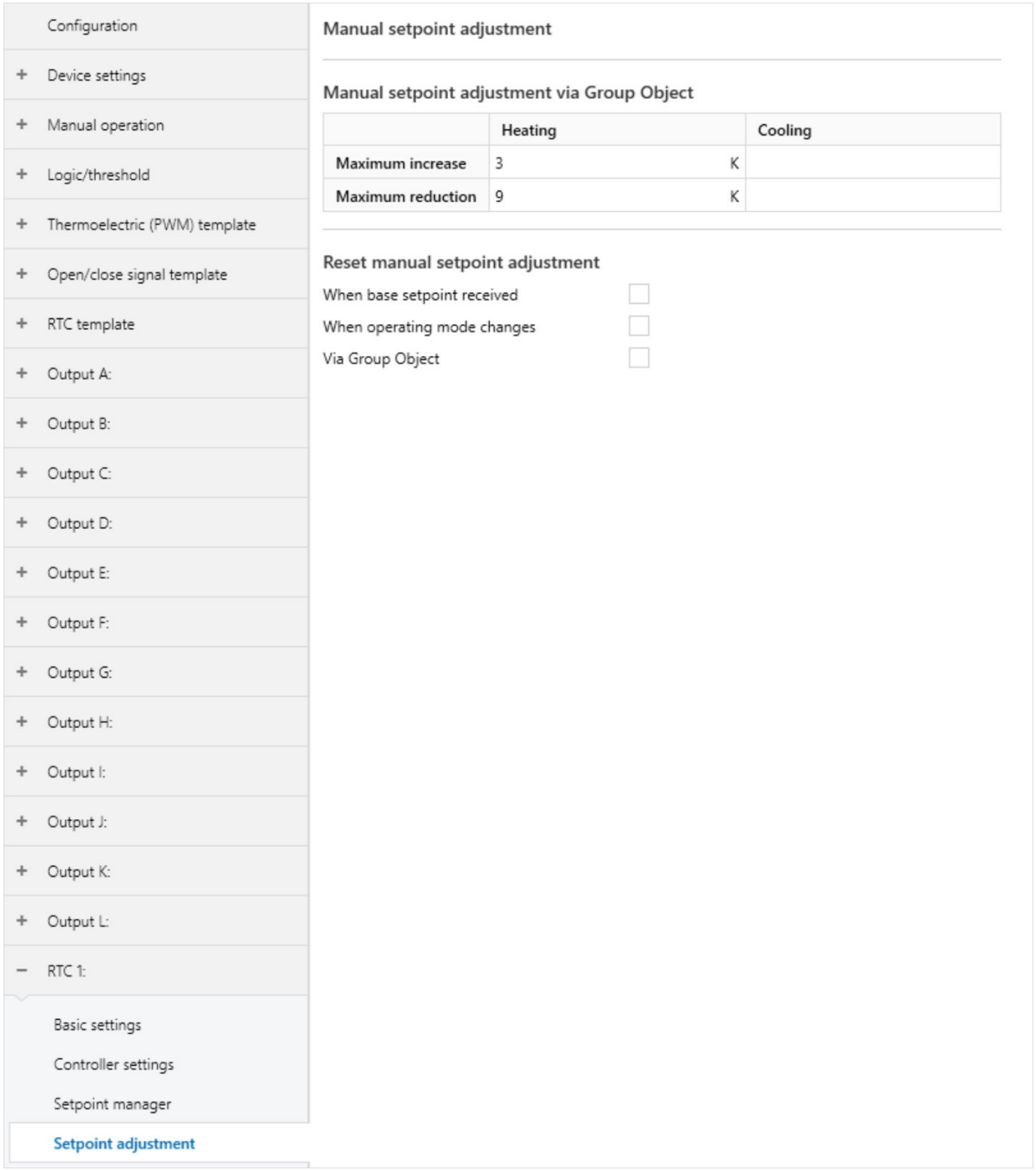


Fig. 31: Parameter window Setpoint adjustment

This parameter window includes the following parameters:

→ Maximum heating increase, Page 137
→ Maximum cooling increase, Page 137
→ Maximum heating decrease, Page 137
→ Maximum cooling decrease, Page 137
→ Reset manual setpoint adjustment when base setpoint received, Page 137
→ Reset manual setpoint adjustment when operating mode changes, Page 138
→ Reset manual setpoint adjustment via Group Object, Page 138

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Enable RTC x* \ Option *Yes*
  - Parameter *RTC x Template* \ Option *No*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*

  Or
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

7.2.9.4.1 Maximum heating increase

This parameter is used to define how far the *Comfort heating* setpoint can be increased via manual setpoint adjustment.

Option
0 … 3 … 9 K

7.2.9.4.2 Maximum cooling increase

This parameter is used to define how far the *Comfort cooling* setpoint can be increased via manual setpoint adjustment.

Option
0 … 9 K

7.2.9.4.3 Maximum heating decrease

This parameter is used to define how far the *Comfort heating* setpoint can be decreased via manual setpoint adjustment.

Option
0 … 9 K

7.2.9.4.4 Maximum cooling decrease

This parameter is used to define how far the *Comfort cooling* setpoint can be decreased via manual setpoint adjustment.

Option
0 … 3 … 9 K

7.2.9.4.5 Reset manual setpoint adjustment when base setpoint received

This parameter is used to define whether manual setpoint adjustment is reset when a new base setpoint is received on one of the following Group Objects:

- *Base setpoint Comfort heating*
- *Base setpoint Comfort cooling*

Example
- Old base setpoint: 21 °C
- Manual adjustment: 1.5 K
- Old temperature setpoint: 22.5 °C

Receipt of new base setpoint:
- New base setpoint: 18 °C
- New temperature setpoint
  - Without manual adjustment reset: 19.5 °C
  - With manual adjustment reset: 18 °C

Option	
No	Manual setpoint adjustment is not reset.
Yes	Manual setpoint adjustment is reset. The new setpoint corresponds to the received base setpoint.

7.2.9.4.6 Reset manual setpoint adjustment when operating mode changes

This parameter is used to define whether manual setpoint adjustment is reset when the operating mode changes.

Option	
No	Manual setpoint adjustment is not reset.
Yes	Manual setpoint adjustment is reset. The new setpoint corresponds to the setpoint for the active operating mode. Shifts in the base setpoint are taken into account.

7.2.9.4.7 Reset manual setpoint adjustment via Group Object

This parameter is used to define whether manual setpoint adjustment is reset via the following Group Object:
- *Reset setpoint adjustment*

Option	
No	Manual setpoint adjustment is not reset.
Yes	Manual setpoint adjustment is reset. The new setpoint corresponds to the base setpoint. The following dependent Group Objects are displayed: • *Reset setpoint adjustment*

7.2.9.5 Parameter windows Temperature acquisition

The following settings can be made in this parameter window:

- Enable temperature monitoring
- Specify type of operation and control value during a fault

More information: → Temperature acquisition monitoring, Page 29.

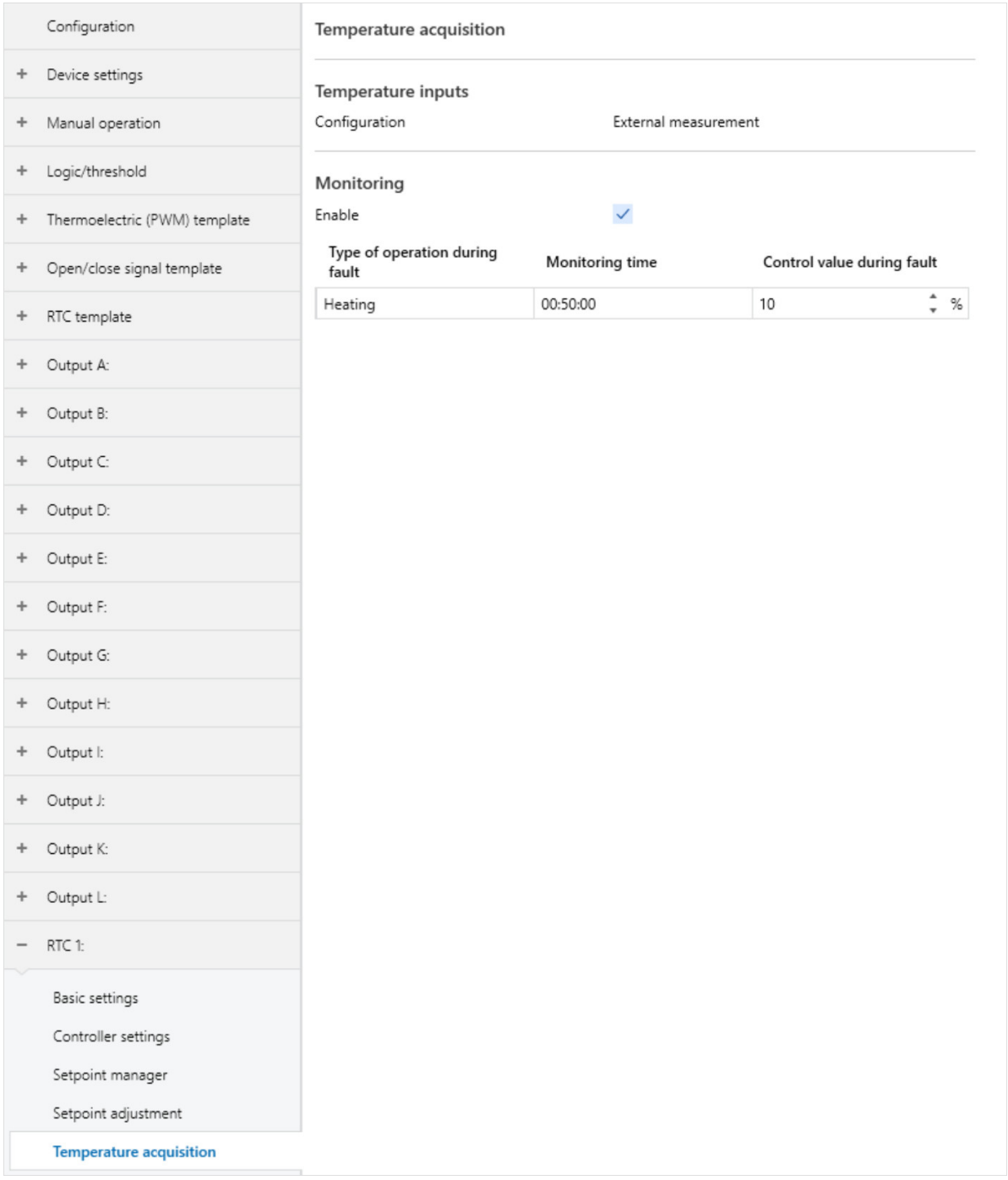


Fig. 32: Parameter window Temperature acquisition

This parameter window includes the following parameters:

→ Enable monitoring, Page 140
→ Type of operation during fault, Page 140
→ Monitoring time, Page 141
→ Control value during fault, Page 141

Prerequisites for visibility

- Parameter window *Configuration*
  - Parameter *Enable RTC x* \ Option *Yes*
  - Parameter *RTC x Template* \ Option *No*

7.2.9.5.1 Enable monitoring

This parameter enables monitoring of the actual temperature.

Option	
No	Monitoring of the actual temperature is not enabled.
Yes	The following dependent parameters are displayed: • *Type of operation during fault* • *Monitoring time* • *Control value during fault* The following dependent Group Objects are displayed: • *Fault: actual temperature*

7.2.9.5.2 Type of operation during fault

This parameter is used to define which type of operation is set if the actual temperature fails. The options available depend on the controller configuration. If only one type of operation (*Heating* or *Cooling*) is configured, or if heating/cooling is changed over via a Group Object (→ parameter *Heating/cooling changeover*), the current type of operation remains active if the actual temperature fails.

Option	
Heating	During a fault, the type of operation *Heating* is set. This option is available only if the controller is configured for the type of operation *Heating* and the option *Automatic* is selected in the parameter *Heating/cooling changeover*.
Cooling	During a fault, the type of operation *Cooling* is set. This option is available only if the controller is configured for the type of operation *Cooling* and the option *Automatic* is selected in the parameter *Heating/cooling changeover*.
Via Group Object	The type of operation during a fault is set by the Group Object *Heating/cooling changeover*. This option is permanently set if the controller is configured for both types of operation (*Heating* and *Cooling*) and the option *Via Group Object* is selected in the parameter *Heating/cooling changeover*.

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Heating/cooling changeover* \ Option *Automatic*
- Parameter window *RTC x* \ Parameter window *Temperature acquisition* \ Parameter *Enable monitoring of actual value* \ Option *Yes*

7.2.9.5.3 Monitoring time

This parameter is used to define the period during which a value must be received on the Group Object *External actual temperature*. Each time a value is received on the Group Object *External actual temperature*, the monitoring time is reset.

Option
00:00:10 … 00:50:00 … 18:12:15 hh:mm:ss

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Heating/cooling changeover* \ Option *Automatic*
- Parameter window *RTC x* \ Parameter window *Temperature acquisition* \ Parameter *Enable monitoring of actual value* \ Option *Yes*

7.2.9.5.4 Control value during fault

This parameter is used to define the control value that becomes active if the actual temperature fails. This control value applies until a value is received on the Group Object *External actual temperature*.

Option
0 …10…100 %

Prerequisites for visibility

- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Heating/cooling changeover* \ Option *Automatic*
- Parameter window *RTC x* \ Parameter window *Temperature acquisition* \ Parameter *Enable monitoring of actual value* \ Option *Yes*

8 Group Objects

8.1 Overview of Group Objects

Function	Group Object name	Data point type	Length	Flags
Enable/Block Manual operation	Central – Manual operation:	DPT 1.003	1 bit	C W
Activate service	Central – Service:	DPT 2.001	2 bit	C W
Activate valve purge	Output X – Valve:	DPT 1.017	1 bit	C W
Activate/block control value override	Output X – Valve	DPT 1.003	1 bit	C W
Additional-stage cooling limit temperature	RTC x – Room temperature controller	DPT 9.001	2 bytes	C W U
Additional-stage heating limit temperature	RTC x – Room temperature controller	DPT 9.001	2 bytes	C W U
Base setpoint Comfort cooling	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Base setpoint Comfort heating	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Basic load	RTC x – Room temperature controller	DPT 1.011	1 bit	C W U
Basic-stage cooling limit temperature	RTC x – Room temperature controller	DPT 9.001	2 bytes	C W U
Basic-stage heating limit temperature	RTC x – Room temperature controller	DPT 9.001	2 bytes	C W U
Block/enable cooling stages separately	RTC x – Room temperature controller:	DPT 20.116	1 byte	C W T U
Block/enable cooling	RTC x – Room temperature controller:	DPT 1.003	1 bit	C W T U
Block/enable heating stages separately	RTC x – Room temperature controller:	DPT 20.116	1 byte	C W T U
Block/enable heating	RTC x – Room temperature controller:	DPT 1.003	1 bit	C W T U
Building Protection cooling setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Building Protection heating setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Change lower threshold	Logic/threshold x – Threshold:	DPT 13.010	4 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 13.013	4 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 14.019	4 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 14.056	4 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 5.001	1 byte	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 5.010	1 byte	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 7.001	2 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 9.001	2 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 9.004	2 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 9.005	2 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 9.021	2 bytes	C W
Change lower threshold	Logic/threshold x – Threshold:	DPT 9.024	2 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 13.010	4 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 13.013	4 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 14.019	4 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 14.056	4 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 5.001	1 byte	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 5.010	1 byte	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 7.001	2 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 9.001	2 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 9.004	2 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 9.005	2 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 9.021	2 bytes	C W
Change upper threshold	Logic/threshold x – Threshold:	DPT 9.024	2 bytes	C W
Comfort heating setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Comfort cooling setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Confirm setpoint adjustment (main/ secondary)	RTC x – Room temperature controller:	DPT 9.002	2 bytes	C R T
Connection A	Logic/threshold x – Logic:	DPT 1.002	1 bit	C W
Connection B	Logic/threshold x – Logic:	DPT 1.002	1 bit	C W
Control value additional-stage cooling	RTC x – Room temperature controller:	DPT 5.001	1 byte	C T
Control value additional-stage heating	RTC x – Room temperature controller:	DPT 5.001	1 byte	C T
Control value basic-stage cooling	RTC x – Room temperature controller:	DPT 5.001	1 byte	C T
Control value basic-stage heating	RTC x – Room temperature controller:	DPT 5.001	1 byte	C T
Control value override	Output X – Valve	DPT 5.001	1 byte	C W
Control value	Output X – Valve:	DPT 5.001	1 byte	C W
Control value	Output X – Valve:	DPT 1.001	1 bit	C W
Controller synchronization of input	RTC x – Room temperature controller:	DPT 20.105	1 byte	C W U
Controller synchronization of output	RTC x – Room temperature controller:	DPT 20.105	1 byte	C R T
Cooling active	RTC x – Room temperature controller:	DPT 1.011	1 bit	C T
Cooling setpoints	RTC x – Room temperature controller:	DPT 275.100	8 bytes	C W T U
Current operating mode (main/secondary)	RTC x – Room temperature controller:	DPT 20.102	1 byte	C R T
Current setpoint (main/secondary)	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C R T
Deactivate manual operation	Central – Manual operation:	DPT 1.017	1 bit	C W
Dew point alarm	RTC x – Room temperature controller:	DPT 1.019	1 bit	C W T U

Function	Group Object name	Data point type	Length	Flags
Economy cooling setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Economy heating setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
External actual temperature	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
External pump control x	Central – Pump control	DPT 1.001	1 bit	C W
Fault: actual temperature	RTC x – Room temperature controller:	DPT 1.005	1 bit	C R T
Forced operation 1 bit	Output X – Valve	DPT 1.003	1 bit	C W
Forced operation 2 bit	Output X – Valve:	DPT 2.001	2 bit	C W
Heating active	RTC x – Room temperature controller:	DPT 1.011	1 bit	C T
Heating setpoints	RTC x – Room temperature controller:	DPT 275.100	8 bytes	C W T U
Heating/cooling changeover	RTC x – Room temperature controller:	DPT 1.100	1 bit	C W U
In operation	Central – General:	DPT 1.002	1 bit	C R T
Maximum external control value	Central – Maximum control value	DPT 5.001	1 byte	C W
Operating mode (main/secondary)	RTC x – Room temperature controller	DPT 20.102	1 byte	C W T U
Operating mode override	RTC x – Room temperature controller:	DPT 20.102	1 byte	C W T U
Presence detector	RTC x – Room temperature controller:	DPT 1.018	1 bit	C W T U
Pump on/off x	Central – Pump control:	DPT 1.002	1 bit	C R T
Request setpoint adjustment (main/secondary)	RTC x – Room temperature controller:	DPT 9.002	2 bytes	C W T U
Request status values	Central – General:	DPT 1.017	1 bit	C W
Reset setpoint adjustment	RTC x – Room temperature controller:	DPT 1.015	1 bit	C W U
Reset valve output fault	Central – General:	DPT 1.017	1 bit	C W
Standby cooling setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Standby heating setpoint	RTC x – Room temperature controller:	DPT 9.001	2 bytes	C W T U
Status byte	Output X – Valve:	DPT 22.102	1 byte	C R T
Status Control value	Output X – Valve:	DPT 5.001	1 byte	C R T
Status Control value	Output X – Valve:	DPT 1.002	1 bit	C R T
Status Fault supply voltage	Central – General:	DPT 1.005	1 bit	C R T
Status Fault valve	Output X – Valve:	DPT 1.005	1 bit	C R T
Status Heating/cooling changeover	RTC x – Room temperature controller:	DPT 1.100	1 bit	C T
Status Input value between thresholds	Logic/threshold x – Threshold:	DPT 1.002	1 bit	C R T
Status Manual operation	Central – Manual operation:	DPT 1.011	1 bit	C R T
Status Maximum control value	Central – Maximum control value	DPT 5.001	1 byte	C R T
Status Result [Logic]	Logic/threshold x – Logic:	DPT 1.002	1 bit	C R T
Status Result [threshold]	Logic/threshold x – Threshold:	DPT 1.002	1 bit	C R T
Status RHCC	RTC x – Room temperature controller:	DPT 22.101	16 bit	C R T
Status RTC Controller (main/secondary)	RTC x – Room temperature controller:	DPT 22.103	16 bit	C R T
Status RTSM Controller (main/secondary)	RTC x – Room temperature controller:	DPT 21.107	1 byte	C R T
Status Safety mode	Output X – Valve:	DPT 1.005	1 bit	C R T
Status Service	Central – Service:	DPT 1.002	1 bit	C R T
Status Valve purge	Output X – Valve:	DPT 1.002	1 bit	C R T
Summer/winter mode	Central – General:	DPT 1.002	1 bit	C W
Threshold input	Logic/threshold x – Threshold:	DPT 13.010	4 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 13.013	4 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 14.019	4 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 14.056	4 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 5.001	1 byte	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 5.010	1 byte	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 7.001	2 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 9.001	2 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 9.004	2 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 9.005	2 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 9.021	2 bytes	C W T U
Threshold input	Logic/threshold x – Threshold:	DPT 9.024	2 bytes	C W T U
Window contact	RTC x – Room temperature controller:	DPT 1.019	1 bit	C W T U

8.2 Device settings Group Objects

Function	Group Object name	Data point type	Length	Flags
In operation	**Central – General:**	**DPT 1.002**	**1 bit**	**C R T**

This Group Object cyclically sends an In operation telegram on the bus (ABB i-bus® KNX). The sending cycle is set in parameter *Sending cycle*.
The telegram value depends on the setting in the parameter *Enable Group Object "In operation"*.
Telegram value:
- 1 = Device in operation
- 0 = Device in operation

Note
Readiness can be monitored by another KNX device using this Group Object. If a telegram is not received, the sending device could be faulty or the bus cable to the transmitting device could be interrupted.

Prerequisites for visibility
- Parameter window *Device settings* \ Parameter *Enable Group Object "In operation"* \ Option *Yes, send value 0 cyclically* / *Yes, send value 1 cyclically*

Function	Group Object name	Data point type	Length	Flags
Request status values	**Central – General:**	**DPT 1.017**	**1 bit**	**C W**

If a telegram is received on this Group Object, the values of the Status Group Objects are sent on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Send status values
- 0 = Send status values

Prerequisites for visibility
- Parameter window *Device settings* \ Parameter *Enable Group Object "Request status values"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Reset valve output fault	**Central – General:**	**DPT 1.017**	**1 bit**	**C W**

This Group Object is used to reset a fault message on the valve output via the bus (ABB i-bus® KNX) (reset). Resetting is successful only if the fault has been rectified.

Note
Before resetting the fault message, correct the cause of the fault (overload/short circuit or no supply voltage), otherwise the device may get damaged. Cyclical sending on this Group Object is not permitted.

Note
The fault message can also be reset via the membrane keypad, → Operating and display elements, Page 23

Telegram value:
- 1 = Reset fault
- 0 = Reset fault

Prerequisites for visibility
- Parameter window *Device settings* \ Parameter *Enable Group Object "Reset valve output fault"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Status Fault supply voltage	**Central – General:**	**DPT 1.005**	**1 bit**	**C R T**

This Group Object sends the status of the supply voltage (fault = no supply voltage, or no fault = supply voltage present) on the bus (ABB i-bus® KNX).

Note
This Group Object reports a fault if there is no supply voltage (VC/S 6.2.2.2: terminal pair 1/2, VC/S 12.2.2.2: terminal pairs 1/2 and 15/16). If the upper outputs (valve outputs A ... F) or the lower outputs (valve outputs G ... L) are not in use on the VC/S 12.2.2.2 and the corresponding supply voltage is not connected, the corresponding outputs should be deactivated (→ parameter *Output X Valve type*) to avoid unwanted fault messages.

Telegram value:
- Dependent on the setting in the parameter *Value of Group Object "Status Fault supply voltage"*

Prerequisites for visibility
- Parameter window *Device settings* \ Parameter *Enable Group Object "Status Fault supply voltage"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Summer/winter mode	**Central – General:**	**DPT 1.002**	**1 bit**	**C W**

This Group Object is used to receive the "Summer mode" or "Winter mode" information via the bus (ABB i-bus® KNX).

Telegram value:
- Dependent on the setting in the parameter *Value of Group Object "Summer/winter mode"*.

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Summer/winter mode* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Activate service	**Central – Service:**	**DPT 2.001**	**2 bit**	**C W**

This Group Object is used to activate or deactivate Service via the bus (ABB i-bus® KNX).
More information: → Function Service, Page 31.

Telegram value (bit 1 | bit 0):
- 0 | 0 = Service inactive
- 0 | 1 = Service inactive
- 1 | 0 = Service active "OFF" (= 0 %, valve closed)
- 1 | 1 = Service active "ON" (= 100 %, valve open)

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Service* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Status Service	**Central – Service:**	**DPT 1.002**	**1 bit**	**C R T**

This Group Object sends the Service status on the bus (ABB i-bus® KNX). The value of this Group Object is sent on change or on request.

Telegram value:
- 1 = Service active
- 0 = Service inactive

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Service* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
External pump control x	**Central – Pump control**	**DPT 1.001**	**1 bit**	**C W**
This Group Object is used to receive an external pump activation (cascade activation) via the bus (ABB i-bus® KNX). More information: → Cascading, Page 171. Telegram value: • 1 = On • 0 = Off Prerequisites for visibility • Parameter window *Configuration* \ Parameter *Enable function: Pump control x* \ Option *Yes* • Parameter window *Device settings* \ Parameter window *Pump control x* \ Parameter *Enable Group Object "External pump control x"* \ Option *Yes*				
Pump on/off x	**Central – Pump control:**	**DPT 1.002**	**1 bit**	**C R T**
This Group Object is used to switch the pump via the bus (ABB i-bus® KNX). Telegram value: • Dependent on the setting in the parameter *Invert pump control* Prerequisites for visibility • Parameter window *Configuration* \ Parameter *Enable function: Pump control x* \ Option *Yes*				
Maximum external control value	**Central – Maximum control value**	**DPT 5.001**	**1 byte**	**C W**
This Group Object is used to receive the maximum external control value (cascade activation) via the bus (ABB i-bus® KNX). More information: → Cascading, Page 171. Telegram value: • 0 … 100 % Prerequisites for visibility • Parameter window *Configuration* \ Parameter *Enable function: Maximum control value* \ Option *Yes* • Parameter window *Device settings* \ Parameter window *Maximum control value* \ Parameter *Enable Group Object "Maximum external control value"* \ Option *Yes*				
Status Maximum control value	**Central – Maximum control value**	**DPT 5.001**	**1 byte**	**C R T**
This Group Object sends the maximum control value on the bus (ABB i-bus® KNX). Only those outputs for which the use of the function *Maximum control value* is activated are taken into account (→ parameter *Output X Maximum control value*). The send behavior depends on the setting in the parameter *Send value of Group Object "Status Maximum control value"*. More information: → Function Maximum control value, Page 31. Telegram value: • 0 … 100 % Prerequisites for visibility • Parameter window *Configuration* \ Parameter *Enable function: Maximum control value* \ Option *Yes*				

8.3 Group Objects Manual operation

Function	Group Object name	Data point type	Length	Flags
Status Manual operation	**Central – Manual operation:**	**DPT 1.011**	**1 bit**	**C R T**
This Group Object sends the status of the *Manual operation* mode on the bus (ABB i-bus® KNX). Telegram value: • 1 = Manual operation active • 0 = Manual operation inactive Prerequisites for visibility • Parameter window *Manual operation* \ Parameter *Enable manual operation* \ Option *Yes*				
Enable/Block Manual operation	**Central – Manual operation:**	**DPT 1.003**	**1 bit**	**C W**
This Group Object is used to enable or block *Manual operation* mode. If *Manual operation* mode is active, it will be deactivated and blocked with telegram value 0. If *Manual operation* mode was deactivated and blocked via this Group Object, the only way to reactivate it is via this Group Object. Telegram value: • 1 = Enable Manual operation • 0 = Deactivate and block Manual operation Prerequisites for visibility • Parameter window *Manual operation* \ Parameter *Enable manual operation* \ Option *Yes*				
Deactivate manual operation	**Central – Manual operation:**	**DPT 1.017**	**1 bit**	**C W**
This Group Object receives, via the bus (ABB i-bus® KNX), the command to deactivate Manual operation. Telegram value: • 1 = Deactivate Manual operation • 0 = Deactivate Manual operation Prerequisites for visibility • Parameter window *Manual operation* \ Parameter *Enable manual operation* \ Option *Yes*				

8.4 Group Objects Logic/threshold

Function	Group Object name	Data point type	Length	Flags
Connection A	**Logic/threshold x – Logic:**	**DPT 1.002**	**1 bit**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), an input value for the function *Logic*.

Telegram value:
- 1 = Logically true
- 0 = Logically false

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the sending Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE* / *1 bit Inverter*

Function	Group Object name	Data point type	Length	Flags
Connection B	**Logic/threshold x – Logic:**	**DPT 1.002**	**1 bit**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), an input value for the function *Logic*.

Telegram value:
- 1 = Logically true
- 0 = Logically false

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the sending Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE*

Function	Group Object name	Data point type	Length	Flags
Status Result [Logic]	**Logic/threshold x – Logic:**	**DPT 1.002**	**1 bit**	**C R T**

This Group Object sends the result of the function *Logic* on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Logically true
- 0 = Logically false

Note
The result can be inverted,→ parameter *Invert result*.

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *AND* / *OR* / *Exclusive OR* / *GATE* / *1 bit Inverter*

Function	Group Object name	Data point type	Length	Flags
Status Input value between thresholds	**Logic/threshold x – Threshold:**	**DPT 1.002**	**1 bit**	**C R T**

This Group Object sends the value 1 on the bus (ABB i-bus® KNX) if the input value for the function *Threshold* is between the thresholds.

Telegram value:
- 1 = Input value is between the thresholds (logically true)
- 0 = Input value is not between the thresholds (logically false)

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Monitor range between thresholds* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 13.010**	**4 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 2147483647 Wh

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Wh (DPT 13.010)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 13.013**	**4 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 2147483647 kWh

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *kWh (DPT 13.013)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 14.019**	**4 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 24 A

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *A (DPT 14.019)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 14.056**	**4 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 10000 W

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *W (DPT 14.056)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 5.001**	**1 byte**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
0 … 100 %

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Percent (DPT 5.001)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 5.010**	**1 byte**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 255

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Counter pulses (DPT 5.010)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 7.001**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 65535

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Counter pulses (DPT 7.001)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- −273 … 670760 °C

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Temperature (DPT 9.001)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 9.004**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 670760 lux

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Lux (DPT 9.004)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 9.005**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 100 m/s

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Wind (DPT 9.005)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 9.021**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 240000 mA

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *mA (DPT 9.021)*

Function	Group Object name	Data point type	Length	Flags
Threshold input	**Logic/threshold x – Threshold:**	**DPT 9.024**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the input value for the function *Threshold*.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 10 kW

Note
Prerequisites for automatically updating the Group Object:
- Parameter *Read input Group Objects after KNX voltage recovery or download* \ Option *Yes*
- The read flag is set for the Group Object

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *kW (DPT 9.024)*

Function	Group Object name	Data point type	Length	Flags
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 13.010**	**4 bytes**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 2147483647 Wh

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Wh (DPT 13.010)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 13.013**	**4 bytes**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 2147483647 kWh

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *kWh (DPT 13.013)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 14.019**	**4 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 24 A

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *A (DPT 14.019)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 14.056**	**4 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 10000 W

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *W (DPT 14.056)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 5.001**	**1 byte**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Percent (DPT 5.001)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 5.010**	**1 byte**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 255

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Counter pulses (DPT 5.010)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 7.001**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 65535

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Counter pulses (DPT 7.001)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 9.001**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- −273 … 670760 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Temperature (DPT 9.001)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 9.004**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 670760 lux

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Lux (DPT 9.004)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 9.005**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 100 m/s

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Wind (DPT 9.005)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 9.021**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 240000 mA

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *mA (DPT 9.021)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change upper threshold	**Logic/threshold x – Threshold:**	**DPT 9.024**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the upper threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 10 kW

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *kW (DPT 9.024)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 13.010**	**4 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 2147483647 Wh

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Wh (DPT 13.010)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 13.013**	**4 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 2147483647 kWh

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *kWh (DPT 13.013)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 14.019**	**4 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 24 A

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *A (DPT 14.019)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 14.056**	**4 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 10000 W

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *W (DPT 14.056)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 5.001**	**1 byte**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Percent (DPT 5.001)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 5.010**	**1 byte**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 255

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Counter pulses (DPT 5.010)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 7.001**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 65535

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Counter pulses (DPT 7.001)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags	
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 9.001**	**2 bytes**	**C**	**W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- −273 … 670760 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Temperature (DPT 9.001)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 9.004**	**2 bytes**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 670760 lux

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Lux (DPT 9.004)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 9.005**	**2 bytes**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 100 m/s

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *Wind (DPT 9.005)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 9.021**	**2 bytes**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 240000 mA

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *mA (DPT 9.021)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Change lower threshold	**Logic/threshold x – Threshold:**	**DPT 9.024**	**2 bytes**	**C W**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a new value for the lower threshold.
The data point type for the Group Object depends on the option selected in the parameter *Data point type of Group Object "Threshold input"*.

Telegram value:
- 0 … 10 kW

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x*
  - Parameter *Logic function* \ Option *Threshold*
  - Parameter *Data point type of Group Object "Threshold input"* \ Option *kW (DPT 9.024)*
  - Parameter *Change thresholds via Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Status Result [threshold]	**Logic/threshold x – Threshold:**	**DPT 1.002**	**1 bit**	**C R T**

This Group Object sends the result of the function *Threshold* on the bus (ABB i-bus® KNX).

Telegram value:
- Dependent on the settings in the following parameters:
  - *Result if upper threshold is exceeded*
  - *Result if lower threshold is dropped below*

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable function: Logic/threshold* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/threshold configuration* \ Parameter *Enable groups: Logic/threshold x-y* \ Option *Yes*
- Parameter window *Logic/threshold* \ Parameter window *Logic/Threshold x* \ Parameter *Logic function* \ Option *Threshold*

8.5 Group Objects Output X – Valve:

Function	Group Object name	Data point type	Length	Flags
Control value	**Output X – Valve:**	**DPT 5.001**	**1 byte**	**C W**

This Group Object is used to receive the control value (*Heating* or *Cooling*) from an external controller via the bus (ABB i-bus® KNX).

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*
  - Parameter *Output X Controller* \ Option *External*

Or
- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Open/close signal*
  - Parameter *Output X Controller* \ Option *External*
- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *DPT Group Object "Control value"* \ Option *Percent (DPT 5.001)*

Control value	Output X – Valve:	DPT 1.001	1 bit	C W

This Group Object is used to receive the control value (*Heating* or *Cooling*) from an external controller via the bus (ABB i-bus® KNX).

Telegram value:
- Dependent on the setting in the parameter *Valve drive operating principle, de-energized*

Prerequisites for visibility
- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ Option *Open/close signal*
  - Parameter *Output X Controller* \ Option *External*
  - Parameter *Enable function: Summer/winter mode* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *DPT Group Object "Control value"* \ Option *1 bit (DPT 1.001)*

Status Control value	Output X – Valve:	DPT 5.001	1 byte	C R T

This Group Object sends, on the bus (ABB i-bus® KNX), the currently valid control value that is acting on the valve.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ Option *Thermoelectric (PWM)*
- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Enable Group Object "Status Control value"* \ Option *Yes*

Status Control value	Output X – Valve:	DPT 1.002	1 bit	C R T

This Group Object sends, on the bus (ABB i-bus® KNX), the currently valid control value that is acting on the valve.

Telegram value:
- Dependent on the setting in the parameter *Valve drive operating principle, de-energized*

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ Option *Open/close signal*
- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *Enable Group Object "Status Control value"* \ Option *Yes*

Status byte	Output X – Valve:	DPT 22.102	1 byte	C R T

This Group Object sends the following status information on the bus (ABB i-bus® KNX):

- Bit 0: Valve position
  - 1 = > 0 %
  - 0 = 0 %
- Bit 1: Not used, permanent 0
- Bit 2: Overload
  - 1 = Active
  - 0 = Inactive
- Bit 3: Valve purge
  - 1 = Active
  - 0 = Inactive
- Bit 4: Function Service
  - 1 = Active
  - 0 = Inactive
- Bit 5: Manual operation
  - 1 = Active
  - 0 = Inactive
- Bit 6: Forced operation
  - 1 = Active
  - 0 = Inactive
- Bit 7: Control value override
  - 1 = Active
  - 0 = Inactive

Prerequisites for visibility
- Parameter window *Configuration*
  - Parameter *Output X Valve type* \ all options except *Deactivated*
  - Parameter *Output X Template* \ Option *No*
- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Enable Group Object "Status byte"* \ Option *Yes*
  Or
- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *Enable Group Object "Status byte"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Status Valve purge	**Output X – Valve:**	**DPT 1.002**	**1 bit**	**C R T**

This Group Object sends the valve purge status on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Valve purge active
- 0 = Valve purge inactive

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*
- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *Enable Group Object "Status Valve purge"* \ Option *Yes*
  Or
- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *Enable Group Object "Status Valve purge"* \ Option *Yes*

Activate valve purge	**Output X – Valve:**	**DPT 1.017**	**1 bit**	**C W**

This Group Object is used to activate the valve purge via the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Activate valve purge
- 0 = Activate valve purge

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*
- Parameter window *Output X:* \ Parameter window *Thermoelectric (PWM)* \ Parameter *valve purge* \ Option *Via Group Object* / *Automatic or via Group Object*
  Or
- Parameter window *Output X:* \ Parameter window *Open/close signal* \ Parameter *valve purge* \ Option *Via Group Object* / *Automatic or via Group Object*

Control value override	**Output X – Valve**	**DPT 5.001**	**1 byte**	**C W**

This Group Object is used to receive the setpoint for the manual valve override via the bus (ABB i-bus® KNX).

Note
The value in this Group Object becomes active only if the override has been activated by the Group Object *Activate/block control value override*.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*
- Parameter window *Output X:* \ Parameter window \ Parameter *Enable manual valve override* \ Option *Yes*

Activate/block control value override	**Output X – Valve**	**DPT 1.003**	**1 bit**	**C W**

This Group Object is used to activate or block manual valve override via the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Activate manual valve override
- 0 = Block manual valve override

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*
- Parameter window *Output X:* \ Parameter window \ Parameter *Enable manual valve override* \ Option *Yes*

Status Safety mode	**Output X – Valve:**	**DPT 1.005**	**1 bit**	**C R T**

This Group Object sends the status of Safety mode on the bus (ABB i-bus® KNX).
More information: → Safety mode, Page 29.

Telegram value:
- 1 = Alarm, Safety mode active
- 0 = No alarm, Safety mode not active

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ Option *External*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Cyclical monitoring of control value* \ Option *Yes*
  Or
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Cyclical monitoring of "Control value"* \ Option *Yes*

Forced operation 1 bit	**Output X – Valve**	**DPT 1.003**	**1 bit**	**C W**

This Group Object is used to activate or deactivate 1-bit forced operation via the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Forced operation 1 bit activated
- 0 = Forced operation 1 bit deactivated

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*
  Or
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Forced operation 1 bit"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Forced operation 2 bit	**Output X – Valve:**	**DPT 2.001**	**2 bit**	**C W**

This Group Object is used to activate or deactivate 2-bit forced operation via the bus (ABB i-bus® KNX).
Bit 1 activates or deactivates forced operation. Bit 0 is used to change over between the states *Forced operation active "ON"* and *Forced operation active "OFF"*.

Telegram value (bit 1 | bit 0):
- 0 | 0 = Forced operation inactive
- 0 | 1 = Forced operation inactive
- 1 | 0 = Forced operation active "OFF"
- 1 | 1 = Forced operation active "ON"

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*
  Or
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Forced operation 2 bit"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Status Fault valve	**Output X – Valve:**	**DPT 1.005**	**1 bit**	**C R T**

This Group Object sends the valve status (fault or no fault) on the bus (ABB i-bus® KNX).
An active fault message must be reset after correcting the fault, → Group Object *Reset valve output fault*.

Note
This Group Object reports a fault if at least one of the following errors occurs on the output pair (two adjacent outputs, e.g. A/B, C/D, etc.):
- Overload/short circuit lasting > 10 s
- No supply voltage (VC/S 6.2.2.2: terminal pair 1/2, VC/S 12.2.2.2: terminal pairs 1/2 and 15/16)

If the upper outputs (valve outputs A ... F) or the lower outputs (valve outputs G ... L) are not in use on the VC/S 12.2.2.2 and the corresponding supply voltage is not connected, the corresponding outputs should be deactivated (→ parameter *Output X Valve type*) to avoid unwanted fault messages.

Telegram value:
- Dependent on the setting in the following parameters:
  – *Value of Group Object "Status Fault valve"* (PWM)
  – *Value of Group Object "Status Fault valve"* (open/close)

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Output X Valve type* \ all options except *Deactivated*
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [thermoelectric (PWM)]* \ Parameter *Enable Group Object "Status Fault valve"* \ Option *Yes*
  Or
- Parameter window *Output X:* \ Parameter window *Monitoring and safety [open/close signal]* \ Parameter *Enable Group Object "Status Fault valve"* \ Option *Yes*

8.6 Group Objects RTC x – Room temperature controller

Function	Group Object name	Data point type	Length	Flags
Control value basic-stage heating	**RTC x – Room temperature controller:**	**DPT 5.001**	**1 byte**	**C T**

This Group Object sends the control value for basic-stage heating on the bus (ABB i-bus® KNX). The control value is also output on the selected output, → parameter *Output X Controller*.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*

Function	Group Object name	Data point type	Length	Flags
Control value additional-stage heating	**RTC x – Room temperature controller:**	**DPT 5.001**	**1 byte**	**C T**

This Group Object sends the control value for additional-stage heating on the bus (ABB i-bus® KNX).

Note
The control value for the additional stage is not linked within the device and cannot be output on an output.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  – Parameter *Basic-stage heating* \ all options except *Deactivated*
  – Parameter *Additional-stage heating* \ all options except *Deactivated*

Function	Group Object name	Data point type	Length	Flags
Control value basic-stage cooling	**RTC x – Room temperature controller:**	**DPT 5.001**	**1 byte**	**C T**

This Group Object sends the control value for basic-stage cooling on the bus (ABB i-bus® KNX). The control value is also output on the selected output, → parameter *Output X Controller*.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*

Function	Group Object name	Data point type	Length	Flags
Control value additional-stage cooling	**RTC x – Room temperature controller:**	**DPT 5.001**	**1 byte**	**C T**

This Group Object sends the control value for additional-stage cooling on the bus (ABB i-bus® KNX).

Note
The control value for the additional stage is not linked within the device and cannot be output on an output.

Telegram value:
- 0 … 100 %

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Additional-stage cooling* \ all options except *Deactivated*

External actual temperature	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive the measured temperature value from an external temperature sensor via the bus (ABB i-bus® KNX).

Telegram value:
- -30 … 120 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*

Fault: actual temperature	**RTC x – Room temperature controller:**	**DPT 1.005**	**1 bit**	**C R T**

This Group Object sends an alarm telegram on the bus (ABB i-bus® KNX) if no value is received on the Group Object *External actual temperature*.
More information: → Temperature acquisition monitoring, Page 29.

Telegram value:
- 1 = Alarm
- 0 = No alarm

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Temperature acquisition* \ Parameter *Enable monitoring* \ Option *Yes*

Current setpoint (main/secondary)	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C R T**

This Group Object sends the currently valid setpoint on the bus (ABB i-bus® KNX).

Telegram value:
- 0 … 120 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*

Request setpoint adjustment (main/secondary)	**RTC x – Room temperature controller:**	**DPT 9.002**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint adjustment via the bus (ABB i-bus® KNX).

Telegram value:
- 0 … 120 K

Note
The theoretically possible telegram value is limited by the settings in the following parameters:
- *Maximum heating increase*
- *Maximum cooling increase*
- *Maximum heating decrease*
- *Maximum cooling decrease*

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*

Confirm setpoint adjustment (main/secondary)	**RTC x – Room temperature controller:**	**DPT 9.002**	**2 bytes**	**C R T**

This Group Object sends the confirmation of the setpoint adjustment on the bus (ABB i-bus® KNX) as was requested via Group Object *Request setpoint adjustment (main/secondary)*.

Telegram value:
- 0 … 120 K

Note
The theoretically possible telegram value is limited by the settings in the following parameters:
- *Maximum heating increase*
- *Maximum cooling increase*
- *Maximum heating decrease*
- *Maximum cooling decrease*

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Confirm setpoint adjustment (main/secondary)"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Reset setpoint adjustment	**RTC x – Room temperature controller:**	**DPT 1.015**	**1 bit**	**C W U**

This Group Object is used to reset a setpoint adjustment via the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Reset setpoint adjustment
- 0 = No function

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
    Or
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint adjustment* \ Parameter *Reset manual setpoint adjustment via Group Object* \ Option *Yes*

Heating setpoints	**RTC x – Room temperature controller:**	**DPT 275.100**	**8 bytes**	**C W T U**

This Group Object sends the setpoints for all *Heating* operating modes on the bus (ABB i-bus® KNX) as four KNX float values.

Telegram value:
- 4 KNX float values

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Heating setpoints" (DPT 275.100)* \ Option *Yes*

Cooling setpoints	**RTC x – Room temperature controller:**	**DPT 275.100**	**8 bytes**	**C W T U**

This Group Object sends the setpoints for all *Cooling* operating modes on the bus (ABB i-bus® KNX) as four KNX float values.

Telegram value:
- 4 KNX float values

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Cooling setpoints" (DPT 275.100)* \ Option *Yes*

Base setpoint Comfort heating	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a change in the base setpoint.
The base setpoint is defined in the parameter *Base setpoint is*, and it can be overridden by a temperature value received on this Group Object.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Comfort heating setpoint = Comfort cooling setpoint* \ Option *Yes*
    Or
  - Parameter *Base setpoint is* \ Option *Comfort heating setpoint*

Comfort heating setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Comfort heating* via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Comfort heating setpoint*. The overridden setpoint is limited to the valid value range (10 … 40 °C).
Manual setpoint adjustment acts on the overridden setpoint.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*
  - Parameter *Base setpoint is* \ Option *Comfort cooling setpoint*

Standby heating setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Standby heating* via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Standby heating setpoint*. The overridden setpoint is limited to the valid value range (10…40 °C) and limited by the value *Comfort heating*.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Economy heating setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Economy heating* via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Economy heating setpoint*. The overridden setpoint is limited to the valid value range (10…40 °C) and limited by the value *Comfort heating*.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Building Protection heating setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Building Protection heating* (frost protection) via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Building Protection heating setpoint*. The overridden setpoint is limited to the valid value range (5…15 °C) and limited by the value *Comfort heating*.

Telegram value:
- 5 … 15 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Base setpoint Comfort cooling	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), a change in the base setpoint.
The base setpoint is defined in the parameter *Base setpoint is*, and it can be overridden by a temperature value received on this Group Object.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Comfort heating setpoint = Comfort cooling setpoint* \ Option *No*
  - Parameter *Base setpoint is* \ Option *Comfort cooling setpoint*

Function	Group Object name	Data point type	Length	Flags
Comfort cooling setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Comfort cooling* via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Comfort cooling setpoint*. The overridden setpoint is limited to the valid value range (10 … 40 °C).
Manual setpoint adjustment acts on the overridden setpoint.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage heating* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*
  - Parameter *Base setpoint is* \ Option *Comfort heating setpoint*

Function	Group Object name	Data point type	Length	Flags
Standby cooling setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Standby cooling* via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Standby cooling setpoint*. The overridden setpoint is limited to the valid value range (10…40 °C) and limited by the value *Comfort cooling*.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Economy cooling setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Economy cooling* via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Economy cooling setpoint*. The overridden setpoint is limited to the valid value range (10…40 °C) and limited by the value *Comfort cooling*.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Building Protection cooling setpoint	**RTC x – Room temperature controller:**	**DPT 9.001**	**2 bytes**	**C W T U**

This Group Object is used to receive a setpoint change for the operating mode *Building Protection cooling* (heat protection) via the bus (ABB i-bus® KNX).
This Group Object overrides the value set in the parameter *Building Protection cooling setpoint*. The overridden setpoint is limited to the valid value range (27 … 45 °C) and limited by the value *Comfort cooling*.

Telegram value:
- 10 … 40 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Setpoint manager*
  - Parameter *Deactivate Standby and Economy* \ Option *No*
  - Parameter *Activate absolute setpoints* \ Option *Yes*
  - Parameter *Enable Setpoint Group Objects* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Operating mode (main/secondary)	**RTC x – Room temperature controller**	**DPT 20.102**	**1 byte**	**C W T U**

This Group Object is used to receive, via the bus (ABB i-bus® KNX), the operating mode to be set.

Telegram value:
- 1 = Comfort
- 2 = Standby
- 3 = Economy
- 4 = Building Protection

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*

Function	Group Object name	Data point type	Length	Flags
Operating mode override	**RTC x – Room temperature controller:**	**DPT 20.102**	**1 byte**	**C W T U**

This Group Object is used to receive the override of the current operating mode via the bus (ABB i-bus® KNX).

Note
When the operating modes Standby and Economy are deactivated (→ parameter *Deactivate Standby and Economy*), the telegram values "2" and "3" are ignored.

Telegram value:
- 0 = Automatic
- 1 = Comfort
- 2 = Standby
- 3 = Economy
- 4 = Building Protection

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
    Or
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Operating mode override"* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Setpoint manager* \ Parameter *Deactivate Standby and Economy* \ Option *No*

Function	Group Object name	Data point type	Length	Flags
Current operating mode (main/secondary)	**RTC x – Room temperature controller:**	**DPT 20.102**	**1 byte**	**C R T**

This Group Object sends the currently valid operating mode on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Comfort
- 2 = Standby
- 3 = Economy
- 4 = Building Protection

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Window contact	**RTC x – Room temperature controller:**	**DPT 1.019**	**1 bit**	**C W T U**

This Group Object is used to receive the window status via the bus (ABB i-bus® KNX).

Note
When the window is open (1), the operating mode *Building Protection* is activated after the delay time (→ parameter *Delay time*). When the window is closed (0), the operating mode specified by the Group Object *Operating mode (main/secondary)* applies.

Telegram value:
- 1 = Window open
- 0 = Window closed

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable function: Window contact* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Presence detector	**RTC x – Room temperature controller:**	**DPT 1.018**	**1 bit**	**C W T U**

This Group Object is used to receive the presence status (person in the room) via the bus (ABB i-bus® KNX).

Note
When the room is occupied (1), the operating mode *Comfort* is activated. When the room is unoccupied (0), the operating mode specified by the Group Object *Operating mode (main/secondary)* applies.

Telegram value:
- 1 = Room occupied/person in room
- 0 = Room unoccupied/no person in room

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable function: Presence detector* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Dew point alarm	**RTC x – Room temperature controller:**	**DPT 1.019**	**1 bit**	**C W T U**

This Group Object is used to receive the dew point status via the bus (ABB i-bus® KNX).

Note
The reaction to an active dew point alarm (1) is defined in the parameter *When dew point alarm is active*.

Telegram value:
- 1 = Dew point alarm active
- 0 = Dew point alarm inactive

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Enable Group Object "Dew point alarm"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Status RTSM Controller (main/secondary)	**RTC x – Room temperature controller:**	**DPT 21.107**	**1 byte**	**C R T**

This Group Object sends the status information of the controller as an 8-bit value on the bus (ABB i-bus® KNX).

Telegram value:
- Bit 0: Window status
  - 1 = Window open
  - 0 = Window closed
- Bit 1: Presence detector
  - 1 = Room occupied/person in room
  - 0 = Room unoccupied/no person in room
- Bit 2: Presence button
  - 1 = Room occupied/person in room
  - 0 = Room unoccupied/no person in room
- Bit 3: Comfort extension
  - 1 = Active
  - 0 = Inactive
- Bit 4: Override
  - 1 = Active
  - 0 = Inactive
- Bit 4 … 7: Not used, permanent 0

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Status RTSM controller (main/secondary)" (DPT 21.107)* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Status RTC Controller (main/secondary)	**RTC x – Room temperature controller:**	**DPT 22.103**	**16 bit**	**C R T**

This Group Object sends the status information of the controller as a 16-bit value on the bus (ABB i-bus® KNX).

Telegram value:

- Bit 0: Error
  - 1 = Error
  - 0 = No error
- Bit 1: Type of operation
  - 1 = Heating
  - 0 = Cooling
- Bit 2: Dew point alarm
  - 1 = Active (controller blocked)
  - 0 = Inactive (controller enabled)
- Bit 3: Frost alarm
  - 1 = Active
  - 0 = Inactive
- Bit 4: Overheating alarm
  - 1 = Active
  - 0 = Inactive
- Bit 5: Status Controller
  - 1 = Inactive (control value = 0)
  - 0 = Active
- Bit 6: Additional-stage heating or additional-stage cooling
  - 1 = Active
  - 0 = Inactive
- Bit 7: Status Type of operation Heating
  - 1 = Enabled
  - 0 = Blocked
- Bit 8: Status Type of operation Cooling
  - 1 = Enabled
  - 0 = Blocked
- Bit 9 … 15: Not used, permanent 0

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Status RTC (main/secondary)" (DPT 22.103)* \ Option *Yes*

Status RHCC	RTC x – Room temperature controller:	DPT 22.101	16 bit	C R T

This Group Object sends the status information of the controller as a 16-bit value on the bus (ABB i-bus® KNX).

Telegram value:

- Bit 0: Error status
  - 1 = Error
  - 0 = No error
- Bit 1: Heating controller
  - 1 = Inactive
  - 0 = Active
- Bit 2 … 6: Not used, permanent 0
- Bit 7: Heating deactivated
  - 1 = Deactivated
  - 0 = Activated
- Bit 8: Type of operation
  - 1 = Heating
  - 0 = Cooling
- Bit 9: Cooling controller
  - 1 = Inactive
  - 0 = Active
- Bit 10: Not used, permanent 0
- Bit 11: Cooling deactivated
  - 1 = Deactivated
  - 0 = Activated
- Bit 12: Dew point alarm
  - 1 = Active
  - 0 = Inactive
- Bit 13: Frost alarm
  - 1 = Active
  - 0 = Inactive
- Bit 14: Overheating alarm
  - 1 = Active
  - 0 = Inactive
- Bit 15: Not used, permanent 0

Prerequisites for visibility

- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings* \ Parameter *Enable Group Object "Status RHCC" (DPT 22.101)* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Heating active	**RTC x – Room temperature controller:**	**DPT 1.011**	**1 bit**	**C T**

This Group Object sends the status of the type of operation *Heating* on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Heating active, control value > 0
- 0 = Heating inactive, control value = 0

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Enable Group Object "Heating active" (control value > 0)* \ Option *Yes*

Cooling active	**RTC x – Room temperature controller:**	**DPT 1.011**	**1 bit**	**C T**

This Group Object sends the status of the type of operation *Cooling* on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Cooling active, control value > 0
- 0 = Cooling inactive, control value = 0

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Enable Group Object "Cooling active" (control value > 0)* \ Option *Yes*

Heating/cooling changeover	**RTC x – Room temperature controller:**	**DPT 1.100**	**1 bit**	**C W U**

This Group Object is used to receive the change of the type of operation (*Heating/ Cooling*) via the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Heating
- 0 = Cooling

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Heating/cooling changeover* \ Option *Via Group Object*

Status Heating/cooling changeover	**RTC x – Room temperature controller:**	**DPT 1.100**	**1 bit**	**C T**

This Group Object sends the status of the active type of operation (*Heating* or *Cooling*) on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Heating active
- 0 = Cooling active

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*

Block/enable heating	**RTC x – Room temperature controller:**	**DPT 1.003**	**1 bit**	**C W T U**

This Group Object is used to block or enable the type of operation *Heating* via the bus (ABB i-bus® KNX).

Note
The type of operation *Heating* is always enabled, and can be blocked with the telegram value "0". If the type of operation *Heating* is blocked via this Group Object, the heating stages cannot be activated and the control value 0 % is sent to the outputs or on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Enable heating
- 0 = Block heating

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Enable Group Object "Block/enable heating"* \ Option *Yes*

Block/enable cooling	**RTC x – Room temperature controller:**	**DPT 1.003**	**1 bit**	**C W T U**

This Group Object is used to block or enable the type of operation *Cooling* via the bus (ABB i-bus® KNX).

Note
The type of operation *Cooling* is always enabled, and can be blocked with the telegram value "0". If the type of operation *Cooling* is blocked via this Group Object, the cooling stages cannot be activated and the control value 0 % is sent to the outputs or on the bus (ABB i-bus® KNX).

Telegram value:
- 1 = Enable cooling
- 0 = Block cooling

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Enable Group Object "Block/enable cooling"* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags
Block/enable heating stages separately	**RTC x – Room temperature controller:**	**DPT 20.116**	**1 byte**	**C W T U**

This Group Object is used to block or enable the heating stages via the bus (ABB i-bus® KNX).

Telegram value:
- 0 = Block all heating stages, control values are set to 0
- 1 = Enable basic-stage heating, only the basic-stage control value is sent, and is output on the output
- 2 = Enable additional-stage heating, only the additional-stage control value is sent
- 3 = Enable all heating stages, all control values are sent, and the output outputs the control value for the basic stage

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Enable Group Object "Block/enable heating stages separately" (DPT 21.106)* \ Option *Yes*

Block/enable cooling stages separately	RTC x – Room temperature controller:	DPT 20.116	1 byte	C W T U

This Group Object is used to block or enable the cooling stages via the bus (ABB i-bus® KNX).

Telegram value:
- 0 = Block all cooling stages, control values are set to 0
- 1 = Enable basic-stage cooling, only the basic-stage control value is sent, and is output on the output
- 2 = Enable additional-stage cooling, only the additional-stage control value is sent
- 3 = Enable all cooling stages, all control values are sent, and the output outputs the control value for the basic stage

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Enable Group Object "Block/enable cooling stages separately" (DPT 21.106)* \ Option *Yes*

Controller synchronization of output	RTC x – Room temperature controller:	DPT 20.105	1 byte	C R T

This Group Object sends the HVAC control mode to the slave via the bus (ABB i-bus® KNX).

Telegram value:
- 0 = Automatic Heating/cooling changeover
- 1 = Heating
- 3 = Cooling

Note
Other telegram values are ignored.

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Enable function: Controller synchronization (DPT 20.105)* \ Option *Yes*
  - Parameter *Synchronization settings* \ Option *Master*

Controller synchronization of input	RTC x – Room temperature controller:	DPT 20.105	1 byte	C W U

This Group Object is used to receive the HVAC control mode from the master via the bus (ABB i-bus® KNX).

Telegram value:
- 0 = Automatic Heating/cooling changeover
- 1 = Heating
- 3 = Cooling

Note
Other telegram values are ignored.

Prerequisites for visibility
- Parameter window *Configuration*Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Enable function: Controller synchronization (DPT 20.105)* \ Option *Yes*
  - Parameter *Synchronization settings* \ Option *Slave*

Basic-stage heating limit temperature	RTC x – Room temperature controller	DPT 9.001	2 bytes	C W U

This Group Object is used to receive the limit temperature for basic-stage *Heating* via the bus (ABB i-bus® KNX).

Telegram value:
- 0 … 120 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Controller settings* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window → Controller settings, Page 123
  - Parameter *Extended settings* \ Option *Yes*
  - Parameter *Temperature limitation* \ Option *Yes*

Function	Group Object name	Data point type	Length	Flags		
Additional-stage heating limit temperature	**RTC x – Room temperature controller**	**DPT 9.001**	**2 bytes**	**C**	**W**	**U**

This Group Object is used to receive the limit temperature for additional-stage *Heating* via the bus (ABB i-bus® KNX).

Telegram value:
- 0 … 120 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
  - Parameter *Additional-stage heating* \ all options except *Deactivated*
  - Parameter *Controller settings* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Controller settings*
  - Parameter *Extended settings* \ Option *Yes*
  - Parameter *Temperature limitation* \ Option *Yes*

Basic-stage cooling limit temperature	RTC x – Room temperature controller	DPT 9.001	2 bytes	C	W	U

This Group Object is used to receive the limit temperature for basic-stage *Cooling* via the bus (ABB i-bus® KNX).

Telegram value:
- 0 … 120 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Controller settings* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Controller settings*
  - Parameter *Extended settings* \ Option *Yes*
  - Parameter *Temperature limitation* \ Option *Yes*

Additional-stage cooling limit temperature	RTC x – Room temperature controller	DPT 9.001	2 bytes	C	W	U

This Group Object is used to receive the limit temperature for additional-stage *Cooling* via the bus (ABB i-bus® KNX).

Telegram value:
- 0 … 120 °C

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Additional-stage cooling* \ all options except *Deactivated*
  - Parameter *Controller settings* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Controller settings*
  - Parameter *Extended settings* \ Option *Yes*
  - Parameter *Temperature limitation* \ Option *Yes*

Basic load	RTC x – Room temperature controller	DPT 1.011	1 bit	C	W	U

This Group Object is used to receive the activation of the basic load via the bus (ABB i-bus® KNX).
The basic load is specified in the parameter *Minimum control value [Basic load]* and can be set individually for each heating and cooling stage.
The basic load is always activated jointly for all stages, but it is applicable only to the active type of operation (*Heating* or *Cooling*). The basic load remains active during the change in type of operation.

Telegram value:
- 1 = Activate basic load
- 0 = Deactivate basic load

Prerequisites for visibility
- Parameter window *Configuration* \ Parameter *Enable RTC x* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Basic settings*
  - Parameter *Basic-stage heating* \ all options except *Deactivated*
    Or
  - Parameter *Basic-stage cooling* \ all options except *Deactivated*
  - Parameter *Controller settings* \ Option *Yes*
- Parameter window *RTC x* \ Parameter window *Controller settings*
  - Parameter *Extended settings* \ Option *Yes*
  - Parameter *Basic load* \ Option *Via Group Object*

9 Operation

9.1 Manual operation

Note
The KNX power supply must be established to operate the device in *Manual operation* mode. Manual operation is deactivated in the event of KNX voltage failure.

Manual operation mode permits on-site operation of the device using a membrane keypad.

Note
The safety functions as well as operation using the i-bus® Tool have a higher priority than the *Manual operation* mode. An output cannot be operated using the membrane keypad if it is blocked by a safety function or can be operated using the i-bus® Tool. If the safety function is canceled in *Manual operation* operating mode, the output reacts corresponding to its parameterization.

Note
When Manual operation is active, the outputs will continue to react to KNX commands until they are switched via Manual operation. If an output has been switched via Manual operation, incoming KNX commands will be processed in the background and no longer executed until Manual operation is de-activated.
More information: → State after deactivating manual operation, Page 62.

The device is in *KNX operation* after connection to the bus (ABB i-bus® KNX), KNX voltage recovery, ETS download or ETS reset. The *Manual operation* LED is off.

9.1.1 Central operation via membrane keypad

When the device is in manual mode, all outputs can be operated together using the membrane keypad. If all outputs are selected, they can be opened or closed together.

Open all outputs
1. Press and hold the *S button* (> 5 seconds).
   ⇨ *S* LED lights up for 2 s, then flashes for 3 s. Outputs selected. Zone LEDs light up.
2. Briefly press the *Valve output I* button.

⇨ The LED of the first output zone lights up. All outputs are open. Device remains in manual mode.

Close all outputs
1. Press and hold the *S button* (> 5 seconds).
   ⇨ *S* LED lights up for 2 s, then flashes for 3 s. Outputs selected. Zone LEDs light up.
2. Briefly press the *Valve output II* button.

⇨ The LED of the first output zone lights up. All outputs are closed. Device remains in manual mode.

Note
The safety functions as well as operation using the i-bus® Tool have a higher priority than the *Manual operation* mode. An output cannot be operated using the membrane keypad if it is blocked by a safety function or can be operated using the i-bus® Tool. If the safety function is canceled in *Manual operation* operating mode, the output reacts corresponding to its parameterization.

9.1.2 Activating manual operation

- Press and hold *S*-button for 2 … 5 seconds.

⇨ *Manual operation* LED lights up. Manual operation is active.

Note
If Manual operation is not enabled (→ parameter *Enable manual operation*) or is blocked via the Group Object *Enable/Block Manual operation*, then *Manual operation* mode cannot be activated.

9.1.3 Blocking manual operation

Manual operation mode can be blocked via the Group Object *Enable/Block Manual operation*. If *Manual operation* mode was blocked via this Group Object, the only way to reactivate it is via this Group Object.

9.1.4 Deactivate manual operation

Manual operation mode can be deactivated in various ways:
- Press and hold *S*-button for 2 … 5 seconds.
- Automatically after a time set in the parameter *Automatic reset from manual operation to KNX operation*.
- By a download. After the download is complete, the devices are in KNX operation.
- Via the Group Object *Deactivate manual operation*.

10 Maintenance and cleaning

10.1 Service

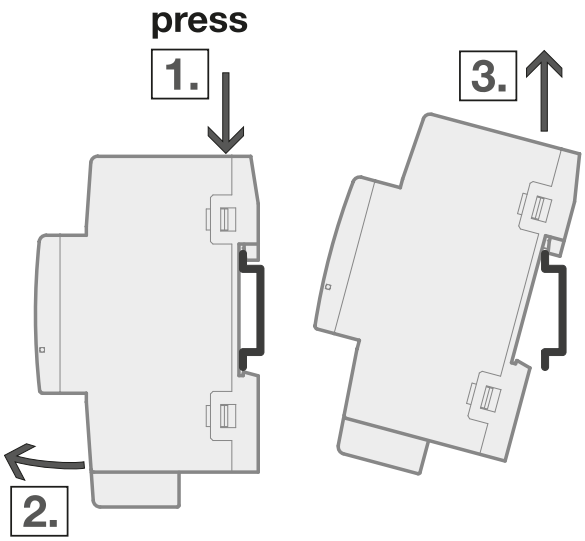
The devices are maintenance-free if used properly. In the event of damage, e.g. during transport and/or storage, repairs are not allowed to be carried out.

10.2 Cleaning

1. Disconnect devices from the electrical power supply before cleaning.
2. Clean dirty devices using a dry cloth or a slightly damp cloth.

11 Removal and disposal

11.1 Removal



2CDC072012F0015

Fig. 33: Removing from the mounting rail

1. Press on the top of the device.
2. Release the bottom of the device from the mounting rail.
3. Lift the device up and off the mounting rail.

11.2 Environment

Consider environmental protection.

Electrical and electronic devices must not be disposed of as domestic waste.

The device contains valuable resources that can be recycled. Therefore, please take the device to a suitable recycling center. All packaging materials and devices are provided with markings and test seals for proper disposal. Always dispose of packaging material and electrical devices or their components at collection points or disposal companies authorized for this purpose. The products comply with the statutory requirements, particularly the law on electrical and electronic equipment and the REACH regulation. (EU directive 2012/19/EU WEEE and 2011/65/EU RoHS) (EU REACH regulation and the law implementing the regulation (EC) no.1907/2006)

12 Planning and application

12.1 Priorities

Note
The priorities are in descending order from highest to lowest.

12.1.1 Device priorities

a) Service
b) Forced operation 2 bit
c) Forced operation 1 bit
d) i-bus® Tool
e) Type of operation *Manual operation*
f) Manual valve override
g) Valve purge
h) Safety mode
i) Type of operation *KNX operation*
j) KNX voltage recovery or ETS download

Note
The interface to i-bus® Tool is available from the following software versions:
- Application V1.1 or later

12.2 Basic knowledge

12.2.1 Hysteresis

The hysteresis indicates the difference by which a value must change before there is a reaction to the value change (e.g. switch on the heating if temperature drops below setpoint). Hysteresis prevents frequent switching in response to minimal changes. A differentiation is made between single-sided and double-sided hysteresis.

Example
Single-sided hysteresis:
- Setpoint temperature = 22 °C
- Hysteresis = 4 K

The heating is switched on if an actual temperature drops below 18 °C and switched off if an actual temperature exceeds 22 °C.

Double-sided hysteresis:
- Setpoint temperature = 22 °C
- Hysteresis = 4 K

The heating is switched on if an actual temperature drops below 20 °C and switched off if an actual temperature exceeds 24 °C.

12.2.2 Manual valve override

During manual valve override, the active control value is overridden. The active control value is the control value calculated by the internal controller or sent via the bus (ABB i-bus® KNX) by an external controller.

If manual valve override is enabled (→ parameter *Enable manual valve override*), the active valve control value is overridden with the value of Group Object *Control value override*.

When manual valve override is blocked by the Group Object *Activate/block control value override*, the active control value cannot be overridden.

If a valve override is active, it remains active even after KNX voltage recovery or ETS download.

Possible applications:
- System function test
- Specific override of the active control value

12.2.3 valve purge

To prevent the valve from sticking during an extended idle period, the valve is completely opened and closed one time during the valve purge.

The valve purge can be triggered via a Group Object, or after a specified purge cycle. A KNX voltage failure extends the purge cycle by up to one day plus the duration of the KNX voltage failure. If the valve purge is interrupted by a KNX voltage failure, it does not resume automatically after KNX voltage recovery.

When automatic valve purge is activated, the following events reset the purge cycle:
- Valve purge performed (automatically or via Group Object)
- ETS download
- Settings for the following parameters:
  - *Reset purge cycle* (open/close signal)
  - *Reset purge cycle* (PWM) and *Reset purge cycle from control value greater than* (PWM)

12.2.4 Pump seizure protection

To prevent the pump from seizing up during an extended idle period, the pump is switched on for a parameterizable duration when pump seizure protection is active. The pump switches on automatically after the specified cycle. A KNX voltage failure extends the cycle by up to one day plus the duration of the KNX voltage failure.

When pump seizure protection is activated, the following events reset the cycle:
- Activation of the pump (in standard operation or through pump seizure protection)
- ETS download

12.2.5 Cascading

In engineering and electronics, cascading refers to connecting or chaining several modules or assemblies in series. This method allows the functions of one device to be integrated into another. This means that the results and signals from several devices can be used as part of a single calculation or control.

The connection between the individual modules is unidirectional, which means that the signals flow in a specified direction. Cascading reduces the need to send the results of each individual device to the system separately. Instead, the data can be processed centrally, which makes the system clearer and more efficient.

Cascading is not monitored. If a device in the cascade fails, malfunctions may occur. Cascading should only be used in non-critical systems.

Example

Control/regulation without cascading

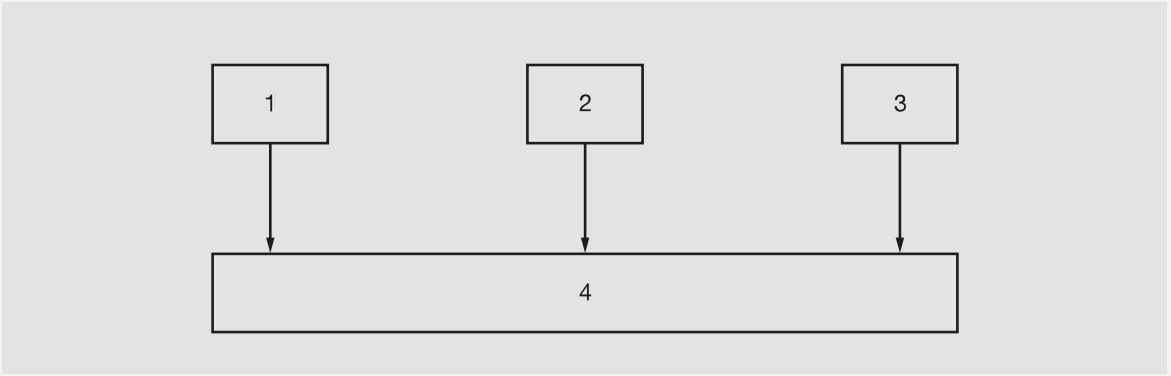


Fig. 34: Control/regulation without cascading

Each device (1, 2, 3) sends telegrams directly to the receiver (4).

Example

Control/regulation with cascading

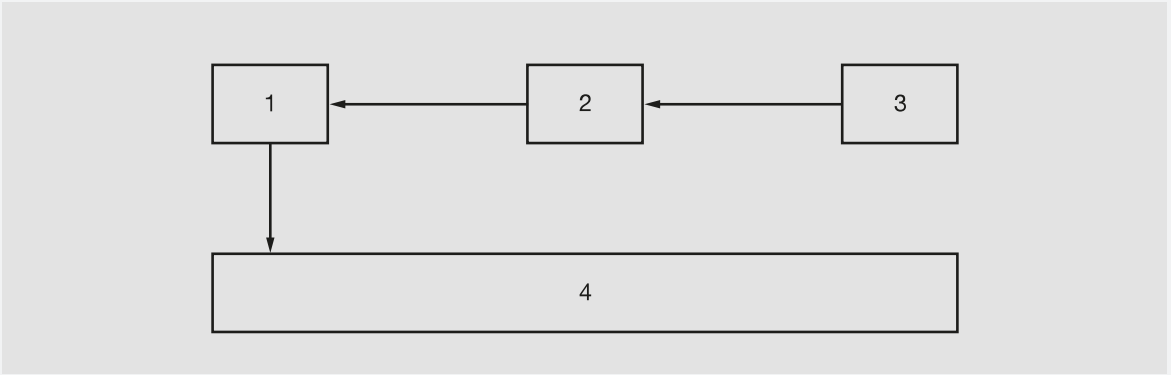


Fig. 35: Control/regulation with cascading

The devices (1, 2, 3) are connected in series (cascaded) and divided into master and slave units. Only the master (1) sends telegrams to the receiver (4). The slaves (2, 3) only communicate with the master.

12.2.6 KNX Secure

Note

ETS version 6 or later is required when using KNX Secure. The use of the latest ETS version is recommended. Using older ETS versions can cause errors during project planning, problems during commissioning (e.g. while replacing devices), or while undertaking diagnostics on group addresses and devices.

KNX Secure is an encryption technology that guarantees data protection in a KNX twisted pair network. KNX Secure is based on end-to-end encryption that ensures all data exchanged between KNX devices are encrypted and can only be read by authorized users.

In conventional KNX networks (colloquially termed "KNX Plain"), data are sent unencrypted on the bus. The data could be read by anyone with access to the bus and can be intercepted or falsified by unauthorized individuals.

Using KNX Secure protects transmitted data against unauthorized access, ensures data integrity and minimizes potential security risks. KNX Secure helps to increase security and privacy in KNX-based smart home or building automation systems. KNX devices that only support KNX Plain can be used in the same installation with the aid of a suitable coupler.

Fig. 36: KNX Secure logo

A KNX Secure product is identifiable by the KNX Secure logo on the packaging or the product. This logo indicates that the product meets the KNX Secure security standard and supports KNX Secure encryption technology.

KNX Secure distinguishes between two types of encrypted KNX telegrams:

- KNX IP Secure can only be used on the KNX IP medium (typically the backbone line) exposed to an external IP network (e.g. the Internet). KNX IP Secure telegrams are fully encrypted.
- KNX Data Secure can be used on any KNX medium, but is only permitted to be used for the part of the KNX installation not exposed to an external IP network (e.g. the Internet). KNX Data Secure telegrams are encrypted.

For more information, see:
→ ABB documentation "KNX Secure"
→ Documentation available at knx.org

12.2.7 Network (cyber) security

The industry is increasingly faced with cyber security risks. To increase the stability, security and robustness of its solutions, ABB has introduced cyber security robustness tests as part of the product development process.

In addition, the sections below include guidelines and mechanisms that you can use to improve the security of KNX systems.

12.2.7.1 Preventing unauthorized access

The basis for any protection concept is the careful shielding of the system against unauthorized access. The following points must be taken into consideration when planning and installing a KNX system:

- Only authorized persons (installers, custodians, users) should be allowed to have physical access to the KNX system.
- Sub-distributions with KNX devices should be closed, or in rooms to which only authorized persons have access.
- If available, use the anti-theft features on the KNX devices.
- All components in a KNX system should be permanently installed and protected from unauthorized access.
- The bus cable (ABB i-bus® KNX) should not be visible inside or outside the building. Cables outdoors are an increased risk. Physical access should be made particularly difficult here.
- Devices installed in areas with limited protection (e.g. outdoor areas, underground parking lots, restrooms, etc.) should be designed using a line coupler as a separate line.
- If possible, KNX DATA Secure should be used for data transmission in KNX networks (→ KNX Secure, Page 172).
- The system should be divided into security segments that are based on the available security functions of the devices used. This is done by using segment couplers.

12.2.7.2 IP cabling inside the building

For building automation, use a separate LAN or WiFi network with its own hardware (routers, switches, etc.). Regardless of the KNX system, apply the usual security mechanisms for IP networks:

- MAC filter
- Encryption of wireless networks
- Usage of strong passwords, and password protection against access by unauthorized persons

12.2.7.3 Using filter tables

Filter tables are used to improve system security and ensure that only authorized telegrams are forwarded.

12.2.8 Refreshed KNX state

If an input or an output is blocked by device-specific functions (e.g. manual operation, alarms, block, forced operation, switching delay), it will not react to telegrams received via the bus (ABB i-bus® KNX) while the block is active.

While a block is active, the device processes the telegrams received in the background. Active functions (e.g. staircase lighting, logic, position, brightness value) are executed in the background, but the results are not sent. The actual value is sent to the input or output when the block is canceled.

If the input or output has not received any telegrams via the bus (ABB i-bus® KNX) while a block is active, the input or output will assume the state it was in before the block.

12.2.9 Sending or switching delay

No telegrams are sent on the bus during the sending or switching delay (ABB i-bus® KNX).

Telegrams received (e.g. requests from a visualization system) are sent to the outputs after the sending or switching delay expires. The state of the outputs is set according to the settings in the ETS application or the telegram values of the Group Objects.

Time sequences (e.g. staircase lighting time) are started immediately during the sending or switching delay. If, at the time of reception, the staircase lighting time is shorter than the remaining sending or switching delay, the staircase lighting time elapses during the sending or switching delay. After the sending or switching delay has elapsed, there is no switching command; the staircase lighting is not switched on.

Note
The sending or switching delay includes the device initialization time.

12.2.10 Telegram rate limit

The bus load generated by the device can be limited using the telegram rate limit. This limit relates to all telegrams sent by the device.

The device counts the number of telegrams sent within the parameterized period. As soon as the maximum number of sent telegrams is reached, no further telegrams are sent on the bus (ABB i-bus® KNX) until the end of the period. A new period commences automatically at the end of the previous period. The telegram counter is reset to zero. Telegrams can be sent again. The Group Object always sends the current telegram value.

The first period (break time) is not precisely predefined. The break time can be anywhere between 0 seconds and the parameterized period. The subsequent periods correspond to the parameterization.

Example
- Number of telegrams = 20
- Maximum number of telegrams per period = 5
- Period = 5 s

The device immediately sends 5 telegrams. The next 5 telegrams are sent after a maximum of 5 seconds. From this point, a further 5 telegrams are sent via the bus (ABB i-bus® KNX) every 5 seconds.

12.2.11 Cyclical monitoring

The reception of a telegram on a Group Object can be monitored using cyclical monitoring. If a telegram is not received on the Group Object within a parameterizable time (monitoring time), the sending device may be faulty or the bus cable to the sending device may be interrupted.

Depending on the device, there are two possible reactions to a missing telegram: either the reaction set in the application-specific parameters; or, on devices without those parameters, the corresponding predefined alarm is triggered.

After receiving a telegram, ETS download, or KNX voltage recovery, the monitoring time is restarted.

Note
The monitoring time should be at least quadruple the cyclical sending time of the sending device. As a result, the set reactions or the alarms will not be triggered immediately if a telegram is missing, e.g. due to high bus load.

13 Appendix

ABB AG – STOTZ-KONTAKT
Eppelheimer Str. 82
DE-69123 Heidelberg
go.abb/contact
Telephone: +49 (0)6221 701 607
Email: knx.marketing@de.abb.com

Additional information and regional points of contact:
www.abb.de/knx
www.abb.com/knx

Publication number 9AKK108464A0992 Rev. A | 18.12.2025 | EN-US