KNX Mix Actuator

Reference Manual

KNX Secure Mix Actuator MX324-16A-S	WRKT4824J
KNX Secure Mix Actuator MX320-16A-S	WRKT4820J
KNX Secure Mix Actuator MX316-16A-S	WRKT4816J
KNX Secure Mix Actuator MX312-16A-S	WRKT4812J
KNX Secure Mix Actuator MX308-16A-S	WRKT4808J

V 1.0

Panasonic

1 Contents

2 List of figures

3 List of tables

4 Document overview

4.1 Document updates

Version	Date	Modifications
1.0	March 2024	Preparation of the final version

Table 1

4.2 List of abbreviations

Abbreviation	Description
KNX Communication Flags	
C	Communication
R	Read
W	Write
T	Transmit
U	Update
Other	
Par.	Parameter
Obj.	Object
LC	Last command
TC	Transmitted command

Table 2

5 Product description

5.1 General information

Panasonic KNX Next Generation Mix Actuator is output device designed to fulfill the automation requirements of residential and commercial KNX projects. It comes with KNX Data Secure feature which assures secure data communication on twisted pair bus line for the compatible devices. Panasonic KNX Next Generation Mix Actuator has competent control features such as Lighting, Heating, Shutter and Fan Coil. The current status of output signals can be visualized on the device with its status LEDs. Signals can be manually overridden using the device's manual control button. Many complex configurations and various scenarios can be realized, thanks to the powerful auxiliary functions (i.e. logic, converter and sequence functions, etc.).

5.2 Main features

- KNX Data Security
- Manual Control
- Switching (Lighting / Heating)
- Shutter / Blind Control (AC / DC)
- Fan Coil (2 pipe / 4 pipe)
- Auxiliary Functions (Sequencer, Counter, Scene Actuator, Filter, Converter, Logic Gate, Presence Detector Controller, Staircase Controller)

5.3 KNX Data Secure

KNX Secure, protects KNX devices against unwanted access and manipulation. A device with the KNX Secure feature transmits telegrams sent to the KNX bus line in a secure and encrypted manner.

Figure 1 KNX Secure Device Certificate Label

Two removable labels with a device certificate come with the product. One of the labels is on the product and the other is inside the box. Installation of the product can be done via ETS using the device certificate.

⚠ Keep the removable labels with the device certificate in a safe place.

5.4 Factory Default

In the delivery state, manual control (see below) is enabled with some restrictions to test the wiring. Only one channel can be set at a certain time to protect the connected equipment (e.g., Shutter/Blind). KNX

socket must be connected to power up the device. Non restricted manual control is available after ETS programming.

5.5 ETS Programming

User can define the device behavior and connect it to another KNX devices after programming it by ETS. The behavior of the device after programming with the ETS depends on the configuration. The description of the features, parameters and objects is in the device reference manual.

5.6 Factory Reset

The factory reset restores the device to its basic settings, including the physical address (15.15.255), and disables the KNX Data Security. During secure operation, a commissioned KNX secure device can be recommissioned with the device certificate after a factory reset.

Performing a factory reset;

- Disconnect the KNX bus connector from the device.
- Wait for 3 seconds.
- Connect the KNX bus to the device again.
- Pres and old the [ ] and [RST] button at the same time for nearly 5 seconds until see the programming LED blinks.
- Blinking programming LED visualizes the successful reset of the device to factory default settings. The device performs a master reset, and device is ready for operation again after approx. 5 second.

Note: Device allows factory reset only for 12 seconds after power up.

5.7 Manual Control

Manual control is available only after enabling it in ETS. To enter the manual control mode, press [ ] button. An output's LED will start to blink indicating that the manual control is operated on that output. Press [← →] buttons to change the position of the controlled output to the right and left. Likewise, press [↕]button to change the position of the controlled output to the up and down. Press [SET] and [RST] buttons to simulate setting and resetting an output signal. Press [ ] button to exit the manual control mode.

Note: The KNX telegrams that comes from KNX bus do not accept while manual control is activated.

6 Connection

6.1 Dry Contact Connection

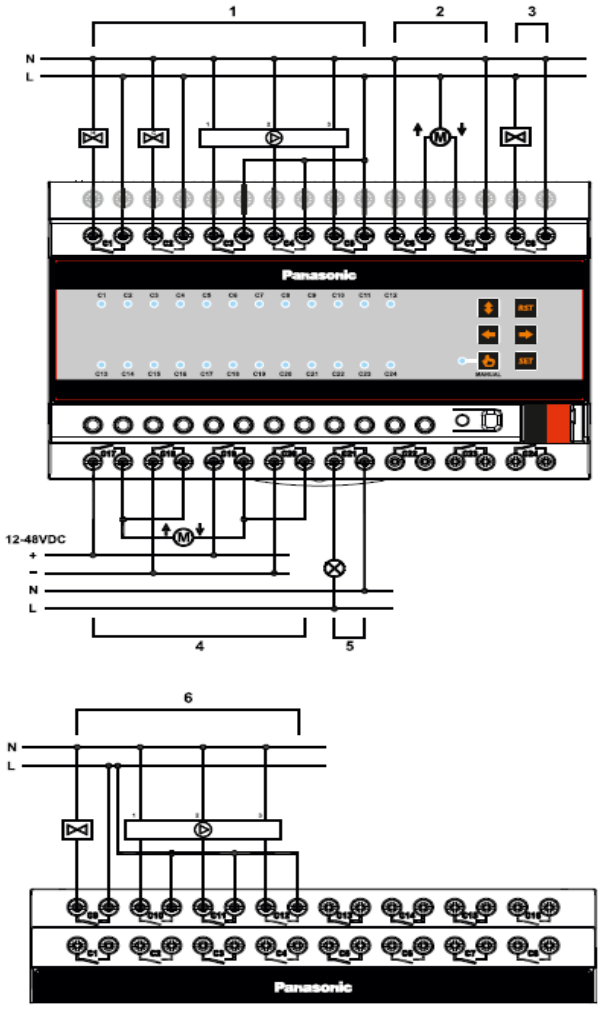


Figure 2 Dry Contact Connection for MX324-16A-S

1. **Fan Coil (Heating and Cooling)**
   4 Pipe uses 5 output channels. Channel 1 to channel 5 are used for fan coil. Channel 1 uses for heating, Channel 2 uses for cooling, Channel 3, Channel 4 and channel 5 uses for fan steps.
2. **Shutter/Blind**
   Channel 6 and channel 7 are used for shutter/blind connection.
3. **Heating**
   Channel 8 is used for heating function.
   NOTE: For the outputs use circuit breakers for the respective rated current.
4. **Shutter/Blind DC**
   4 output channels are used.
   Channel 9 to channel 12 are used for shutter/blind DC connection.

***To avoid short circuit, make sure that the ETS configuration of the channels which connected to the DC motor are done correctly.**
***Fuse or short circuit protection should be used with DC power supply.**

5. **Switching**
   Channel 13 is used for switching – lighting function.
6. **Fan Coil (Heating and Cooling)**
   2 Pipe uses 4 output channels. Channel 1 to channel 4 are used for fan coil. Channel 1 uses for heating or cooling, Channel 2, channel 3 and channel 3 uses for fan step.

6.2 Technical information

Power	
Operating voltage	DC 21-32 V (from KNX bus)
Current consumption	Max. 40mA
Environmental conditions	
Ambient temperature	-5 C …+45 C
Storage temperature	-10 C … +55 C
Ambient humidity	5…93% (non-condensing)
Housing	
Dimensions (HxWxD)	90mmx143,8mmx65,8mm
Mounting (IEC60715)	35 mm top-hat rail (TH35)
Mounting width	DIN rail 144mm (8 modules)
KNX bus connection	KNX connector (243-211 Wago)
Connection type	Screw terminal Single wire: 1.5mm^2-4mm^2 or 2x1.5mm^2-2x2.5mm^2 Stranded wire without ferrule: 0.75mm^2-4mm^2 Stranded wire without ferrule: 0.5mm^2-2.5mm^2
Weight	8 Channels – 420gr 12 Channels – 495gr 16 Channels – 570gr 20 Channels – 645gr 24 Channels – 720gr
Input/Output	
24 x 16A (NO) Relay Contact	Potential-free closing contacts, µ contact, 230V AC 50/60 Hz, 16A (PF=1) max. 20ms (incand. Lamps) 165A peak max. 200µs 800A peak
Electrical safety	
Protection type (IEC60529)	IP 20
Pollution degree (IEC60664)	2
Protection class (IEC61140)	II
Overvoltage category (IEC60664)	III

Standards	
EMC	EN 60669-2-1
LVD	EN 60669-2-1
KNX	EN 50090

Table 3

Note: Protect the switch contacts with a series-connected miniature circuit-breaker of characteristic B for a max. rated current of 16A.

6.3 Product versions

Product Features	WRKT4808J (MX308)	WRKT4812J (MX312)	WRKT4816J (MX316)	WRKT4820J (MX320)	WRKT4824J (MX324)
Switching - Lighting	Up to 8	Up to 12	Up to 16	Up to 20	Up to 24
Switching – Heating	Up to 8	Up to 12	Up to 16	Up to 20	Up to 24
Shutter Blind AC	Up to 4	Up to 6	Up to 8	Up to 10	Up to 12
Shutter Blind DC	Up to 2	Up to 3	Up to 4	Up to 5	Up to 6
Fan Coil 2-pipe (*)	Up to 4	Up to 6	Up to 8	Up to 10	Up to 12
Fan Coil 4-pipe (*)	Up to 2	Up to 4	Up to 5	Up to 6	Up to 8
Auxiliary Functions (v2.0)	24	24	24	24	24

Table 4

(*) when only one fan level is controlled.

6.4 Dimensional drawings

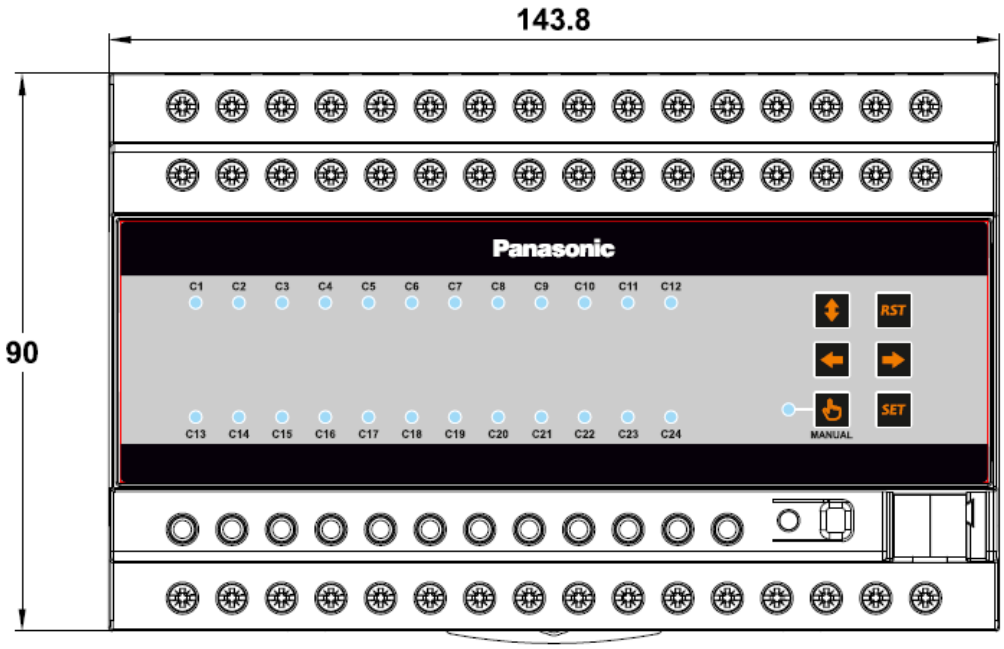


Figure 3 Front dimension for MX324-16A-S

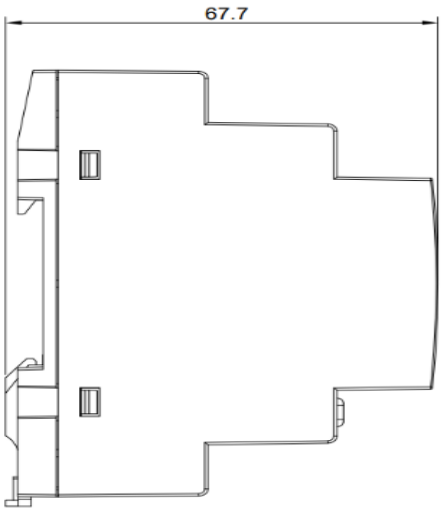


Figure 4 Side dimension for MX324-16A-S

7 Product database

Manufacturer	Panasonic
Product family	Actuator
Product type	Mix Actuator
Product name	KNX Secure Mix Actuator MX324-16A-S KNX Secure Mix Actuator MX320-16A-S KNX Secure Mix Actuator MX316-16A-S KNX Secure Mix Actuator MX312-16A-S KNX Secure Mix Actuator MX308-16A-S

Table 5

8 ETS Database

8.1 General Settings

8.1.1 General Settings Parameters

Name	Values	Description
Startup delay	0…**1**…250 s	This parameter defines the delay time for startup in seconds. After bus voltage recovery, the device always waits the delay time to expire before sending telegrams to the bus and start operation.
Send device in operation telegram	**No** Yes	Selecting Yes enables “Device In Operation” object.
Device in operation telegram value	**Off** On	This parameter specifies the value of the telegram that will be sent from “Device In Operation” object.
Telegram cycle time unit	Second **Minute** Hour Day	This parameter sets the unit of the time interval at which the “Device In Operation” group object sends a telegram cyclically.
Telegram cycle time value	**1**…255	This parameter sets the value of the time interval at which the “Device In Operation” group object sends a telegram cyclically.
Enable telegram limitation	**No** Yes	This parameter is used to enable telegram limitation. Telegram limitation is used to limit the telegrams that are sent by the device to the bus in order to decrease the load on KNX bus.
Telegram limit time	1 second 5 seconds 10 seconds 30 seconds **1 minute** 5 minutes 10 minutes 30 minutes 60 minutes	This parameter determines the period of the telegram limitation. Only a specified number of telegrams will be sent in this period. Other telegrams will be postponed to the next period.
Telegram limit number	1…**10**…255	This parameter determines the maximum number of telegrams that can be sent by the device in telegram limit period.
Enable request status values	**No** Yes	This parameter enables “Request Status Values” object that is used to send “Manual Control Status” and “Input – Status” objects values to the bus.
Send request status values when request object value is	Off **On** Off or on	This parameter determines the value of “Request Status Values” object that will trigger sending status objects values.
Manual control	**Enabled** Disabled Enable/Disable via communication object	This parameter enables or disables the manual control on the device. If “Enabled” is selected, the user can enter manual control mode via manual operation buttons.

Name	Values	Description
		If “Disabled” is selected, the user cannot enter manual control mode via manual operation buttons. If “Enable/Disable via communication object” is selected, the operator can use “Manual Control – Enable/Disable” object to allow or prevent entering manual control mode.
Manual control after bus return	Disabled **Enabled** Read from bus As before bus failure.	This parameter determines the status of the manual control after bus voltage return. If “Read from bus” is selected, the device will send a read request for “Manual Control – Enable/Disable” object, if no response is received the manual control will be disabled.
Enable “Manual Control Status – Started / Stopped” object	**No** Yes	This parameter enables “Manual Control Status” object that is used to indicate when the manual control mode is started or stopped.
Send manual control status	Do not send, update only **On change**	“Don’t send, update only”: “Manual Control Status” object value is updated when the manual control mode is started or stopped but not sent to the bus. The user can read the object value or send a telegram to “Request Status Values” object to get manual control status. On change: Manual control status is sent to the bus when the manual operation is started or stopped.
Stop manual control automatically	Not active After 1 minute After 5 minutes **After 10 minutes** After 30 minutes After 60 minutes	This parameter is used to stop the manual control mode automatically after a specific time from the last pressing of manual operation buttons.
Power saving mode for LEDs	**Not active** After 1 minute After 5 minutes After 10 minutes After 30 minutes After 60 minutes	This parameter is used to activate power saving mode for LEDs. After the specified time, the device enters power saving mode where it turns off all input status LEDs. The device exits power saving mode and displays inputs status with LEDs when a manual operation button is pressed.

Table 6

8.1.2 General Group Objects

No	Object Name	Function	Size	Datapoint Type	Flags				
					C	R	W	T	U
1	*Device In Operation*	*Trigger*	1 Bit	1.001 Switch	C			T	

Table 7

This object is available if “Send device in operation telegram” parameter is set to “Yes”. It is used to monitor the presence of the device on KNX bus. After startup delay time the device sends telegrams cyclically to this object according to “Device in operation telegram value”, “Telegram cycle time unit” and “Telegram cycle time value” parameters values.

No	Object Name	Function	Size	Datapoint Type	Flags				
					C	R	W	T	U
2	*Request status values*	*Send values*	1 Bit	1.001 Switch	C		W		

Table 8

This object is available if “Enable request status values object” parameter is set to “Yes”.
When this object receives a proper telegram according to “Send status values when request status object value is” parameter, all enabled input status objects and manual control status object values are sent to the bus.

No	Object Name	Function	Size	Datapoint Type	Flags				
					C	R	W	T	U
3	*Manual Control*	*Enable / Disable*	1 Bit	1.003 Enable	C		W		

Table 9

This object is available if “Manual Control” parameter is set to “Enable/Disable via communication object”.
If this object receives “Enable” telegram, the user will be able to start manual control mode via manual operation buttons.
If this object receives “Disable” telegram, the user will not be able to start manual control mode via manual operation buttons.

No	Object Name	Function	Size	Datapoint Type	Flags				
					C	R	W	T	U
4	*Manual Control Status*	*Started / Stopped*	1 Bit	1.010 Start/Stop	C	R		T	

Table 10

This object is available if “Enable “Manual Control Status – Started / Stopped” object” parameter is set to “Yes”.
This object indicates the status of the manual control mode.

8.1.3 **Manual Control**

The status of the outputs can be simulated and overridden in manual control mode with manual operation buttons. In manual control mode, the output which its status is overridden will preserve the new status until the device exits manual control mode. The new status is displayed with the status LED too.

Manual control is available only after enabling it in ETS. To enter the manual control mode, press [ ] button. An output’s LED will start to blink indicating that the manual control is operated on that output. Press [ ] buttons to change the position of the controlled output to the right and left. Likewise, press [ ]button to change the position of the controlled output to the up and down. Press [SET] and [RST] buttons to simulate setting and resetting an output signal. Press [ ] button to exit the manual control mode.

8.2 Enable Outputs Page

In this page the outputs of the device can be enabled and its functionalities can be selected from the next types:

1. Switching (Lighting)
2. Switching (Heating)
3. Shutter/Blind (AC)
4. Shutter/Blind (DC)
5. Fan Coil (2-Pipe)
6. Fan Coil (4-Pipe)

In case of enabling a fan coil function on an output, “Number of fan levels” parameter will be shown to specify the number of levels (steps) of the controlled fan in the fan coil system.

8.3 Switching (Lighting) Parameters and Group Objects

8.3.1 Lighting Settings Parameters

Name	Values	Description
Output name		The user can give an output an optional name that describes its functionality or the connected device. This parameter value has no effect on the channel work.
Use template parameters	**No** Yes	If yes is selected the parameter values in “Template -> Switching (Lighting)” page will be used for the configured output and the parameters of this output will be hidden.
Contact type	**Normally open** Normally close	The relay of a switching output can be configured as noramally open or normally closed contacts. In this way, the inversion of switching states is possible. Normally open: • Switching state = off -> Relay contact is open • Switching state = on -> Relay contact is closed Normally closed: • Switching state = off -> Relay contact is closed • Switching state = on -> Relay contact is open
Function type	**Switch on/off** Time delay Flashing Staircase	Function selection for lighting. An additional parameter page will be shown depending on selected option.
Feedback	**Disabled** Enabled	This parameter can be used to disable or enable the feedback function. When the function is enabled the relevent parameter will be available.
Behaviour	**Disabled** Enabled	This parameter can be used to disable or enable the behavior function.

Name	Values	Description
		When the function is enabled the relevent parameter will be available.
Scene	**Disabled** Enabled	This parameter can be used to disable or enable the scene function. When the function is enabled the relevent parameter will be available.
Logic input	**Disabled** Enabled	This parameter can be used to disable or enable the logic input function. When the function is enabled the relevent parameter will be available.
Working hours counter	**Disabled** Enabled	This parameter can be used to disable or enable the working hours counter function. When the function is enabled the relevent parameter will be available.
Joining Central functions	**Disabled** Enabled	This parameter can be used to disable or enable the central functions. When the function is enabled the relevent parameter will be available.

Table 11

8.3.2 Time Delay Parameters

Name	Values	Description
Enable / Disable function with object	**No** Yes	If “Yes” is selected “Output x – Enable Function” object will be shown. This object can be used to disable the function of the output so it will work as normal switch on/off function. Note: By default, the function is enabled after ETS download operation and it restores its previous status after bus voltage return.
Time delay is applied for	**Switching on** Switching off Switching on & off	Switching on: A time delay will be applied after receiving on telegrams only. Its working principle is shown on Figure 4. Switching off: A time delay will be applied after receiving off telegrams only. Its working principle is shown on Figure 5. Switching on & off: A time delay will be applied after receiving on and off telegrams. Its working principle is shown on Figure 6.
Switching on delay time	00:00:01…**00:00:10**…09:06:07 (hh:mm:ss)	This parameter is used for setting the duration of the switch on delay.
Retriggerable	**No** Yes	Yes: A switch on delay still in progress can be retriggered by another "ON" telegram. No: the retriggering time is suppressed.
Switching off delay time	00:00:01…**00:01:00**…09:06:07 (hh:mm:ss)	This parameter is used for setting the duration of the switch off delay.

Name	Values	Description
Retriggerable	**No** Yes	Yes: A switch off delay still in progress can be retriggered by another "OFF" telegram. No: the retriggering time is suppressed.

Table 12

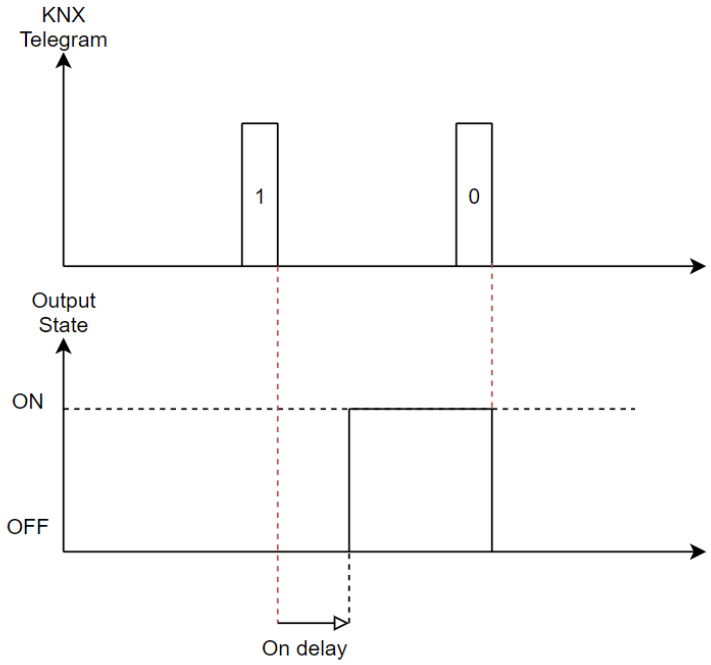


Figure 5 On delay

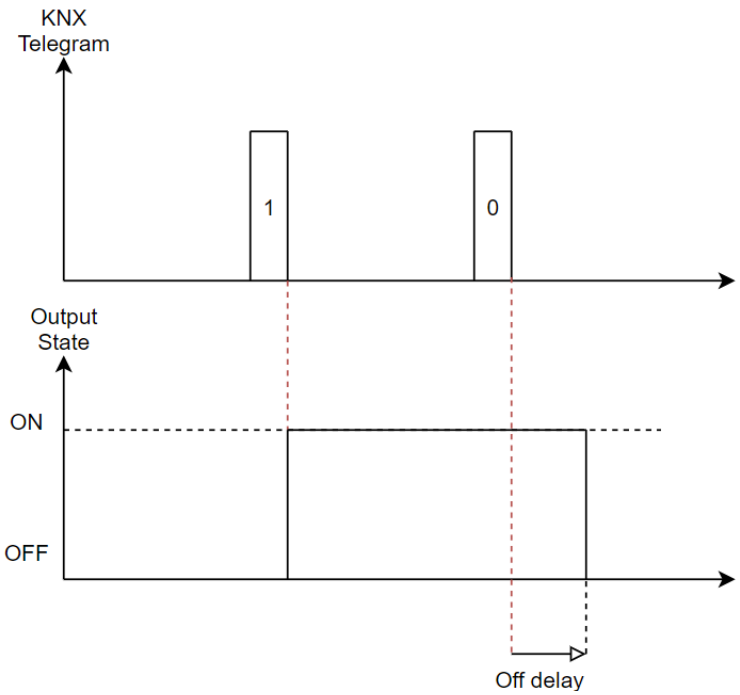


Figure 6 Off delay

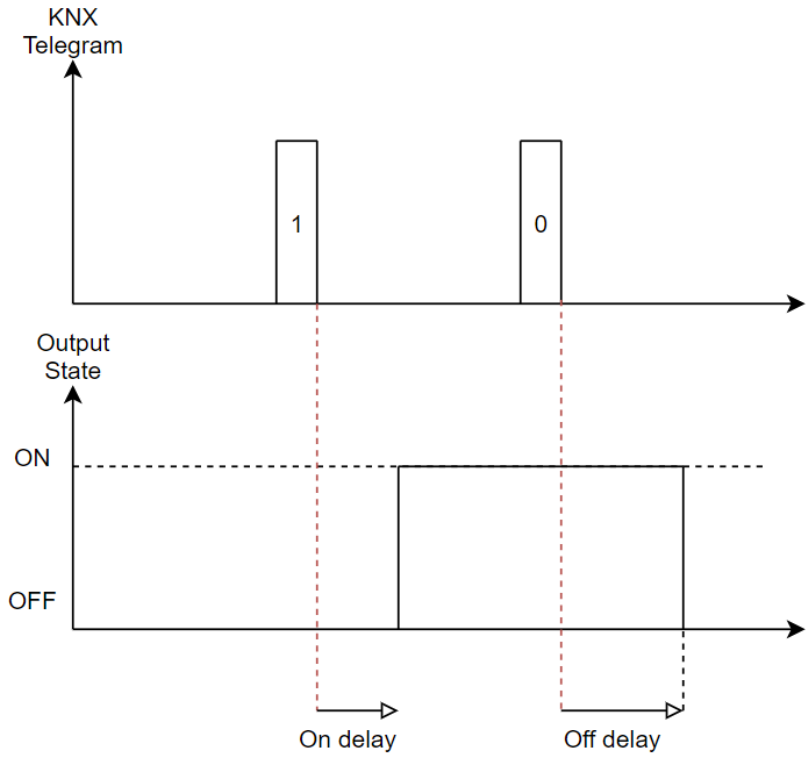


Figure 7 On&Off delay

8.3.3 Flashing Parameters

Name	Values	Description
Enable / Disable function with object	**No** Yes	If "Yes" is selected "Output x – Enable Function" object will be shown. This object can be used to disable the function of the output so it will work as normal switch on/off function. Note: By default, the function is enabled after ETS download operation and it restores its previous status after bus voltage return.
On time	**00:00:01**…09:06:07 (hh:mm:ss)	This parameter specifies the amount of time the output stays on during flashing periode. Flashing function working principle is shown on Figure 7.
Off time	**00:00:01**…09:06:07 (hh:mm:ss)	This parameter specifies the amount of time the output stays off during flashing periode. Flashing function working principle is shown on Figure 8.

Table 13

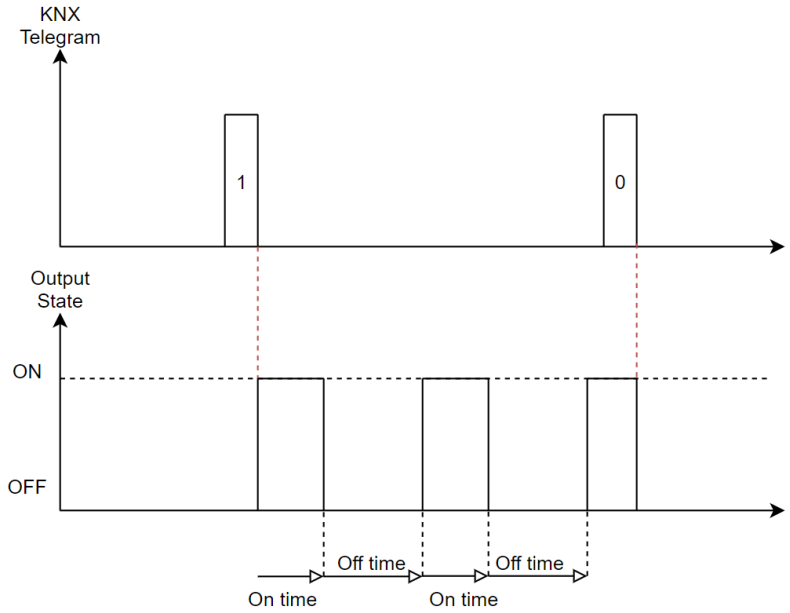


Figure 8 Flashing function

8.3.4 Staircase Parameters

Name	Values	Description
Enable / Disable function with object	**No** Yes	If "Yes" is selected "Output x – Enable Function" object will be shown. This object can be used to disable the function of the output so it will work as normal switch on/off function. Note: By default, the function is enabled after ETS download operation and it restores its previous status after bus voltage return.
Staircase time	00:00:01…**00:00:10**…09:06:07 (hh:mm:ss)	This parameter specifies the amount of time the output stays on after receiving ON telegrams. When staircase time elapses, the output is switched off automatically. Staircase function working principle is shown on Figure 8.
Retirggerable	**No** Yes	Yes: Receiving new ON telegrams during an active staircase time retriggeres it (reset the timer). No: Receiving new ON telegrams during an active staircase time has no effect.

Table 14

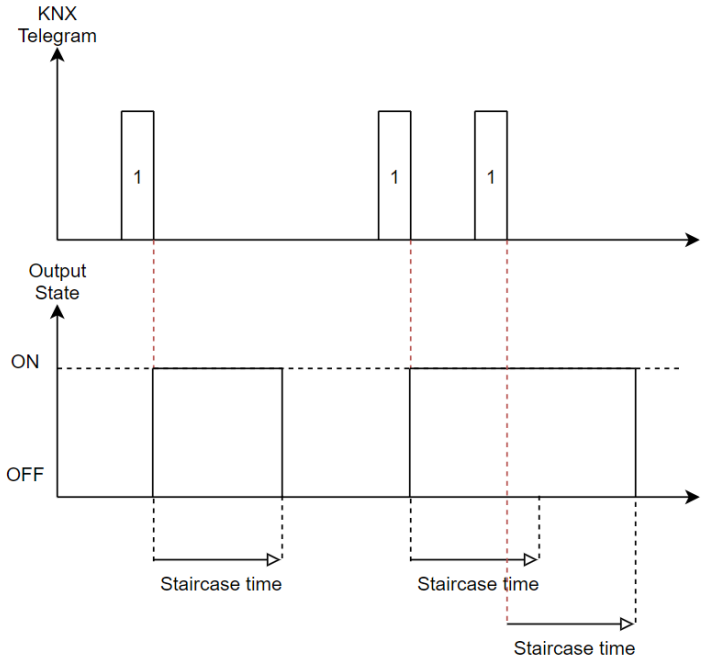


Figure 9 Staircase function

8.3.5 **Feedback Page**

Name	Values	Description
Feedback type	**Normal** Inverted	Normal: Sends the state of the output Inverted: Sends the inverted state of the output
Send feedback telegrams	Do not send, update only **Send on change**	"Don't send, update only": "Feedback" object value is updated when the output status is changed but not sent to the bus. The user can read the object value to get the output status. On change: The new output status is sent to the bus when it changes.
Send feedback cyclically	**No** Yes	This parameter enables sending the output status cyclically to the bus.
Feedback cycle time	00:00:01 … **00:01:00** … 09:06:07	This parameter determines the cycle time between the repeated feedback telegrams.
Limit cyclically sending count	**No** Yes	"No": the feedback telegrams will be sent to the bus cyclically forever. "Yes": the feedback telegrams will be sent to the bus cyclically a certain number of times.
Feedback Cycle count	1…**10**…65535	This parameter determines how many times a feedback telegram will be sent cyclically.

Table 15

8.3.6 **Behaviour Parameters**

The preferred relay contact positions after bus voltage return or after ETS programming can be preset separately for each output. Since the actuator is equipped with bus-independent bitstable relays, the relay switching state at bus voltage failure can be defined as well.

Name	Values	Description
Behaviour after ETS programming	**Off** On Do not change	Off: After ETS programming operation, the related output will be OFF. On: After ETS programming operation, the related output will be ON. Do not change: After ETS programming operation, the related output state will be as before starting the programming operation.
Response to bus voltage failure	Off On **Do not change**	Off: After bus volatage failures, the related output will be OFF. On: After bus volatage failures, the related output will be ON. Do not change: The related output state doesn't change after bus volatage failures.
Behaviour after bus voltage return	**Off** On **As before bus failure**	Off: After bus volatage returns, the related output will be OFF. On: After bus volatage returns, the related output will be ON. As before bus failure: The related output state will be as same as the latest state before bus voltage failure.

Table 16

8.3.7 **Scenes Parameters**

Up to 12 scenes can be programmed and scene values stored separately in the actuator for each switching output. The scene values are recalled or stored via a separate scene extension object by means of extension telegrams. The datapoint type of the extension object permits addressing a maximum of 64 scenes. This means that, in the configuration of a scene, it is possible to specify which scene number (1...64) contacts the internal scene (1...12).

Name	Values	Description
Scene x is assigned to	**Not assigned** Scene number 1..64	This parameter defines the scene number that will trigger the scene function to change the output state
State	**Off** On	This parameter is used for configuring the switching state which will be set when the scene is recalled.
Storage function	**Disabled** Enabled	Disabled: The storage (learn) telegrams are ignored. Enabled: the current switching state can be stored internally via "Scene Input" object when a storage (learn) telegram is received.

Name	Values	Description
Delay	**0**…255 (s)	This parameter is used for setting the duration of the scene delay time. If its value is between 1…255, the received recall scene commands will be delayed before execution.

Table 17

8.3.8 Logic Input Parameters

With the logic function a new communication object “Logic Inpue” is added and logically linked with the object “Switching On/Off”. The values of these two objects are evaluated and then the output will be switched on or off depending on the result of the logic. Logic Input function truth table is shown on Table 18.

Function	Switch On/Off	Logic Input	Output State
AND	0	0	0
	0	1	0
	1	0	0
	1	1	1
OR	0	0	0
	0	1	1
	1	0	1
	1	1	1
XOR	0	0	0
	0	1	1
	1	0	1
	1	1	0

Table 18

Name	Values	Description
Logic gate type	**And** Or Xor	This parameter defines the type of the logic gate.

Table 19

8.3.9 Working Hours Counter Parameters

The working hours counter determines the switch-on time of an output. The totalled working hours are added in an internal meter and stored permanently in the device. The current meter reading can be transmitted cyclically to the bus by the "Working Hour Counter – Counter Value" object or when there is a change in an interval value. An alarm telegram can be sent to the bus when the working hours count reaches a limit value in ascending counters or when it reaches 0 in descending counters.

Name	Values	Description
Counting direction	**Decrement** Increment	Decrement": the counter will be loaded with "Counter start value" at the beginning and it will count down to 0. "Increment": the counter will be loaded with 0 at the beginning and it will count up to "Counter limit value".
Counter start value	1…**1000**…65535	This parameter is available if "Counting direction" parameter is set to "Decrement". It determines the value that the timer will load it at the beginning.
Counter limit value	1…**1000**…65535	This parameter is available if "Counting direction" parameter is set to "Increment". It determines the value that the timer will count up to it.
Counter object type	**4-Byte value in s(DPT 13.100)** 2-Byte value in h(DPT 7.007)	This parameter defines the DPT of "Counter Value" object.
Send counter value	Do not send, update only **When counter value is changed** At specific interval Cyclically only When counter value is changed and cyclically At specific interval and cyclically	This parameter defines when the function will send the counter value to the bus. If "Do not send, update only" is selected, the user can send read request to "Counter Value" object to get the current counter value. If the counter value is set to be sent at specific interval, an additional parameter will be shown to enter the interval value. The counter value will be sent to the bus if its new value is divisible by the interval value. If the counter value is set to be sent cyclically, additional parameters will be shown to enter the cycle time.

Table 20

8.3.10 Joining Central Functions Parameters

Name	Values	Description
Join central continuous off	**No** Yes	No: The output will ignore telegrams sent to "Central Continuous Off – Start/Stop" object. Yes: The output will paritcipate in central continuous off according to telegrams sent to "Central Continuous Off – Start/Stop" object. "Central Continuous Off" object is prior to all other objects. This means that if the "Central Continuous Off" object is set to "Start = 1" the outputs which are participated to central continuous off function are switched off regardless of all other objects (switching object, central switching object …etc). If "Central Continuous Off" object is set to "Start = 1", the participated outputs stay continuously in off position and they can not be swithed on until this object is set to "Stop = 0".

Name	Values	Description
Join central continuous on	**No** Yes	No: The output will ignore telegrams sent to “Central Continuous On – Start/Stop” object. Yes: The output will paritcipate in central continuous on according to telegrams sent to “Central Continuous On – Start/Stop” object. “Central Continuous On” object is prior to all other objects. This means that if the “Central Continuous On” object is set to “Start = 1” the outputs which are participated to central continuous on function are switched on regardless of all other objects (switching object, central switching object …etc). If the central continuous on object is set to “Start = 1”, the participated outputs stay continuously in on position and they can not be swithed off until this object is set to “Stop = 0”.
Join central switching	**No** Yes	No: The output will ignore telegrams sent to “Central Switching – On/Off” object. Yes: The output will paritcipate in central switching according to telegrams sent to “Central Switching – On/Off” object. In central switching, the behaviour of the outputs is identical with 'normal' activation via the "Switching On/Off" object. If “Central Switching – On/Off” object is set to “On = 1”, the outputs which are participated to central switching are switched on and if it is set to “Off = 0” the participated outputs are switched off.

Table 21

8.3.11 Switching (Lighting) Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Switching On/Off*	1 Bit	1.001 Switch	C		W		

Table 22

This object is used to activate a switching output ("1" = Switch on / "0" = Switch off)

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Switching Feedback*	1 Bit	1.001 Switch	C	R		T	

Table 23

This object is available if “Feedback” parameter is set to “Enabled”. It’s used to give feedback on the switching state of an independent switching output ("1" = Switched on / "0" = Switched off).

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Enable Function*	1 Bit	1.003 Enable	C	R	W		

Table 24

This object is available if “Enable/Disable function with object” parameter is set to “Yes” in time delay, flashing or staircase functions’ pages. It’s used to enable/disable the function manually. When the function is disabled the output will act as a normal switch on/off function. By default, the function is enabled after ETS download operation and it restores its previous status after bus voltage return.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Working Hours Counter – Counter Value (Seconds)* *Working Hours Counter – Counter Value (Hours)*	4 Bytes 2 Bytes	13.100 Time lag (s) 7.007 Time (h)	C	R		T	

Table 25

This object is available if “Working hours counter” parameter is set to “Yes”. Its value represents the reached counter value and its DPT is specified by “Counter object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Working Hours Counter – Reset Counter*	1 Bit	1.015 Reset	C		W		

Table 26

This object is available if “Working hours counter” parameter is set to “Yes”. It’s used to reload the counter with its start value if “Counting direction” is “Decrement” or with 0 value if “Counting direction” is “Increment” when it receives a “Reset command = 1” telegram.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Working Hours Counter - Alarm*	1 Bit	1.005 Alarm	C	R		T	

Table 27

This object is available if “Working hours counter” parameter is set to “Yes. It’s used to send alarm when counter value reaches 0 if “Counting direction” is “Decrement” or when the counter value reaches “Counter limit value” if “Counting direction” is “Increment”.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Logic Input*	1 Bit	1.001 Switch	C		W		

Table 28

This object is available if “Logic input” parameter is set to “Yes. It represents the second input of the logical operation (The first input is “Switching On/Off” object).

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Scene Input*	1 Byte	18.001 Scene Control	C		W		

Table 29

This object is available if “Scenes” parameter is set to “Yes. It’s used to recall (activate) or save (learn) a scene.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Reset Storaged Scenes*	1 Bit	1.015 Reset	C		W		

Table 30

This object is available if “Scenes” parameter is set to “Yes”. It’s used to reload the storaged states of scenes with its original parameter values.

8.4 Switching (Heating) Parameters and Group Objects

The switching heating functionality in mix actuators is used for heating systems such as underfloor heating, water based heating and electrical heating.

8.4.1 Priorities for Actauting Values in Switching (Heating) Outputs

The Switching (Heating) actuator distinguishes between various functions that can have an effect on the output. In order to prevent conflicting states, each available function has a certain priority. The function with the higher priority overrides the one with the lower priority.

Taking into account all possible functions, the following priorities are defined:

1. Manual mode
2. Forced mode
3. Valve protection
4. Summer mode
5. Monitoring failure
6. Actuating values from behavior functions or from “Actuating Value” object.

8.4.2 Heating Settings Parameters

Name	Values	Description
Output name		The user can give an output an optional name that describes its functionality or the connected device. This parameter value has no effect on the channel work.

Name	Values	Description
Use template parameters	**No** Yes	If yes is selected the parameter values in “Template -> Switching (Heating)” page will be used for the configured output and the parameters of this output will be hidden.
Valve state when output’s contact are closed	**Open** Closed	This parameter is used to determine the operation mode of the controlled valve. Open: The valve is open when it’s energized. Closed: The valve is closed when it’s energized.
Type of control	**Switching (1-bit)** Continuous PWM (1-byte)	The heating output receives 1-bit or 1-byte command value telegrams transmitted, for example, by KNX room temperature controllers. Usually, the room temperature controller determines the room temperature and generates the command value telegrams using a control algorithm. The actuator controls its valve outputs either in switching form or with a PWM signal. Switching (1-bit): the telegram received via “Actuating Value” object is used directly in controlling the output. If on telegram is received the valve is opened, and if off telegram is received the valve is closed. Continuous PWM (1-byte): the telegram received via “Actuating Value” object is used to generate pulse-width-modulated switch signal at the valve output according to the set parameters in page “Continuous PWM Control Settings”.
Feedback	**Disabled** Enabled	This parameter is used to disable or enable the feedback function. When the function is enabled the relevent parameter will be available.
Winter / Summer modes	**Disabled** Enabled	This parameter is used to disable or enable summer/winter modes. When the function is enabled the relevent parameter will be available. In summer mode, the valve is closed and the values received from “Actuating Value” object are ignored. When winter / summer mode is enabled the user can set different actuating values for summer and winter seasons in forced mode settings.
Forced mode	**Disabled** Enabled	This parameter is used to disable or enable the forced mode. When the function is enabled the relevent parameter will be available. Forced mode is used to block any telegram from the KNX bus line from changing the position of the output.
Valve protection	**Disabled** Enabled	This parameter is used to disable or enable valve protection function. When the function is enabled the relevent parameter will be available. This function prevents the valve from seizing and is executed if the valve is closed for a long time. When this function is executed, the valve is open for a short period.

Name	Values	Description
Monitoring	**Disabled** Enabled	This parameter is used to disable or enable monitoring function. When the function is enabled the relevent parameter will be available. If monitoring is enabled, the heating output must receive an actuating value telegram regularly. If no new actuating value is received within a configured monitoring time, failure of the room thermostat (actuating value sender) is assumed and a configurable fixed actuating value is used.
Working hours counter	**Disabled** Enabled	This parameter is used to disable or enable the working hours counter function. When the function is enabled the relevent parameter will be available.
Behavior	**Disabled** Enabled	This parameter is used to disable or enable the behavior function. When the function is enabled the relevent parameter will be available.
Joining heating pump control	**Disabled** Enabled	Mix actuator allows switching activation of a circulation pump of the heating circuit via a 1-bit KNX telegram. The pump is only switched on by the actuator via "Central Heating Pump Control" object, when at least one valve of the participated outputs is open. The pump is switched off when all participated outputs' valves are closed. Disabled: The related output state will not be included in the pump control function. Enabled: The related output state will be included in the pump control function.

Table 31

8.4.3 **Continuous PWM Control Settings Parameters**

Name	Values	Description
PWM period	1…**15**…60 **min**	This parameter specifies the switching frequency of the pulse-width-modulated output signal of a valve output.
Actuating value limitation	**Disabled** Enabled	This parameter is used to enable or disable a limitation on the value received from "Actuating Value" object.
Minimum actuating value	**0**…100	Lowest permissible actuating value.
Maximum actuating value	0…**100**	Highest permissible actuating value.
When received actuating value violates min/max values	**Use min / max values** If < min use 0% / if > max use 100% If 0% use 0% else use min / max values	Use min / max values: Restrict values to the set maximum and minimum actuating value. If < min use 0% / if > max use 100%: Actuate channel with 0% when the received value is lower than the minimum actuating value, actuate channel with 100% when the received value is higher than the maximum actuating value. If 0% use 0% else use min / max values: If the received actuating value is 0%, accept this value and close the valve. Other values are

Name	Values	Description
		restricted according to the configured minimum and maximum actuating values.

Table 32

8.4.4 Feedback Parameters

Name	Values	Description
Send feedback telegrams	Do not send, update only **Send on change**	"Don't send, update only": "Feedback" object value is updated when the used actuating value is changed but not sent to the bus. The user can read the object value to get the used actuating value. On change: The used actuating value is sent to the bus when it changes.
Send feedback cyclically	**No** Yes	This parameter enables sending the used actuating value cyclically to the bus.
Feedback cycle time	00:00:01 … **00:01:00** … 09:06:07	This parameter determines the cycle time between the repeated feedback telegrams.
Limit cyclically sending count	**No** Yes	"No": the feedback telegrams will be sent to the bus cyclically forever. "Yes": the feedback telegrams will be sent to the bus cyclically a certain number of times.
Feedback Cycle count	1…**10**…65535	This parameter determines how many times a feedback telegram will be sent cyclically.

Table 33

8.4.5 Winter / Summer Modes Parameters

Name	Values	Description
Winter / Summer polarity	**Winter = 0, Summer = 1** Winter = 1, Summer = 0	This parameter sets the telegram polarity of the "Winter / Summer" object.
Season after bus return	Winter Summer Read from bus **As before bus failure**	This parameter determines the selected season after bus voltage return. If "Read from bus" is selected, the device will send a read request for "Winter / Summer" object, if no response is received winter season will be selected.

Table 34

8.4.6 **Forced Mode Parameters**

Name	Values	Description
Activate forced mode with value	0 **1**	This parameter defines the telegram value that will enable forced mode when it received on “Forced Mode” object.
Forced mode valve state	**Do not change** Valve is open Valve is closed	This parameter is available if “Type of control” parameter is set to “Switching (1-bit)” and “Winter / Summer Modes” parameter is set to “Disabled”. It defines the valve state when forced mode is enabled.
Forced mode valve state in winter	**Do not change** Valve is open Valve is closed	This parameter is available if “Type of control” parameter is set to “Switching (1-bit)” and “Winter / Summer Modes” parameter is set to “Enabled”. It defines the valve state when forced mode is enabled in winter.
Forced mode valve state in summer	**Do not change** Valve is open Valve is closed	This parameter is available if “Type of control” parameter is set to “Switching (1-bit)” and “Winter / Summer Modes” parameter is set to “Enabled”. It defines the valve state when forced mode is enabled in summer.
Forced mode actuating value	**Do not change** Fixed value	This parameter is available if “Type of control” parameter is set to “Continuous PWM (1-byte)” and “Winter / Summer Modes” parameter is set to “Disabled”. It defines the actuating value that will be used when forced mode is enabled. If “Fixed values” is selected another parameter will be shown to enter the actuating value.
Forced mode actuating value in winter	**Do not change** Fixed value	This parameter is available if “Type of control” parameter is set to “Continuous PWM (1-byte)” and “Winter / Summer Modes” parameter is set to “Enabled”. It defines the actuating value that will be used when forced mode is enabled in winter. If “Fixed values” is selected another parameter will be shown to enter the actuating value.
Forced mode actuating value in summer	**Do not change** Fixed value	This parameter is available if “Type of control” parameter is set to “Continuous PWM (1-byte)” and “Winter / Summer Modes” parameter is set to “Enabled”. It defines the actuating value that will be used when forced mode is enabled in summer. If “Fixed values” is selected another parameter will be shown to enter the actuating value.

Table 35

8.4.7 **Valve Protection Parameters**

Name	Values	Description
Valve protection period	1…**7**…255 day	If the valve is not opened at least once during this period, then the actuator will execute valve protection function and open the valve for specific time.
In valve protection, valve is open for	**1**..60 min	This parameter defines how long the valve remains open during the valve protection procedure.

Table 36

8.4.8 **Monitoring Parameters**

Name	Values	Description
Monitoring time	1…**30**…255 min	A monitoring failure will be detected if no new actuating value is received on "Actuating Value" object within this period. The working principle of monitoring actuating valve is shown in Figure 9.
Monitoring failure value	Monitoring failure = 0 **Monitoring failure = 1**	This parameter is used to select the polarity of monitoring failure. When monitoring failure is detected a telegram with the specified value will be sent on "Monitoring Failure" object.
Valve state in case of monitoring failure	**Valve is closed** Valve is open	This parameter is available if "Type of control" parameter is set to "Switching (1-bit)". This parameter specifies the valve state when a monitoring failure is detected.
Actuating value in case of monitoring failure	**0**…100 %	This parameter is available if "Type of control" parameter is set to "Continuous PWM (1-byte)". This parameter specifies the used actuating value when a monitoring failure is detected.

Table 37

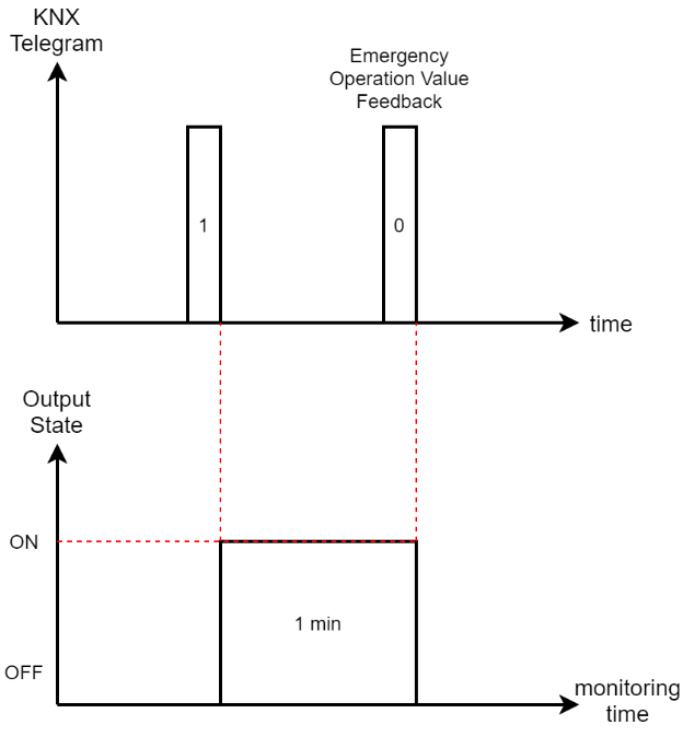


Figure 10 Working principle of monitoring failure

8.4.9 Working Hours Counter Parameters

The working hours counter determines the switch-on time of an output / a valve. The totalled working hours are added in an internal meter and stored permanently in the device. The current meter reading can be transmitted cyclically to the bus by the "Working Hour Counter – Counter Value" object or when there is a change in an interval value. An alarm telegram can be sent to the bus when the working hours count reaches a limit value in ascending counters or when it reaches 0 in descending counters.

Name	Values	Description
Counting direction	**Decrement** Increment	Decrement": the counter will be loaded with "Counter start value" at the beginning and it will count down to 0. "Increment": the counter will be loaded with 0 at the beginning and it will count up to "Counter limit value".
Counter start value	1…**1000**…65535	This parameter is available if "Counting direction" parameter is set to "Decrement". It determines the value that the timer will load it at the beginning.
Counter limit value	1…**1000**…65535	This parameter is available if "Counting direction" parameter is set to "Increment". It determines the value that the timer will count up to it.
Counter object type	**4-Byte value in s(DPT 13.100)** 2-Byte value in h(DPT 7.007)	This parameter defines the DPT of "Counter Value" object.
Send counter value	Do not send, update only	This parameter defines when the function will send the counter value to the bus.

Name	Values	Description
	When counter value is changed At specific interval Cyclically only When counter value is changed and cyclically At specific interval and cyclically	If “Do not send, update only” is selected, the user can send read request to “Counter Value” object to get the current counter value. If the counter value is set to be sent at specific interval, an additional parameter will be shown to enter the interval value. The counter value will be sent to the bus if its new value is divisible by the interval value. If the counter value is set to be sent cyclically, additional parameters will be shown to enter the cycle time.

Table 38

8.4.10 Behaviour Parameters

Name	Values	Description
Valve state after ETS programming	Do not change **Valve is closed** Valve is opened	This parameter is available if “Type of control” parameter is set to “Switching (1-bit)”. Do not change: After ETS programming operation, the related valve state will be as before starting the programming operation. Off: After ETS programming operation, the related valve will be closed. On: After ETS programming operation, the related output will be open.
Actuating value after ETS programming	Do not change **Fixed Value**	This parameter is available if “Type of control” parameter is set to “Continuous PWM (1-byte)”. Do not change: After ETS programming operation, the related valve actuating value will be as before starting the programming operation. Fixed value: After ETS programming operation, the related valve actuating value will be as specified in the next “Actuating value” parameter.
Response to bus failure	**Do not change** Valve is closed Valve is opened	Do not change: The related valve state doesn’t change after bus volatage failures. Valve is closed: After bus volatage failures, the related valve will be closed.

Name	Values	Description
		Valve is opened: After bus volatage failures, the related valve will be open.
Valve state after bus return	**As before bus failure** Valve is closed Valve is open	This parameter is available if “Type of control” parameter is set to “Switching (1-bit)”. As before bus failure: The related valve state will be as same as the latest state before bus voltage failure. Valve is closed: After bus volatage returns, the related valve will be closed. Valve is open: After bus volatage returns, the related valve will be open.
Actuating value after bus return	**As before bus failure** Fixed value	This parameter is available if “Type of control” parameter is set to “Continuous PWM (1-byte)”. As before bus failure: The related valve actuating value will be as same as the actuating value state before bus voltage failure. Fixed value: After bus volatage returns, the related valve actuating value will be as specified in the next “Actuating value” parameter.

Table 39

8.4.11 Switching (Heating) Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Actuating Value*	1 Bit 1 Byte	1.001 Switch 5.001 Percentage	C		W		

Table 40

The actuating value receives data from the room thermostat for the corresponding valve.
It can either be continuous (0-100%) or switching (ON/OFF) depending “Type of control” parameter value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Actuating Value Feedback*	1 Bit 1 Byte	1.001 Switch 5.001 Percentage	C	R		T	

Table 41

This object is available if “Feedback” parameter is set to “Enabled”. It’s used to report the current used actuatuing value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Winter / Summer*	1 Bit	1.001 Switch	C	R	W	T	U

Table 42

This object is available if “Winter / Summer modes” parameter is set to “Enabled”. It’s used to switchover between summer and winter modes. In summer mode, the valve is closed and the values received from “Actuating Value” object are ignored.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Forced Mode*	1 Bit	1.001 Switch	C		W		

Table 43

This object is available if “Forced mode” parameter is set to “Enabled”. It’s used to enable forced mode when it receives a value as specified in “Activate forced mode with value” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Monitoring Failure*	1 Bit	1.001 Switch	C	R		T	

Table 44

This object is available if “Monitoring” parameter is set to “Enabled”. It’s used to report the monitoring failure state. When monitoring failure is detected a telegram with value specified in “Monitoring failure value” parameter will be sent on this object.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Working Hours Counter – Counter Value (Seconds)* *Working Hours Counter – Counter Value (Hours)*	4 Bytes 2 Bytes	13.100 Time lag (s) 7.007 Time (h)	C	R		T	

Table 45

This object is available if “Working hours counter” parameter is set to “Yes”. Its value represents the reached counter value and its DPT is specified by “Counter object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Working Hours Counter – Reset Counter*	1 Bit	1.015 Reset	C		W		

Table 46

This object is available if “Working hours counter” parameter is set to “Yes”. It’s used to reload the counter with its start value if “Counting direction” is “Decrement” or with 0 value if “Counting direction” is “Increment” when it receives a “Reset command = 1” telegram.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Working Hours Counter - Alarm*	1 Bit	1.005 Alarm	C	R		T	

Table 47

This object is available if “Working hours counter” parameter is set to “Yes. It’s used to send alarm when counter value reaches 0 if “Counting direction” is “Decrement” or when the counter value reaches “Counter limit value” if “Counting direction” is “Increment”.

8.5 Shutter/Blind Parameters and Group Objects

There are two basic blind mechanisms.

- Shutter (No slats)
- Blind (With slats)

A blind (or venetian blind) has horizontal slats, one above another. They are suspended by strips of cloth called tapes, or by cords, by which all slats in unison can be rotated through nearly 180 degrees. In this drive mode, the slats are directly adjusted by way of mechanical linkage when the height of the blind is changed. The actuator assumes that the slats are completely closed when the blind moves downwards and similarly, the actuator assumes that the slats are completely open when the blind moves upwards, or vice versa. This is dependent of the blind driving type mechanism.

“*Movement duration*” parameter specifies how long the blind travels from 0% to 100%. Depending on “*Movement durations to up and down*” parameter, if it is selected as “Same” upper end position to lower end position movement and lower end position to upper end position movement are same. But if it is selected as “Different” there will be two different parameters for defining movement duration. “M*ovement duration to down”* parameter is defining duration from upper end position to lower end position. “M*ovement duration to down”* parameter is defining the duration from lower end position to upper end position. You can find Upper movement duration and Lower movement duration at Figure 10 and *Figure 11*. And slat position information at *Figure 12*.

If the up/down object is set to “Up = 0” the blind travels upward, and if it is set to “Down = 1” the blind travels downward. While travelling, if the “Step / Stop” object is updated (“1” or “0”) the blind stops travelling.

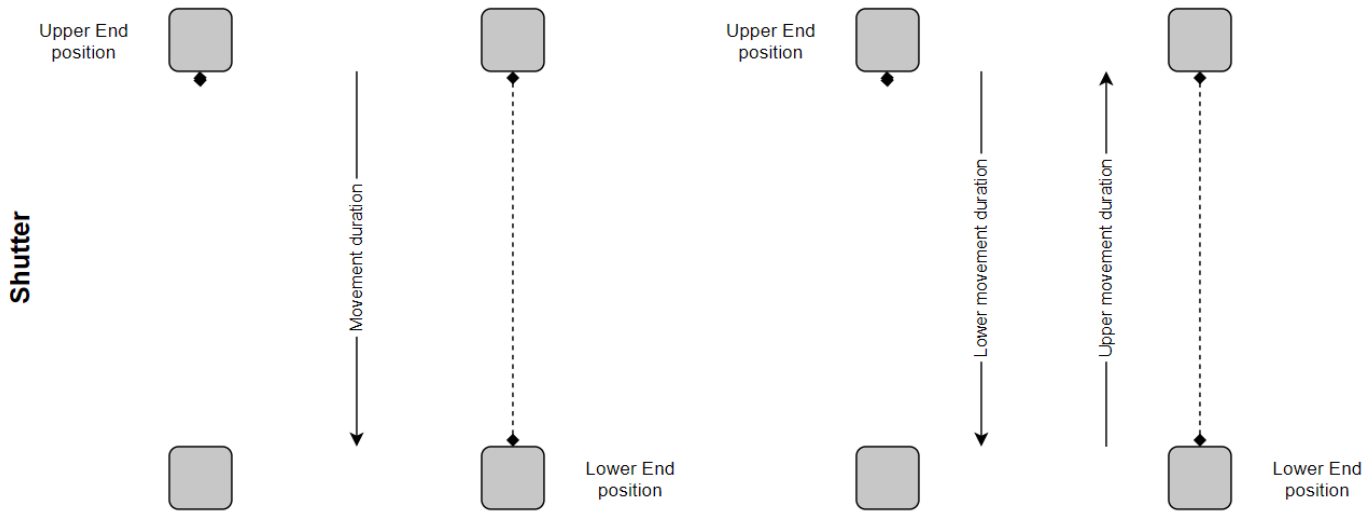


Figure 11 Shutter / Blind - Shutter movement duration and position

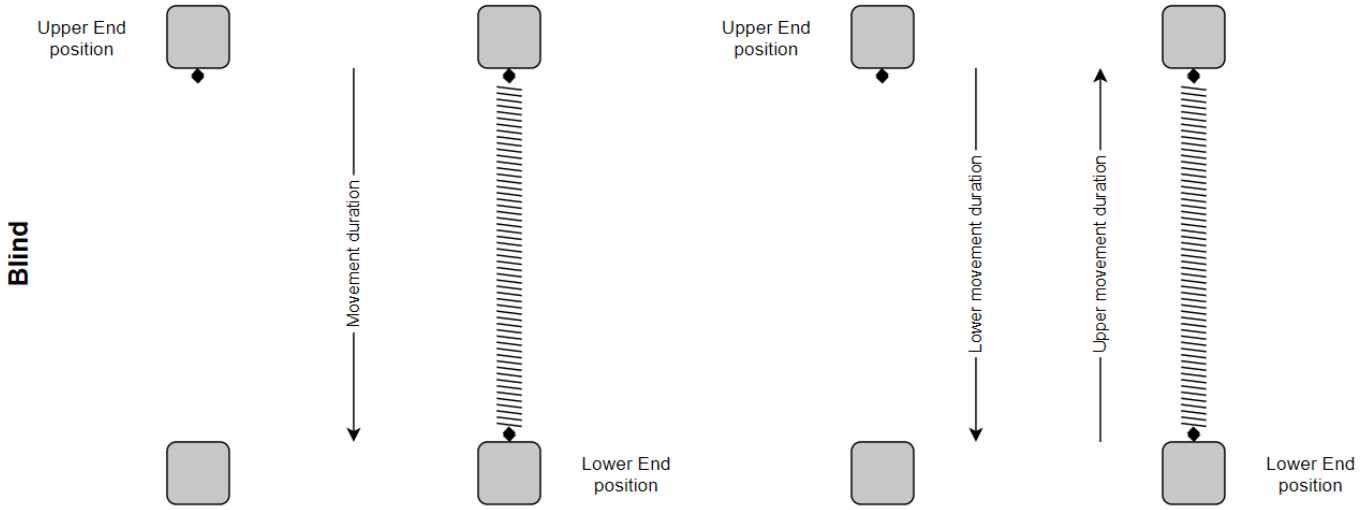


Figure 12 Shutter / Blind - Blind movement duration and position

The default slat positions after up and down monements are shown in Figure 12, however these positions can be inverted with “Slat postion value” parameter.

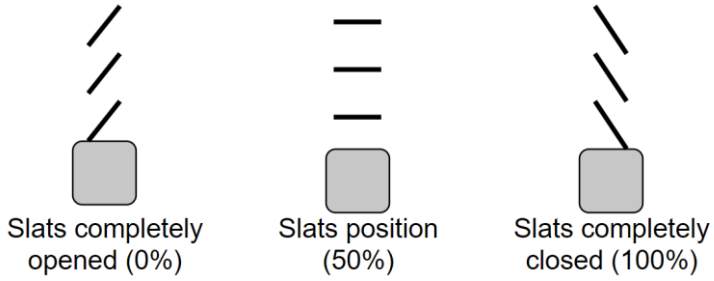


Figure 13 Shutter / Blind – Default slat positions

Mix Actuator can control 230V AC Shutter/Blind and 12…48VDC Shutter/Blind.

12…48VDC Shutter/Blind uses 4 consecutive outputs and 230VAC Shutter/Blind uses 2 consecutive outputs. In this connection diagram Output1, Output2, Output3 and Output4 are used for 12…48VDC Shutter/Blind connection. Output7 and Output8 are used for 230 VAC Shutter/Blind connection.

Position control function is used for adjusting the blind height level for a percent value. For this aim “Shutter / Blind Position” object is used. A value between 0% and 100% can be sent to this object. When a percent value is sent to this object, the blind height level that the blind must reach is calculated according to the complete runtime. The percent adjustment example has been shown below (The blind is at the lower end position):

- Movement duration = 10 seconds
- The value that has been sent to “Shutter / Blind Position” object is %50
- The time that the blind must travels is 5 seconds (10 seconds / 2)

The blind or shutter updown controlling can be realized with central up/down function. The important point here is every blind or shutter outputs must be participated in central up/down function by “Join central up down” parameter. “Central Up / Down” object is used for up/down control of blind/shutter.

The blind actuator can report back via a separate 1-bit communication object per output whether the connected drive is moving. The feedback object has a value of "1" when current is flowing from the output to the drive. Likewise, a "0" is written into the object if the output concerned remains in a stop position.

8.5.1 Shutter / Blind Settings Parameters

Name	Values	Description
Output name		The user can give an output an optional name that describes its functionality or the connected device. This parameter value has no effect on the channel work.
Use template parameters	**No** Yes	If yes is selected the parameter values in “Template -> Shutter / Blind” page will be used for the configured output and the parameters of this output will be hidden.
Operation mode	**Shutter (Without slats)** Blind (With slats)	The actuator can control various drive systems. This parameter defines which type of curtain is connected to the output.
Feedback	**Disabled** Enabled	This parameter can be used to disable or enable the feedback function. When the function is enabled the relevent parameter will be available.
Behaviour	**Disabled** Enabled	This parameter can be used to disable or enable the behavior function. When the function is enabled the relevent parameter will be available.
Scenes	**Disabled** Enabled	This parameter can be used to disable or enable the scene function. When the function is enabled the relevent parameter will be available.
Joining central functions	**Disabled** Enabled	This parameter can be used to disable or enable the central functions. When the function is enabled the relevent parameter will be available.

Table 48

8.5.2 Driving Shuttter / Blind Parameters

Name	Values	Description
Movement duration to up and down	**Same** Different	Same: Upper movement duration and lower movement duration are same. Only one parameter will be shown to enter both up and down movement durations. Different: Upper movement duration and lower movement duration are different. Two parameters will be shown to enter different movenet durations for each up and down movement durations.

Name	Values	Description
Movement duration	5…**50**…600 s	This parameter is only visible when “Movement duration to up and down” parameter is set to “Same”. It defines the complete movement duration from position 0% to position 100%.
Movement duration to up	5…**60**…600 s	This parameter is only visible when “Movement duration to up and down” parameter is set to “Different”. It defines the complete movement duration from position 100% to position 0%.
Movement duration to down	5…**50**…600 s	This parameter is only visible when “Movement duration to up and down” parameter is set to “Different”. It defines the complete movement duration from position 0% to position 100%.
Step time	3…**20**…200 x 100 ms	This parameter defines the duration of step (short-time) operation. Shutter actuator performs a step when it receives a step telegram while it’s not moving. Blind actuator performs a height step when it receives a step telegram while it’s not moving and the slat is on end postion (0% or 100%).
While going up, overrun output with ratio	Do not overrun 2% 10% **20%** 30%	To ensure that the shutter / blind has definitely reached its end position at the end of up movement, the actuator can extend all the up movements into the opened position using the overrun ratio configured here.
While going down, overrun output with ratio	Do not overrun 2% 10% **20%** 30%	To ensure that the shutter / blind has definitely reached its end position at the end of down movement, the actuator can extend all the down movements into the opened position using the overrun ratio configured here.
Time delay at change of direction	**Disabled** Enabled	This parameter enables a delay between changes of direction.
Delay before changing the direction	250**…500**…2000 ms	This parameter specifies the break time in a travel direction change.
Position limitation	**Disabled** Enabled	This parameter enables position limitation feature. Note: This features only works with received telegrams on “Shutter / Blind Position” object.
Lower limit	**0**…100 %	This parameter is available if “Position limitation” parameter is set to “Enabled”. The output ignores all received telegrams on “Shutter / Blind Position” object below this value.
Upper limit	0…**100** %	This parameter is available if “Position limitation” parameter is set to “Enabled”. The output ignores all received telegrams on “Shutter / Blind Position” object higher than this value.

Table 49

8.5.3 **Driving Slats Parameters**

Driving Slats page is visible if the “Operation mode” parameter is set to “Blind (with slats)”.

Name	Values	Description
Slat adjustment time	3…**20**…200 x 100ms	The measured total duration for slat turning from 0% position to 100% position is entered here. The time must be determined as precisely as possible to achieve the best possible result for slat position.
Slat step number	**2**…12	This parameter defines the number of slat adjustments which are required to tilt the slats from fully closed to fully open. The appropriate duration for a slat adjustment is calculated from the duration of a complete slat turn and the desired number of slat adjustments.
Slat position value	**Upper position = 0%, lower position = 100%** Upper position = 100%, lower position = 0%	This parameter defines the values that will be used for slat position after up and down movements.
Restore slat position after up and down movenents	**No** Yes	No: After up movement the slat will stay at the upper position and after down movement the slat will stay at the lower position. Yes: After up and down movements, the slat will return to its old position before up and down movements.

Table 50

8.5.4 **Feedback Parameters**

Name	Values	Description
Enable "Moving Status" object	No Yes	This parameter is used to enable "Moving Status" object that is used to report whether the connected drive is moving, i.e. whether the output is supplying current for any travel direction.
Send feedback telegrams	Do not send, update only **Send on change**	"Don't send, update only": feedback objects values are updated when the moving status or positions are changed but not sent to the bus. The user can read the object values to get the moving status or current position. On change: The new moving status and positions are sent to the bus when they change.
Send position feedbacks cyclically	**No** Yes	This parameter enables sending the positions cyclically to the bus.
Feedback cycle time	00:00:01 … **00:01:00** … 09:06:07	This parameter determines the cycle time between the repeated feedback telegrams.
Limit cyclically sending count	**No** Yes	"No": the feedback telegrams will be sent to the bus cyclically forever. "Yes": the feedback telegrams will be sent to the bus cyclically a certain number of times.
Feedback Cycle count	1…**10**…65535	This parameter determines how many times a feedback telegram will be sent cyclically.

Table 51

8.5.5 **Behaviour Parameters**

Name	Values	Description
Behaviour after ETS programming	**No action** Drive to top Drive to bottom Stop Change shutter / blind position Change slat position Change blind and slat position	This parameter defines the action that will be performed afer ETS programming operation. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.
Response to bus failure	No action Drive to top Drive to bottom **Stop**	This parameter defines the action that will be performed in case of bus voltage failure.
Load old position values after bus return	No **Yes**	This parameter is available only if “Response to bus failure” parameter is set to “Stop”. No: Before the actuator can approach new positions after bus voltage return, the positioning system must at first be calibrated. A position calibration is possible by executing a reference movement which is a complete movement to ther upper or lower position. Yes: Since the response to bus failure action is selected “Stop”, In case of bus voltage failure the actuator will stop driving the shutter / blind if it’s moving and save the reached positions. After bus return the actuator will load the saved reached position values and no reference movement will be required.
Behavior after bus voltage return	Drive to top Drive to bottom **Stop** Change shutter / blind position Change slat position Change blind and slat position Go to before bus failure position	This parameter defines the action that will be performed afer bus voltage return. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.

Table 52

8.5.6 **Weather Alarms Parameters**

To protect the blind/shutter against wind, storms, rains and frost, the actuator can receive alarm telegrams (1 bit). If an alarm occurs, then the blind/shutter can perform a parameterized action and cannot be operated until the occured alarm is deactivated again.

Priorities of weather alarms:

1- Rain alarm
2- Wind alarm
3- Frost alarm

Name	Values	Description
Behavior when frost alarm starts	**No action** Drive to top Drive to bottom Stop Change shutter / blind position Change slat position Change blind and slat position	This parameter defines the action that will be performed when frost alarm starts. The specified action is performed only when there is no other active higher priority alarm. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.
Behavior when frost alarm stops	**No action** Drive to top Drive to bottom Stop Change shutter / blind position Change slat position Change blind and slat position	This parameter defines the action that will be performed when frost alarm stops. The specified action is performed only when there is no other active higher priority alarm. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.
Behavior when wind alarm starts	**No action** Drive to top Drive to bottom Stop Change shutter / blind position Change slat position Change blind and slat position	This parameter defines the action that will be performed when wind alarm starts. The specified action is performed only when there is no other active higher priority alarm. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.
Behavior when wind alarm stops	**No action** Drive to top Drive to bottom Stop Change shutter / blind position Change slat position Change blind and slat position	This parameter defines the action that will be performed when wind alarm stops. The specified action is performed only when there is no other active higher priority alarm. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.
Behavior when rain alarm starts	**No action** Drive to top Drive to bottom Stop Change shutter / blind position Change slat position Change blind and slat position	This parameter defines the action that will be performed when rain alarm starts. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.
Behavior when rain alarm stops	**No action** Drive to top Drive to bottom Stop Change shutter / blind position Change slat position Change blind and slat position	This parameter defines the action that will be performed when rain alarm stops. The specified action is performed only when there is no other alarm active. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.

Table 53

8.5.7 Scenes Parameters

Name	Values	Description
Scene x is assigned to	**Not assigned** Scene number 1..64	This parameter defines the scene number that will trigger the scene function to perform the specified action.
Action	No action Drive to top Drive to bottom Stop Change shutter / blind position Change slat position **Change blind and slat position**	This parameter defines the action that will be performed when the specified scene number is recalled. If one of change position actions is selected more parameters will be available to enter the desired position value. Note: Change slat postion options are only available if “Operation mode” parameter is set to “Blind (with slats)”.
Storage function	**Disabled** Enabled	This parameter is only availabe if “Action” parameter is set to change shutter, blind, slat or both blind and slat postions. By storing a scene (sending learn telegram to “Scene Input” object), the user has the opportunity to change the parameterized positions stored in ETS.
Delay	**0**…255 s	This parameter is used for setting the duration of the scene delay time. If its value is between 1…255, the received recall scene commands will be delayed before execution.

Table 54

8.5.8 Joining Central Functions Parameters

Name	Values	Description
Join central up down	**No** Yes	No: The output will ignore telegrams sent to “Central Up / Down – Up / Down” object. Yes: The output will paritcipate in central up down according to telegrams sent to “Central Up / Down – Up / Down” object. In central up down, the behaviour of the outputs is identical with 'normal' activation via the " Up / Down" object. If “Central Up / Down – Up / Down” object is set to “Up = 0”, the outputs which are participated to central up down goes up and if it is set to “Dwon = 0” the participated outputs goes down.
Join central position	**No** Yes	Yes: The output will paritcipate in central up down according to telegrams sent to “Central Position – Shutter / Blind Position” object. In central position, the behaviour of the outputs is identical with 'normal' activation via the "Central Position – Shutter / Blind Position " object. When “Central Position – Shutter / Blind Position” object receives a position value, the outputs which are participated to central position goes to that position.

Table 55

8.5.9 Shutter / Blind Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Up / Down*	1 Bit	1.001 Switch	C		W		

Table 56

If a telegram with the value (Up = 0) is received at this object, the blind/shutter is moved upwards to the upper end position. If a telegram with the value (Down = 1) is received, the blind/shutter is moved downwards.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Step / Stop*	1 Bit	1.001 Switch	C		W		

Table 57

When a telegram is received at this object the actuator acts as below:

- If the shutter/ blind is moving, it stops.
- If the shutter / blind is not moving:
  1- If the operation mode is set to “Shutter (without slats)”, it goes for one shutter / blind step to the direction specified with the received telegram (Up = 0, Down = 1)
  2- If the operation mode is set to “Blind (with slats)” and the slats are not at the end positions (0% or 100%), it performs a slat movement to the nearst slat step with the direction specified with the received telegram (Up = 0, Down = 1).
  3- If the operation mode is set to “Blind (with slats)” and the slats are at the end positions (0% or 100%), it goes for one shutter / blind step to the direction specified with the received telegram (Up = 0, Down = 1)

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Shutter / Blind Position*	1 Byte	5.001 Percentage	C		W		

Table 58

This object is used to raise/lower the shutter/blind to a certain position (height). The received values are expressed in %. 0% = upper end position and 100% = lower end position.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Shutter / Blind Position Feedback*	1 Byte	5.001 Percentage	C	R		T	

Table 59

This object is available if “Feedback” parameter is set to “Enabled”. It’s used to send the current positioned height of the blind/shutter to the bus. The current position is sent after completion of a movement.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Slat Position*	1 Byte	5.001 Percentage	C		W		

Table 60

This object is available if “Operating mode” parameter is set to “Blind (with slats)”. This object is used to change the slat position to a certain position (height). The received values are expressed in %.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Slat Position Feedback*	1 Byte	5.001 Percentage	C	R		T	

Table 61

This object is available if “Operating mode” parameter is set to “Blind (with slats)” and “Feedback” parameter is set to “Enabled”. It’s used to send the current positioned slat of the blind to the bus. The current position is sent after completion of a movement.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Moving Status*	1 Bit	1.010 Start / Stop	C	R		T	

Table 62

This object is available if “Enable “Moving Status” object” parameter is set to “Yes”. It’s used to report that the connected drive is active, i.e. the output is supplying power to the drive for a travel direction.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Frost Alarm* *Wind Alarm* *Rain Alarm*	1 Bit	1.005 Alarm	C		W		U

Table 63

These objects are available if “Weather Alarms”” parameter is set to “Enabled”. They are used to activate or deactivate a weather alarm.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Scene Input*	1 Byte	18.001 Scene Control	C		W		

Table 64

This object is available if “Scenes” parameter is set to “Yes. It’s used to recall (activate) or save (learn) a scene.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Reset Storaged Scenes*	1 Bit	1.015 Reset	C		W		

Table 65

This object is available if “Scenes” parameter is set to “Yes”. It’s used to reload the storaged positions of scenes with its original parameter values.

8.6 Fan Coil Parameters and Group Objects

Fan coil system refers to the water distribution system serving the in a building. There are two types fan coil systems. The 2 pipes system consists of a single water source connected to two pipes as supply and return, and valve that can function as cold water(CW) or hot water(HW) depending on the mode of the system. This type of fan coil cannot work in cooling and heating mode at the same time. The operation of 2 pipes system is demonstrated in Figure 13.

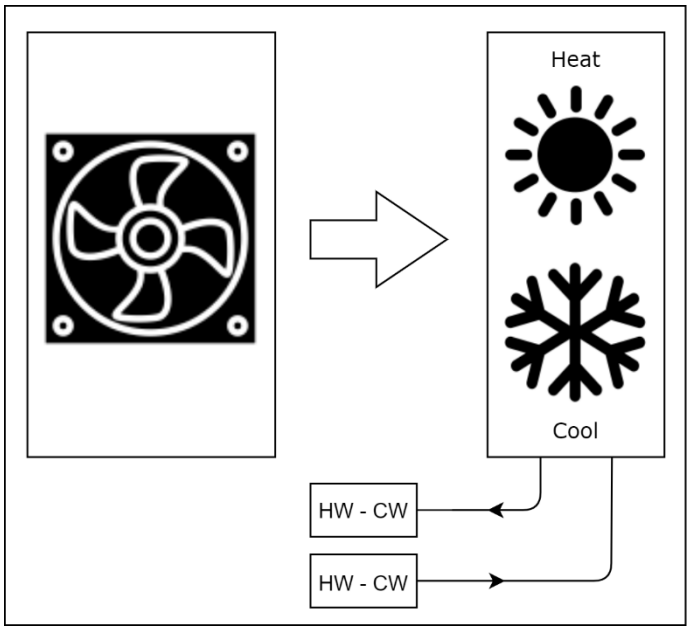


Figure 14 2 pipes fan coil system

The 4 pipe system consists of two separate cold water(CW) and hot water(HW) sources. Each source has its own set of pipes and valves for supply and return. The operation of 4 pipes system is demonstrated in Figure 14.

In both systems, low-noise fans are used to support air circulation and to move the air of the room through the heat exchangers. The fan speed is controllable up to 6 levels.

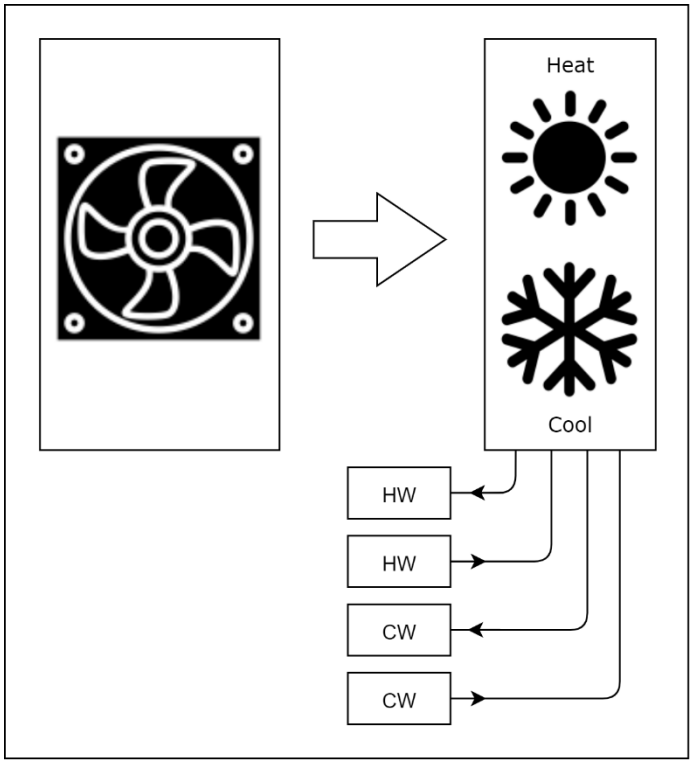


Figure 15 4 pipes fan coil system

8.6.1 Priorities for Actauting Values in Fan Coil Outputs

The fan coil actuator distinguishes between various functions that can have an effect on the output. In order to prevent conflicting states, each available function has a certain priority. The function with the higher priority overrides the one with the lower priority.

Taking into account all possible functions, the following priorities are defined:

1. Manual mode
2. Forced mode
3. Valve protection
4. Montioring failure
5. Actuating values from behavior functions or from “Actuating Value” objects.

8.6.2 Priorities for Fan Level in Fan Coil Outputs

For fan level, the following priorities are defined:

1. Manual mode
2. Forced mode
3. Fan protection
4. Manual fan control
5. Montioring failure
6. Fan level from behavior functions or from “Actuating Value” objects.

8.6.3 **Fan Coil Settings Parameters**

Name	Values	Description
Output name		The user can give an output an optional name that describes its functionality or the connected device. This parameter value has no effect on the channel work.
Use template parameters	**No** Yes	This parameter is available if the number of pipes and fan levels of the fan coil template are equal to the number of pipes and fan levels of the output. If yes is selected the parameter values in "Template -> Fan Coil" page will be used for the configured output and the parameters of this output will be hidden.
Supported fan coil system	**Cooling** Heating **Cooling and Heating**	This parameter determines in what modes of operation (heating or cooling) the output works. Note: If output type is "Fan coil (4-pipe)", "Cooling and Heating" is selected by default.
Manual fan control	**Disabled** Enabled	This parameter can be to used disable or enable the manual fan control function. When the function is enabled the relevant page will be available.
Feedback	**Disabled** Enabled	This parameter can be to used disable or enable the feedback function. When the function is enabled the relevant page will be available.
Forced mode	**Disabled** Enabled	This parameter can be used disable or enable the forced mode. When the function is enabled the relevant page will be available.
Protection	**Disabled** Enabled	This parameter can be used disable or enable the protection function. When the function is enabled the relevant page will be available.
Monitoring	**Disabled** Enabled	This parameter can be used disable or enable the monitoring function. When the function is enabled the relevant page will be available.
Behaviour	**Disabled** Enabled	This parameter can be used disable or enable the behavior function. When the function is enabled the relevent parameter will be available.

Table 66

8.6.4 **Valve Settings Parameters**

Name	Values	Description
Valve state when output's contact are closed	**Open** Closed	This parameter is used to determine the operation mode of the controlled valve. Open: The valve is open when it's energized. Closed: The valve is closed when it's energized.

Name	Values	Description
Type of control	**Switching (1-byte)** Continuous PWM (1-byte)	The heating output receives 1-byte command value telegrams transmitted, for example, by KNX room temperature controllers. Usually, the room temperature controller determines the room temperature and generates the command value telegrams using a control algorithm. The actuator controls its valve outputs either in switching form or with a PWM signal. Switching (1-byte): the telegram received via "Actuating Value" object is used directly in controlling the output. If a telegram with value greater than 0% is received the valve is opened, and if 0% telegram is received the valve is closed. Continuous PWM (1-byte): the telegram received via "Actuating Value" object is used to generate pulse-width-modulated switch signal at the valve output according to the following set parameters.
PWM period	1…**15**…60 **min**	This parameter specifies the switching frequency of the pulse-width-modulated output signal of a valve output.
Actuating value limitation	**Disabled** Enabled	This parameter is used to enable or disable a limitation on the value received from "Actuating Value" object.
Minimum actuating value	**0**…100	Lowest permissible actuating value.
Maximum actuating value	0…**100**	Highest permissible actuating value.
When received actuating value violates min/max values	**Use min / max values** If < min use 0% / if > max use 100% If 0% use 0% else use min / max values	Use min / max values: Restrict values to the set maximum and minimum actuating value. If < min use 0% / if > max use 100%: Actuate channel with 0% when the received value is lower than the minimum actuating value, actuate channel with 100% when the received value is higher than the maximum actuating value. If 0% use 0% else use min / max values: If the received actuating value is 0%, accept this value and close the valve. Other values are restricted according to the configured minimum and maximum actuating values.
Time for closing the valve	**00:00**…15:00 mm:ss	This parameter is available if "Supported fan coil system" parameter is set to "Cooling and heating". In addition to the "Delay between heating and cooling" time, the output will wait this amount of time before switching to another mode.
Delay between heating and cooling	**00:00**…30:00 mm:ss	Delay when changing from heating to cooling or from cooling to heating after the valve of the current mode is completely closed. The valve of the new mode can only be opened after this time has elapsed.

Table 67

8.6.5 Fan Settings Parameters

Name	Values	Description
Number of fan levels	1 level 2 levels **3 levels** 4 levels 5 levels 6 levels	This parameter shows the set parameter in “Enable Outputs” page that specifies the number of levels (steps) of the controlled fan in the fan coil system.
Fan control type	**Individually** Hierarchically	In standard commercially available fan coil units, there are two ways to control the fan levels. Individually: only one level output can be switched on, all other levels must switch off. Hierarchically: the fan level outputs switch one after another, lower levels remain switched on, higher levels switched off. The operation of these ways are shown in Table 69 and Table 70.
Fan level 1 threshold	**1**…100 %	This parameter sets the lower limit of fan level 1.
Fan level 2 threshold	1…**20**…100 %	This parameter is only available if the number of fan levels of the fan coil output is set to “2 levels” or higher. This parameter sets the lower limit of fan level 2.
Fan level 3 threshold	1…**40**…100 %	This parameter is only available if the number of fan levels of the fan coil output is set to “3 levels” or higher. This parameter sets the lower limit of fan level 3.
Fan level 4 threshold	1…**60**…100 %	This parameter is only available if the number of fan levels of the fan coil output is set to “4 levels” or higher. This parameter sets the lower limit of fan level 4.
Fan level 5 threshold	1…**80**…100 %	This parameter is only available if the number of fan levels of the fan coil output is set to “5 levels” or higher. This parameter sets the lower limit of fan level 5.
Fan level 6 threshold	1…**95**…100 %	This parameter is only available if the number of fan levels of the fan coil output is set to “6 levels”. This parameter sets the lower limit of fan level 6.
Hysteresis	**0**…30 %	A fan level is only deactivated when the active actuating value reaches or falls below the threshold of the level less the hysteresis. In the same way, a fan level is only activated when the active actuating value reaches or passes the threshold of the fan level plus the hysteresis.

Table 68

Active Fan Level	Fan 1 Output	Fan 2 Output	Fan 3 Output
None (switched off)	OFF	OFF	OFF
Fan Level 1	**ON**	OFF	OFF
Fan Level 2	OFF	**ON**	OFF
Fan Level 3	OFF	OFF	**ON**

Table 69

Active Fan Level	Fan 1 Output	Fan 2 Output	Fan 3 Output
None (switched off)	OFF	OFF	OFF
Fan Level 1	**ON**	OFF	OFF
Fan Level 2	**ON**	**ON**	OFF
Fan Level 3	**ON**	**ON**	**ON**

Table 70

8.6.6 **Fan Behavior Parameters**

Name	Values	Description
Minimum time to stay within a fan step	**00:00**…15:00 mm:ss	This parameter specifies how short the time intervals for switching the fan levels can be, i.e. the limit to how quickly the fan speed can be varied.
Pause time	**0**…30 x100 ms	This parameter determines how long the fan will remain in off state before a fan level is switched on during the fan level changing process.
Fan switch on delay	**00:00**…15:00 mm:ss	When the fan is switched off and a new actuating value higher than level 1 threshold is recieved, the fan will switch on only after this time. This can be used to prevent blowing unheated or uncoolded air to the room.
Fan switch off delay	**00:00**…15:00 mm:ss	When the fan is switched on and a new actuating value lower than level 1 threshold is received, the fan will switch off only after this time. This can be used to prevent the freezing of the cooling register at the end of a cooling process, or overheating the heating register at the end of a heating process.

Table 71

8.6.7 **Manual Fan Control Page**

Name	Values	Description
Fan auto-manual object value	On = auto, off = manual **Off = auto, on = manual**	This parameter determines the values that will active auto and manual modes when it received on “Fan Auto-Manual Switchover”.

Name	Values	Description
Fan level change over in manual control via	1-bit objects **1-byte object (0-255)** 1-byte object (0%-100%)	This parameter determines the datapoint type of the manual fan level object(s) used to switch between fan levels in manual mode.

Table 72

8.6.8 **Feedback Page**

Name	Values	Description
Fan coil feedback object indicates	Operating mode is active / inactive Valve is open / closed	Operating mode is active / inactive: mode feedback object indicates if its mode is the current mode and the operated actuating value is higher than 0%. Valve is open / closed: valve feedback object indicates if its valve is closed or open (Closed = 0, Open = 1).
Fan level feedback via	1-bit objects **1-byte object (0-255)** 1-byte object (0%-100%)	This parameter determines the datapoint type of the fan level feedback object(s) used to report the current fan level.
Send feedback telegrams	Do not send, update only **Send on change**	"Don't send, update only": feedback objects values are updated when fan level and mode/valve state are changed but not sent to the bus. The user can read the object values to get the current fan level or mode/valve state. On change: The new fan level and mode/valve state are sent to the bus when they change.
Send fan feedbacks cyclically	**No** Yes	This parameter enables sending the fan level cyclically to the bus.
Feedback cycle time	00:00:01 … **00:01:00** … 09:06:07	This parameter determines the cycle time between the repeated feedback telegrams.
Limit cyclically sending count	**No** Yes	"No": the feedback telegrams will be sent to the bus cyclically forever. "Yes": the feedback telegrams will be sent to the bus cyclically a certain number of times.
Feedback Cycle count	1…**10**…65535	This parameter determines how many times a feedback telegram will be sent cyclically.

Table 73

8.6.9 **Forced Mode**

Name	Values	Description
Activate forced mode with value	0 **1**	This parameter defines the telegram value that will enable forced mode when it received on "Forced Mode" object.

Name	Values	Description
Reaction at the beginning of the forced mode	No reaction **Close all valves and switch off the fan** Specific state	No reaction: The current valve position and fan level at the beginning of forced mode is adopted with no changes. Close all valves and switch off the fan: All outputs are switched off. Fan switch on and off delay times remain valid. Specific state: At the beginning of forced mode the outputs will be operated as specified in the following parameters.
Operating mode	**Do not change** Cooling Heating	This parameter is available if “Supported fan coil system” parameter is set to “Cooling and heating”. It specifies the operating mode in forced mode.
Actuating value	**Do not change** Close the valve Open the valve Operate with fixed actuating value	Do not change: The used actuating value before enabling the forced mode will be used after enabling it. Close the valve: If valve control type is switching (1-byte), the valve is closed in forced mode. If valve control type is continuous PWM (1-byte) the valve is actuated with 0% value. Open the valve: If valve control type is switching (1-byte), the valve is opened in forced mode. If valve control type is continuous PWM (1-byte) the valve is actuated with 100% value. Operate with fixed actuating value: This parameter should be selected only when “Type of valve control” parameter is set to “continuous PWM (1-byte)”. If selected, the valve will be operated with actuating value specified in a following parameter.
Fan level	**Do not change** Fan is switched off Level 1 Level 2 Level 3 Level 4 Level 5 Level 6	If “Do not change” is selected, the used fan level before enabling the forced mode will be used after enabling it, else the fan will be operated with the specified level. Note: Selection of fan level is limited by the configured number of fan numbers.

Table 74

8.6.10 Protection Parameters

Name	Values	Description
Valve protection	**Disabled** Enabled	This parameter is used to enabled valve protection function.
Valve protection period	1…**7**…255 day	If a valve is not opened at least once during this period, then the actuator will execute valve protection function and open the valve for specific time.
In valve protection, valve is open for	**1**..60 min	This parameter defines how long the valve remains open during the valve protection procedure.
Fan protection	**Disabled** Enabled	This parameter is used to enabled fan protection function.

Name	Values	Description
Fan protection period	1…**7**…255 day	If the fan is not switched on at least once during this period, then the actuator will execute fan protection function and switch on the fan at the highest level for specific time.
In valve protection, valve is open for	**1**..60 min	This parameter defines how long the fan remains switched on at highest level during the fan protection procedure.

Table 75

8.6.11 Monitoring Parameters

Name	Values	Description
Monitoring time	1…**30**…255 min	A monitoring failure will be detected if no new actuating value is received on actuating value object of the current mode within this period.
Monitoring failure value	Monitoring failure = 0 **Monitoring failure = 1**	This parameter is used to select the polarity of monitoring failure. When monitoring failure is detected a telegram with the specified value will be sent on “Monitoring Failure” object.
Reaction at the beginning of the monitoring failure	No reaction **Close all valves and switch off the fan** Specific state	No reaction: The current valve position and fan level at the beginning of monitoring failure is adopted with no changes. Close all valves and switch off the fan: All outputs are switched off. Fan switch on and off delay times remain valid. Specific state: At the beginning of monitoring failure the outputs will be operated as specified in the following parameters.
Operating mode	**Do not change** Cooling Heating	This parameter is available if “Supported fan coil system” parameter is set to “Cooling and heating”. It specifies the operating mode in monitoring failure.
Actuating value	**Do not change** Close the valve Open the valve Operate with fixed actuating value	Do not change: The used actuating value before detecting the monitoring failure will be used after detecting it. Close the valve: If valve control type is switching (1-byte), the valve is closed during monitoring failure. If valve control type is continuous PWM (1-byte) the valve is actuated with 0% value. Open the valve: If valve control type is switching (1-byte), the valve is opened during monitoring failure. If valve control type is continuous PWM (1-byte) the valve is actuated with 100% value. Operate with fixed actuating value: This parameter should be selected only when “Type of valve control” parameter is set to “continuous PWM (1-byte)”. If selected, the valve will be operated with actuating value specified in a following parameter.

Name	Values	Description
Fan level	**Do not change** Fan is switched off Level 1 Level 2 Level 3 Level 4 Level 5 Level 6	If “Do not change” is selected, the used fan level before detecting the monitoring failure will be used after detecting it, else the fan will be operated with the specified level. Note: Selection of fan level is limited by the configured number of fan numbers.

Table 76

8.6.12 Behaviour Parameters

Name	Values	Description
Behaviour after ETS programming	**Close all valves and switch off the fan**	This parameter indicates that after ETS programming operation all outputs will be switched off.
Response to bus failure	No reaction **Close all valves and switch off the fan** Specific state	No reaction: The valve state and fan level will not be changed in case of bus failure. Close all valves and switch off the fan: All outputs are switched off. Fan switch on and off delay times, minimum time to stay within a fan step are not valid. Specific state: In case of bus failure the outputs state will be as specified in the following parameters.
Valve state	Do not change **Close the valve** Open the valve	This parameter determines the valve state in case of bus failure. Note: Open the valve option is only available for 2-pipe systems.
Fan level	**Do not change** Fan is switched off Level 1 Level 2 Level 3 Level 4 Level 5 Level 6	If “Do not change” is selected, the used fan level before detecting the bus failure will be used after detecting it, else the fan will be switched to the specified level. Note: Selection of fan level is limited by the configured number of fan numbers.
Behaviour after bus return	No reaction **Close all valves and switch off the fan** Specific state	No reaction: The valve state and fan level will not be changed in case of bus voltage return. Close all valves and switch off the fan: All outputs are switched off. Fan switch on and off delay times, minimum time to stay within a fan step are valid. Specific state: In case of bus voltage return the outputs will be operated as specified in the following parameters.
Operating mode	**As before bus failure** Cooling Heating	This parameter is available if “Supported fan coil system” parameter is set to “Cooling and heating”. It specifies the operating mode after bus return.

Name	Values	Description
Actuating value	**As before bus failure** Close the valve Open the valve Operate with fixed actuating value	Do not change: The used actuating value before detecting the bus failure will be used after restoring it. Close the valve: If valve control type is switching (1-byte), the valve is after bus return. If valve control type is continuous PWM (1-byte) the valve is actuated with 0% value. Open the valve: If valve control type is switching (1-byte), the valve is opened after bus return. If valve control type is continuous PWM (1-byte) the valve is actuated with 100% value. Operate with fixed actuating value: This parameter should be selected only when "Type of valve control" parameter is set to "continuous PWM (1-byte)". If selected, the valve will be operated with actuating value specified in a following parameter.
Fan level	**Do not change** Fan is switched off Level 1 Level 2 Level 3 Level 4 Level 5 Level 6	If "Do not change" is selected, the used fan level before detecting the bus failure will be used after restoring it, else the fan will be operated with the specified level. Note: Selection of fan level is limited by the configured number of fan numbers.
Send read request to switchover object at startup	**No** Yes	If "Yes" is selected, the actuater will transmit a read request (ValueRead) to the group address of switchover object at startup to specify a valid mode of operation.
Delay before sending read request to switchover object	0…**10**…255 **s**	The actuator waits this time before sending read request (ValueRead) to the group address of switchover object at startup.

Table 77

8.6.13 Fan Coil Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Switchover*	1 Bit	1.100 Cooling / Heating	C		W	T	U

Table 78

This object is only available if "Supported system type" is set to "Cooling and heating". It's a 1-bit object for mode of operation switchover.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Cooling Actuating Value* *Heating Actuating Value*	1 Byte	5.001 Percentage	C		W		

Table 79

A mode actuating value object is available if the mode is supported as specified in “Supported fan coil system” parameter. The actuating value receives data from the room thermostat for the corresponding valve.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Cooling Mode Feedback Active / Inactive* *Heating Mode Feedback Active/ Inactive*	1 Bit	1.001 Switch	C	R		T	

Table 80

This object is available if “Feedback” parameter is set to “Enabled and “Fan coil feedback objects” parameter is set to “Operating mode is active / inactive”. It’s used to report the current used actuatuing value. Mode feedback object indicates if its mode is the current mode and the operated actuating value is higher than 0%.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Cooling Valve Feedback Open / Close* *Heating Valve Feedback Open / Close* *Cooling and Heating Valve Feedback Open / Close*	1 Bit	1.001 Switch	C	R		T	

Table 81

This object is available if “Feedback” parameter is set to “Enabled and “Fan coil feedback objects” parameter is set to “Valve is open close”. It’s used to report the state of the valve (Open = 1, Close = 0).

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Fan Auto-Manual Switchover*	1 Bit	1.001 Switch	C		W		

Table 82

1-bit object for activating and deactivating manual fan control when it receives a telegram according to “Fan auto-manual object value” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Manual Fan Level*	1 Byte	5.100 Fan stage	C		W		

Table 83

This object is available if “Manual fan control” parameter is set to “Enabled” and “Fan level change over in manual control via” parameter is set to “1-byte object (0-255).

It's used for direct specification of the fan level in manual fan control. The maximum value is limited by the number of configured fan levels. Values greater than the maximum fan level are interpreted by the actuator like the maximum value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Manual Fan Level*	1 Byte	5.001 Percentage	C		W		

Table 84

This object is available if "Manual fan control" parameter is set to "Enabled" and "Fan level change over in manual control via" parameter is set to "1-byte object (0%-100%).

It's used for direct specification of the fan level in manual fan control. A fan level is activated if this object receives a value higher than the related level threshold value that is specified in "Fan Settings" page.

Note: Hysteresis is not applicable in manual fan control.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Manual Fan Level x*	1 Bit	1.001 Switch	C		W		

Table 85

These objects are available if "Manual fan control" parameter is set to "Enabled" and "Fan level change over in manual control via" parameter is set to "1-bit objects".

Sending "ON" value to one of these objects while manual fan control is active activates the related fan level. Sending "OFF" value to the current active fan level object while manual fan control is active switches off the fan.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Fan Level Feedback*	1 Byte	5.100 Fan stage	C	R		T	

Table 86

This object is available if "Feedback" parameter is set to "Enabled and "Fan level feedback via" parameter is set to "1-byte object (0-255)".

It's used to report the current active fan level.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Fan Level Feedback*	1 Byte	5.100 Fan stage	C	R		T	

Table 87

This object is available if "Feedback" parameter is set to "Enabled and "Fan level feedback via" parameter is set to "1-byte object (0%-100%)".

It's used to report the current active fan level by sending its threshold (set in page "Fan Settings") to the bus.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Fan Level x Feedback*	1 Bit	1.001 Switch	C	R		T	

Table 88

These objects are available if "Feedback" parameter is set to "Enabled and "Fan level feedback via" parameter is set to "1-bit objects".

Each object reports its level state active (ON = fan level is active, OFF = Fan level is not active). If all objects' values are OFF then the fan is switched off.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Forced Mode*	1 Bit	1.001 Switch	C		W		

Table 89

This object is available if "Forced mode" parameter is set to "Enabled". It's used to enable forced mode when it receives a value as specified in "Activate forced mode with value" parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
Output x	*Monitoring Failure*	1 Bit	1.001 Switch	C	R		T	

Table 90

This object is available if "Monitoring" parameter is set to "Enabled". It's used to report the monitoring failure state. When monitoring failure is detected a telegram with value specified in "Monitoring failure value" parameter will be sent on this object.

9 Auxiliary Functions

KNX Mix Actuators have 24 independent auxiliary functions. The user can choose their types from “Auxiliary Functions (v2.0)” page. Many complex configurations and creative scenarios can be done with these functions

There are 14 types of auxiliary functions:

1. Converter
2. Filter
3. General counter
4. Logic gate
5. Max/Min/Average value calculator
6. Monitor
7. Presence detector controller
8. Scene actuator
9. Send after delay
10. Send after reset
11. Send cyclically
12. Sequencer
13. Staircase controller
14. Working time counter

Each type has its own objects and parameters and all types have lock feature.

9.1 Converter

The converter function is used to convert data point types and/or telegram values.
There are 11 converter types:
User customized: The user specifies the input and the output object types, the comparison statement (greater than, equal to, between, etc…) for the input value, and the output values for the comparison result.
8 x 1-bit => 1 x 1-byte: Combines 8 1-bit objects into one 1-byte object
1 x 1-byte => 8 x 1-bit: separates a 1-byte object to 8 1-bit objects
2 x 1-byte => 1 x 2-byte: Combines 2 1-byte objects into one 2-byte object
1 x 2-byte => 2 x 1-byte: separates a 2-byte object to 2 1-byte objects
4 x 1-byte => 1 x 4-byte: Combines 4 1-byte objects into one 4-byte object
1 x 4-byte => 4 x 1-byte: separates a 4-byte object to 4 1-byte objects
2 x2-byte => 1 x 4-byte: Combines 2 2-byte objects into one 4-byte object
1 x 4-byte => 2 x 2-byte: separates a 4-byte object to 2 2-byte objects
1 x 2-byte float => 1 x 4-byte float: Converts DPT9 float values to IEEE 754 float values.
1 x 4-byte float => 1 x 2-byte float: Converts IEEE 754 float values to DPT9 float values.

For all converter types, the converter can be bidirectional (converts the telegrams in two directions).

Warning: because of the 2-byte float values encoding method, converting 4-byte float values to 2-byte float values becomes less accurate with large values (larger than 100.0).

9.1.1 Converter Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Converter type	**User customized** 8 x 1-bit => 1 x1-byte 1 x 1-byte => 8 x 1-bit 2 x 1-byte => 1 x 2-byte 1 x 2-byte => 2 x 1-byte 4 x 1-byte => 1 x 4-byte 1 x 4-byte => 4 x 1-byte 2 x2-byte => 1 x 4-byte 1 x 4-byte => 2 x 2-byte 1 x 2-byte float => 1 x 4-byte float 1 x 4-byte float => 1 x 2-byte float	This parameter defines the type of the converter. In user customized converters, the DPT of the input and the output objects, the converting conditions and the output values can be specified by the user. This type is used to convert KNX telegrams to another types and values. The other converter types are used to convert telegrams to another types only. Its values are not changed.
Bidirectional	**No** Yes	The bidirectional converter converts the telegrams in two directions. From X terminal object\s to Y terminal object\s and from Y terminal object\s to X terminal object\s. If “Yes” is selected, “Send output value after delay” and “Send output value cyclically” parameters cannot be used.
User customized converter parameters		
X/Y terminal object type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter defines the DPT of the terminal object. If the converter is not bidirectional, X terminal is the input and Y terminal is the output, else both terminals can be operated as input and output.
If received X/Y terminal object value is	**Equal to** Unequal to Lower than	This parameter defines the condition of the converter. one or two additional parameters will be visible to enter the test values of the condition.

	Equal or lower than Greater than Equal or greater than Between Not between Above or below threshold	If the terminal object type is “1-bit”,”2-bit” “4-bit” “Scene number” or “HVAC mode”, only two conditions can be chosen, “Equal to” and “Unequal to”.
Then send to Y/X terminal		This parameter defines the telegram value to be sent when the condition is met.
Else	**Don’t send telegram** Send telegram	This parameter defines the behavior of the converter when the condition is not met. If “Send telegram” is selected, a parameter will be visible to define the telegram value to be sent when the condition is not met.
Send telegram when	**New telegram is received** Terminal object value changes	If “New telegram is received” is selected, the converter sends telegrams to a terminal every time it receives new telegrams from the other terminal. If “Terminal object value changes” is selected, the converter sends telegrams to a terminal only when its object value is changed.
Send output value after delay	**No** Yes	This parameter enables a delay before sending the output value.
Delay time unit	Second **Minute** Hour Day	This parameter defines the unit of the delay time before the output value is sent.
Delay time value	**1**…255	This parameter defines the value of the delay time before the output value is sent.
Send output cyclically	**No** Yes	This parameter enables sending the output value cyclically to the bus.
Cycle time unit	Second **Minute** Hour Day	This parameter defines the unit of the time period between the repeated output telegrams.
Cycle time value	**1**…255	This parameter defines the value of the time period between the repeated output telegrams.
Converter behavior after bus return	**Wait for new telegrams** Read X terminal objects Read Y terminal objects	This parameter defines the converter behavior after bus voltage return. If “Wait for new telegrams” is selected, no action will be taken. If “Read X/Y terminal objects” is selected, the converter will send read request for the terminal objects to the bus after bus return. Note: To be able to read terminal objects from the bus, the terminal should be an input terminal.
Enable lock	**No** Yes	This function enables “Lock” object that is used to lock the auxiliary function.

Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 91

9.1.2 Converter Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - X Terminal - Input* *Converter - X Terminal – Input / Output*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C C	R	W W	T T	U U

Table 92

This object is available if the converter type is "User customized".
If the converter is bidirectional X terminal is used as an input and an output for the converter, else it is used as an input only.
This object DPT is determined by "X terminal object type" parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - X Terminal – 1-Byte (LSB) - Input* *Converter - X Terminal – 1-Byte (LSB) - Output* *Converter - X Terminal – 1-Byte (LSB) – Input / Output* *Converter - X Terminal – 1-Byte (MSB) – Input* *Converter - X Terminal – 1-Byte (MSB) - Output* *Converter - X Terminal – 1-Byte (MSB) – Input / Output* *Converter - X Terminal – 1-Byte (Byte i) – Input* *Converter - X Terminal – 1-Byte (Byte i) - Output* *Converter - X Terminal – 1-Byte (Byte i) – Input / Output*	1 Byte	5.010 Counter pulses	C C C C C C C C C C	 R R R R R R	W W W W W W W W W W	T T T T T T T T T T	U U U U U U U

Table 93

These objects are available if the converter type is "2 x 1-byte => 1 x 2-byte", "1 x 2-byte => 2 x 1-byte", "4 x 1-byte => 1 x 4-byte" or "1 x 4-byte => 4 x 1-byte".
LSB objects represents the least significant byte of the terminal. MSB objects represents the most significant byte of the terminal.

If the converter type is 2 x 1-byte => 1 x 2-byte" or "4 x 1-byte => 1 x 4-byte", X terminal is used as an input terminal.
If the converter type is "1 x 2-byte => 2 x 1-byte" or "1 x 4-byte => 4 x 1-byte", X terminal is used as an output terminal.
If the converter is bidirectional, X terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - X Terminal – 2-Byte (LSB) - Input*	2 Byte	7.001 Pulses	C		W	T	U
	Converter - X Terminal – 2-Byte (LSB) - Output			C	R	W	T	
	Converter - X Terminal – 2-Byte (LSB) – Input / Output			C	R	W	T	U
	Converter - X Terminal – 2-Byte (MSB) – Input			C		W	T	U
	Converter - X Terminal – 2-Byte (MSB) - Output			C	R	W	T	
	Converter - X Terminal – 2-Byte (MSB) – Input / Output			C	R	W	T	U

Table 94

These objects are available if the converter type is "2 x 2-byte => 1 x 4-byte" or "1 x 4-byte => 2 x 2-byte". LSB objects represents the least significant bytes of the terminal. MSB objects represents the most significant bytes of the terminal.
If the converter type is "2 x 2-byte => 1 x4-byte", X terminal is used as an input terminal.
If the converter type is "1 x 4-byte => 2 x 2-byte", X terminal is used as an output terminal.
If the converter is bidirectional, X terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - X Terminal – 2-Byte (Float) - Input*	2 Byte	9.001 Temperature	C		W	T	U
	Converter - X Terminal – 2-Byte (Float) - Output			C	R	W	T	
	Converter - X Terminal – 2-Byte (Float) – Input / Output			C	R	W	T	U

Table 95

These objects are available if the converter type is "1 x 2-byte float=> 1 x 4-byte float" or "1 x 4-byte float=> 1 x 2-byte float".
If the converter type is "2 x 2-byte float=> 1 x 4-byte float", X terminal is used as an input terminal.
If the converter type is "1 x 4-byte float=> 1 x 2-byte float", X terminal is used as an output terminal.
If the converter is bidirectional, X terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - X Terminal – Bit i – Input*	1 Bit	1.001 Switch	C		W	T	U
	Converter - X Terminal – Bit I – Output			C	R	W	T	
	Converter - X Terminal – Bit i – Input / Output			C	R	W	T	U

Table 96

These objects are available if the converter type is "8 x 1-bit => 1 x1-byte" or "1 x 1-byte => 8 x 1-bit". Each object represents the i-th bit of the terminal.
If the converter type is "8 x 1-bit => 1 x1-byte", X terminal is used as an input terminal.
If the converter type is "1 x 1-byte => 8 x 1-bit", X terminal is used as an output terminal.

If the converter is bidirectional, X terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - Y Terminal - Output*	1 Bit	1.001 Switch	C		W	T	U
		1 Bit	1.008 Up / Down					
		1 Bit	1.007 Step					
		1 Bit	1.017 Trigger					
		2 Bits	2.001 Switch Control					
		4 Bits	3.007 Dimming Control					
		1 Byte	18.001 Scene Control					
		1 Byte	5.001 Percentage					
	Converter - Y Terminal – Input / Output	1 Byte	5.010 Counter pulses	C	R	W	T	U
		1 Byte	6.010 Counter pulses					
		1 Byte	20.102 HVAC mode					
		2 Bytes	7.001 Pulses					
		2 Bytes	8.001 Pulses Difference					
		2 Bytes	9.001 Temperature					
		4 Bytes	12.001 Counter Pulses (Unsigned)					
		4 Bytes	13.001 Counter Pulses (Signed)					
		4 Bytes	14.000 Acceleration					

Table 97

This object is available if the converter type is “User customized”.
If the converter is bidirectional Y terminal is used as an input and an output for the converter, else it is used as an output only.
This object DPT is determined by “Y terminal object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - Y Terminal – 1-Byte - Input*	1 Byte	5.010 Counter pulses	C		W	T	U
	Converter - Y Terminal – 1-Byte - Output			C	R	W	T	
	Converter - Y Terminal – 1-Byte – Input / Output			C	R	W	T	U

Table 98

This object is available if the converter type is “8 x 1-bit => 1 x1-byte” or “1 x 1-byte => 8 x 1-bit”.
It represents the byte object of the converter.
If the converter type is “8 x 1-bit => 1 x 1-byte”, Y terminal is used as an output terminal.
If the converter type is “1 x 1-byte => 8 x 1-bit”, Y terminal is used as an input terminal.
If the converter is bidirectional, Y terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - Y Terminal – 2-Byte - Input*	2 Bytes	7.001 Pulses	C		W	T	U
	Converter - Y Terminal – 2-Byte - Output			C	R	W	T	
	Converter - Y Terminal – 2-Byte – Input / Output			C	R	W	T	U

Table 99

This object is available if the converter type is “2 x 1-byte => 1 x 2-byte” or “1 x 2-byte => 2 x 1-byte”.
It represents the 2-byte object of the converter.

If the converter type is “2 x 1-byte => 1 x 2-byte”, Y terminal is used as an output terminal.
If the converter type is “1 x 2-byte => 2 x 1-byte”, Y terminal is used as an input terminal.
If the converter is bidirectional, Y terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - Y Terminal – 4-Byte - Input*	4 Bytes	12.001 Counter Pulses (Unsigned)	C		W	T	U
	Converter - Y Terminal – 4-Byte - Output			C	R	W	T	
	Converter - Y Terminal – 4-Byte – Input / Output			C	R	W	T	U

Table 100

This object is available if the converter type is “4 x 1-byte => 1 x 4-byte”, “1 x 4-byte => 4 x 1-byte”, “2 x 2-byte => 1 x 4-byte” or “1 x 4-byte => 2 x 2-byte”.
It represents the 4-byte object of the converter.
If the converter type is “4 x 1-byte => 1 x 4-byte” or “2 x 2-byte => 1 x 4-byte”, Y terminal is used as an output terminal.
If the converter type is “1 x 4-byte => 4 x 1-byte” or “1 x 4-byte => 2 x 2-byte”, Y terminal is used as an input terminal.
If the converter is bidirectional, Y terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - Y Terminal – 4-Byte Float- Input*	4 Bytes	14.000 Acceleration	C		W	T	U
	Converter - Y Terminal – 4-Byte Float - Output			C	R	W	T	
	Converter - Y Terminal – 4-Byte Float – Input / Output			C	R	W	T	U

Table 101

This object is available if the converter type is “1 x 2-byte float=> 1 x 4-byte float” or “1 x 4-byte float=> 1 x 2-byte float”.
It represents the 4-byte object of the terminal.
If the converter type is ““2 x 2-byte float=> 1 x 4-byte float”, Y terminal is used as an output terminal.
If the converter type is “1 x 4-byte float=> 1 x 2-byte float”, Y terminal is used as an input terminal.
If the converter is bidirectional, Y terminal is used as an input and output terminal.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Converter - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 102

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.2 Filter

Filter function controls which telegrams will be transmitted to the output when the input receives one according to the telegram value and a pre-set condition or according to the filter status (Enabled / Disabled).

9.2.1 Filter Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Filter objects type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter defines the DPT of the filter's terminals objects. If the converter is not bidirectional, X terminal is the input and Y terminal is the output, else both terminals can be operated as input and output.
Is filter bidirectional	**No** Yes	The bidirectional filter filters the telegrams in two directions. From X terminal object to Y terminal object and from Y terminal object to X terminal object. If "Yes" is selected, "Send passed values after delay" and "Send passed values cyclically" parameters cannot be used.
Enable "Filter – Enable / Disable" object	**No** Yes	"No": the filter is enabled always and passes telegrams from its input to its output according to a condition. "Yes": the filter can be enabled or disabled with an object and its behavior in both cases can be determined with parameters.
Filter status after bus return	Disabled **Enabled** Read from bus As before bus failure	This parameter determines the status of the filter after bus voltage return. If "Read from bus" is selected, the device will send a read request for *"Filter – Enable / Disable" object* of the function, if no response is received the filter will be enabled.
Behavior when filter is disabled	Block all telegrams **Pass all telegrams**	This parameter is available if "*Enable "Filter – Enable / Disable" object" parameter is set to "Yes".* This parameter determines the behavior of the filter when its disabled with *"Filter – Enable / Disable" object.*
Behavior when filter is enabled	**Block all telegrams** Pass all telegrams Pass according to condition	This parameter is available if "*Enable "Filter – Enable / Disable" object" parameter is set to "Yes".* This parameter determines the behavior of the filter when its enabled with *"Filter – Enable / Disable" object.*

Pass telegram if its value is	**Equal to** Unequal to Lower than Equal or lower than Greater than Equal or greater than Between Not between	This parameter is available if "*Enable "Filter – Enable / Disable" object" is set to "No" or if "Behavior when filter is enabled" is set to "Pass according to condition".* This parameter defines the condition of the filter. One or two additional parameters will be visible to enter the test values of the condition. If the filter receives a telegram that meet the condition, it will pass it to the output. If the terminal object type is "1-bit","2-bit" "4-bit" "Scene number" or "HVAC mode", only two conditions can be chosen, "Equal to" and "Unequal to".
Send passed telegrams after delay	**No** Yes	This parameter enables a delay before sending the passed *telegrams*.
Delay time unit	Second **Minute** Hour Day	This parameter defines the unit of the delay time before the passed *telegrams* is sent.
Delay time value	**1**…255	This parameter defines the value of the delay time before the *telegrams* value is sent.
Send passed telegrams cyclically	**No** Yes	This parameter enables sending the passed *telegrams* cyclically to the bus.
Cycle time unit	Second **Minute** Hour Day	This parameter defines the unit of the time period between the repeated telegrams.
Cycle time value	**1**…255	This parameter defines the value of the time period between the repeated telegrams.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 103

9.2.2 Filter Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Filter - X Terminal - Input* *Filter - X Terminal – Input / Output*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage	C C	R	W W	T	

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
		1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration					

Table 104

If the filter is bidirectional X terminal is used as an input and an output for the filter, else it is used as an input only.
This object DPT is determined by “Filter objects type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Filter - Y Terminal - Output* *Filter - Y Terminal – Input / Output*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C C	R R	W	T T	

Table 105

If the filter is bidirectional Y terminal is used as an input and an output for the converter, else it is used as an output only.
This object DPT is determined by “Filter objects type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Filter - Enable / Disable*	1 Bit	1.003 Enable	C	R	W	T	U

Table 106

This object is available if the “Enable “Filter – Enable / Disable” object “is set to “Yes”.
It is used to enable or disable the filter. When the filter is enabled or disabled it will work as it set in “Behavior when filter is enabled” and “Behavior when filter is disabled” parameters.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Filter - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 107

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.3 General Counter

The counter function increases/decreases the output value when it receives a valid telegram from its input object.

The input and output objects types can be selected from many different types. Counter start, limit and step values are configurable.

9.3.1 General Counter Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Counting type	Cyclic Up-Down Two directions **One direction – up** One direction – down One direction – direction changeable via object	This parameter defines the function behavior. Cyclic: The counter goes up (adds the step value to current counter value) at the beginning and when it reaches its limit it returns to the start value. Counting direction can be changed with “Counting Direction” object. For example, in a counter with start value 0, limit value 3 and step value 1, the counter value goes as below: 0-1-2-3-0-1-2… Up-Down: The counter goes up (adds the step value to current counter value) at the beginning and then it changes the direction every time it reaches its limit or start values. Counting direction can be changed with “Counting Direction” object too. For example, in a counter with start value 0, limit value 3 and step value 1, the counter value goes as below: 0-1-2-3-2-1-0-1-2- … Two directions: the user selects when the function will go up (add the step value to current counter value), and when the function will go down (subtract the step value from the current counter value). For example, on telegrams for up, off telegrams for down. One direction – up: The counter goes up (adds the step value to current counter value) always and when it reaches its limit it stops counting. For example, in a counter with start value 0, limit value 3 and step value 1, the counter value goes as below:

		0-1-2-3-3-3… One direction – down: The counter goes down (subtract the step value from current counter value) always and when it reaches its limit it stops counting. For example, in a counter with start value 3, limit value 0 and step value 1, the counter value goes as below: 3-2-1-0-0-0... One direction – direction changeable via object: The counter goes up or down according to "Counting Direction" object. It stops counting when it reaches its start or limit values.
Use counting direction object	**No** Yes	This parameter is available if "Counting type" parameter is set to "Cyclic" or "Up – Down". It enables "Counting Direction" object
Input object type	**1-bit** Scene number 1-byte unsigned value	This parameters defines the DPT of the input object.
Counter value object type	1-byte percentage 1-byte unsigned value 1-byte signed value 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value	This parameter defines the DPT of the counter's output object.
Counter value object is overwritable	**No** Yes	If "Yes" is selected, sending a value to "Counter Value" object will overwrite the current counter value.
Start value	**0%**...100% **0**…255 -128…**0**…127 **0**…65535 -32768…**0**…32767 -670760…**18**…670760 **0**…2147483647 -2147483648…**0**…2147483647	This parameter defines the counter start value.
Limit value	0%...**100%** 0…**255** -128…**127** 0…**65535** -32768…**32767** -670760…**30**…670760 0…**2147483647** -2147483648…**2147483647**	This parameter defines the counter limit value.
Step value	0%...**1%**...100% 0…**1**…255 -128…**1**…127 0…**1**…65535 -32768…1…**32767** -670760…**1**…670760 0…**1**…2147483647 -2147483648…**1**…2147483647	This parameter defines the counter step amount.

Number of input pulses for one counter step	**1**…100000	This parameter determines how many trigger is needed to count a step. For example, if this parameter value is 10 and the input type is object, the counter will move a step each time it receives 10 proper telegram value from its input object.
Send counter value	Do not send, update only **When counter value is changed** At specific interval Cyclically only When counter value is changed and cyclically At specific interval and cyclically	This parameter defines when the function will send the counter value to the bus. If "Do not send, update only" is selected, the user can send read request to "Counter Value" object to get the current counter value. If the counter value is set to be sent at specific interval, an additional parameter will be shown to enter the interval value. The counter value will be sent to the bus if its new value is divisible by the interval value. If the counter value is set to be sent cyclically, additional parameters will be shown to enter the cycle time.
Send counter value after bus return	**No** Yes	This parameter is available if "Send counter value" parameter is not set to "Do not send, update only". If "Yes" is selected, the function will send the loaded counter value at the start-up to the bus.
Enable "Reset Counter" object	**No** Yes	This parameter is used to enable "Reset counter" object that is used to reload the counter with its start value.
Reset when received telegram is	**Off** On Off or on	This parameter defines which value will reset the counter when it is received on "Reset Counter" object.
Enable "Alarm" object	**No** Yes	This parameter is used to enable "Alarm" object that is used to send alarm when counter value reaches a specific value.
Alarm is on when	**Limit value is reached** Specific value is reached	This parameter defines which value will trigger the alarm when the counter reaches it. If "Specific value is reached" is selected, an additional parameter will be shown to enter the alarm value.
After bus return	**Load start value** Load the reached value before bus failure	This parameter defines the behavior of the function after bus voltage return.
Overwrite counter value after download	No **Yes**	This parameter is available if "After bus return" parameter is set to "Load the reached value before bus failure". If "No" is selected, the reached counter value before the application download operation will be loaded after the download operation. If "Yes" is selected, the counter start value will be loaded after the download operation.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus	This parameter determines the status of the function's lock after bus voltage return.

	As before bus failure	If “Read from bus” is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 108

9.3.2 General Counter Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*General Counter - Next Step Input* *General Counter - Previous Step Input*	1 Bit 1 Byte 1 Byte	1.001 Switch 17.001 Scene Number 5.010 Counter Pulses	C		W		

Table 109

Previous Step Input object is available only if “Counting type” parameter is set to “Two directions”.
The DPT of these objects can be specified with “Object type” parameters.
These objects are used to trigger the counter to count when it receives a proper telegram according to “Next step at” and “Previous step at” parameters.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*General Counter - Counting Direction*	1 Bit	1.008 Up / Down	C		W		

Table 110

This object is available if “Counting type” parameter is set to “One direction – changeable via object” or when it is set to “Cyclic” or “Up – down” and “Use “Counting Direction” object” is set to “Yes”.
It’s used to change the counting direction.
When the direction is up, the counter adds the step value to the current counter value when it’s triggered.
When the direction is down, the counter subtracts the step value from the current counter value when it’s triggered.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*General Counter - Counter value*	1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes	5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed)	C	R	W	T	

Table 111

The value of this object represents the reached counter value. Its DPT is specified by “Counter value object type” parameter.

If “Counter value object is overwritable” parameter is set to “Yes”, writing a value to this object will overwrite the current counter value. If the written value isn’t between the start and the limit values, the counter will use the start or the limit value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*General Counter - Reset Counter*	1 Bit	1.001 Switch	C		W		

Table 112

This object is available if “Enable “Reset Counter” object” parameter is set to “Yes”.
It’s used to reload the counter with its start value when it receives a telegram as specified in “Reset when received telegram is” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*General Counter - Alarm*	1 Bit	1.005 Switch	C	R		T	

Table 113

This object is available if “Enable “Alarm” object” parameter is set to “Yes”.
It’s used to send alarm when counter value reaches a specific value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*General Counter - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 114

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.4 Logic Gate

This function acts as a logic gate with maximum 8 x 1-bit input objects, and one output object.

The output object type can be selected from 14 types.

9.4.1 Logic Gate Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Logic gate type	**AND**	AND: gives a true output only if all its inputs are 1.

	OR XOR NAND NOR XNOR One hot NOT	OR: gives a true output if one or more of its inputs are 1. XOR: gives a true output when the number of 1 inputs is odd. NAND: Its output is true if any of the inputs are 0. NOR: produces an output which is false only if all its inputs are 1. XNOR: gives a true output when the number of 1 inputs is even. One hot: gives a true output if there is only one input is 1. For example: The gate gives true with these input values: 0-1-0-0-0-0 or 0-0-0-0-0-1 or 1-0-0-0-0-0 The gate gives false with these input values: 0-0-0-0-0-0 or 0-1-1-0-0-0 or 1-1-1-1-1-1 NOT: produces an inverted version of the input at its output. It is also known as an inverter.
Number of used inputs	**2** … 8	This parameter is shown if the logic gate type is not “NOT” gate. It defines how many inputs the logic gate will have.
Number of used NOT gates	**1** … 4	This parameter is shown if the logic gate type is “NOT” gate. It defines how many NOT gate the auxiliary function will have.
Input X polarity	**Normal** Inverted	This parameter is shown if the logic gate type is not “NOT” gate. It is used to invert the input object value.
Input X value after mains voltage recovery	**0** 1 As before bus failure Read from bus Block output until new telegram is received	This parameter is shown if the logic gate type is not “NOT” gate. If “Read from bus” is selected, the function will send read request for the input object. If no response is received, the input value will be 0. If “Block output until new telegram is received” is selected, the output value will not be sent to the bus until the logic gate receives a telegram from this input.
Output object type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter is shown if the logic gate type is not “NOT” gate. This parameter defines the DPT of the output object of the logic gate, and according to its value two parameters will be shown to specify the output values for true and false.
Send output telegram when	**Input is updated** Output changes	If “Input is updated” is selected, the gate will send an output telegram every time a telegram is sent to an input object even if the output state didn’t change.

		If “Output changes” is selected, the gate will send an output telegram only when its output state changes from true to false or from false to true.
Send output value after delay	**No** Yes	This parameter is shown if the logic gate type is not “NOT” gate. It enables a delay before sending the output value.
Delay time unit	Second **Minute** Hour Day	This parameter defines the unit of the delay time before the output value is sent.
Delay time value	**1**…255	This parameter defines the value of the delay time before the output value is sent.
Send output cyclically	**No** Yes	This parameter is shown if the logic gate type is not “NOT” gate. It enables sending the output value cyclically to the bus.
Cycle time unit	Second **Minute** Hour Day	This parameter defines the unit of the time period between the repeated output telegrams.
Cycle time value	**1**…255	This parameter defines the value of the time period between the repeated output telegrams.
Enable lock	**No** Yes	This function enables “Lock” object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function’s lock after bus voltage return. If “Read from bus” is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 115

9.4.2 Logic Gate Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n *AF n*	*Logic Gate - Input x* *NOT Gate x - Input*	1 Bit	1.002 Boolean	C		W	T	U

Table 116

The input objects of the logic gate.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Logic Gate - Output*	1 Bit	1.001 Switch	C	R		T	
AF n	*Logic Gate - Output*	1 Bit	1.008 Up / Down					
AF n	*Logic Gate - Output*	1 Bit	1.007 Step					
AF n	*Logic Gate - Output*	1 Bit	1.017 Trigger					
AF n	*Logic Gate - Output*	2 Bits	2.001 Switch Control					

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Logic Gate - Output*	4 Bits	3.007 Dimming Control					
AF n	*Logic Gate - Output*	1 Byte	18.001 Scene Control					
AF n	*Logic Gate - Output*	1 Byte	5.001 Percentage					
AF n	*Logic Gate - Output*	1 Byte	5.010 Counter pulses					
AF n	*Logic Gate - Output*	1 Byte	6.010 Counter pulses					
AF n	*Logic Gate - Output*	1 Byte	20.102 HVAC mode					
AF n	*Logic Gate - Output*	2 Bytes	7.001 Pulses					
AF n	*Logic Gate - Output*	2 Bytes	8.001 Pulses Difference					
AF n	*Logic Gate - Output*	2 Bytes	9.001 Temperature					
AF n	*Logic Gate - Output*	4 Bytes	12.001 Counter Pulses (Unsigned)					
AF n	*Logic Gate - Output*	4 Bytes	13.001 Counter Pulses (Signed)					
AF n	*Logic Gate - Output*	4 Bytes	14.000 Acceleration					
AF n	*NOT Gate x- Output*	1 Bit	1.002 Boolean					

Table 117

The output object of the logic gate. Its DPT is specified by “Output object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Logic Gate - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 118

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.5 Min/Max/Average Value Calculator

The value calculator calculates the minimum, the maximum or the average value of many input object values.

9.5.1 Min/Max/Average Value Calculator Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Objects type	1-byte percentage 1-byte unsigned value 1-byte signed value 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter determines the DPT of the input objects and the output object of the calculator.
Inputs number	**2** … 8	This parameter defines how many inputs the function will have.

Output value is	**The largest value of the inputs** The smallest value of the inputs The average value of the inputs	This parameter determines the type of the calculator. Note: When "the average value of the inputs" is selected, the sum of the input values must be less than 2147483648 and bigger than -2147483648. If the sum of the input values exceeds 2147483648 the calculator may give wrong output values.
Send output when	**Input is updated** Output value is changed	If "Input is updated" is selected, the calculator will send an output telegram every time a telegram is sent to an input object even if the output value didn't change. If "Output changes" is selected, the calculator will send an output telegram only when output value changes.
Function behavior after bus return	**Use only received inputs values** Block output until all inputs are updated Send read request for all input objects	This parameter defines the behavior of the calculator after bus return. If "Block output until all inputs are updated" is selected, the calculator will send output values only if all input objects had received one telegram at least. If "Send read request for all input object" is selected, the calculator will send read request for all input objects, then it will use the received values only.
Send output value after delay	**No** Yes	This parameter enables a delay before sending the output value.
Delay time unit	Second **Minute** Hour Day	This parameter defines the unit of the delay time before the output value is sent.
Delay time value	**1**…255	This parameter defines the value of the delay time before the output value is sent.
Send output cyclically	**No** Yes	This enables sending the output value cyclically to the bus.
Cycle time unit	Second **Minute** Hour Day	This parameter defines the unit of the time period between the repeated output telegrams.
Cycle time value	**1**…255	This parameter defines the value of the time period between the repeated output telegrams.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 119

9.5.2 Min/Max/Average Value Calculator Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Value Calculator - Input x*	1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C	R	W	T	U

Table 120

The input objects of the calculator. The number of input objects is determined with “Inputs number” parameter. Its DPT is specified with “Objects type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Value Calculator - Output*	1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C	R		T	

Table 121

The output object of the calculator. Its DPT is specified with “Objects type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Value Calculator - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 122

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.6 Monitor

The monitor observes the status of received telegrams of an object. It sends an alarm telegram if the input object is not updated with value specified in the parameters within a certain period of time.

9.6.1 Monitor Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Monitor time unit	Second Minute **Hour** Day	This parameter defines the unit of the monitoring time.
Monitor time value	**1**…255	This parameter defines the value of the monitoring time.
Input object type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter determines the DPT of the input object.
Reset monitor timer at	**Any telegram value** Specific telegram value	This parameter determines that telegram value that will reset the monitoring timer when it's received. If "Specific telegram value" is selected, a parameter will be available to enter the telegram value that will reset the monitor.
Send monitor alarm telegram	Do not sent, update only **On change**	"Don't send, update only": "*Monitor Alarm*" object value is updated when the alarm status is changed but not sent to the bus. The user can read the object value manually. "On change": The alarm status is sent to the bus when it is changed.
Send monitor alarm cyclically	**No** Only when alarm is off Only when alarm is on Always	This parameter determines when to send the alarm status cyclically.
Cycle time unit	Second **Minute** Hour Day	This parameter defines the unit of the time period between the repeated alarm status telegrams.
Cycle time value	**1**…255	This parameter defines the value of the time period between the repeated alarm status telegrams.

Enable "Additional Alarm" object	**No** Yes	An additional alarm object with configurable DPT can be used to send specific telegrams when the alarm status is changed. If "Yes" is selected, many parameters will be available to let the user specify the DPT of "Additional Alarm" object and the telegram values to send when the alarm becomes on or off.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 123

9.6.2 Monitor Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Monitor - Input*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C		W		

Table 124

The input object of the monitor. Its DPT is specified with "Input object type" parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Monitor - Alarm*	1 Bit	1.005 Switch	C	R		T	

Table 125

This object value represents the alarm status of the function.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Monitor - Additional Alarm*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C	R		T	

Table 126

This object is available if “Enable “Additional Alarm” object” parameter is set to “Yes”. It is used to send specific telegrams when the alarm status is changed.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Monitor - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 127

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.7 Presence Detector Controller

Presence detector controller function is used to control the lights according to presence information that is supplied from a presence detector. The received on telegrams from the input object indicate a motion is started, and the received off telegrams indicate the motion is stopped. When the detector detects a motion the lights are switched on immediately, when it stops detecting motion for a specific time the lights are switched off. The delay between the last motion detection and switching the lights off is configurable with a parameter and an object.

9.7.1 Presence Detector Controller Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.

Output object type	**1-bit switch** 1-byte percentage	This parameter determines the DPT of the output object. If "1-bit switch" is selected, on telegrams will be sent to switch the lights on and off telegrams to switch them off. If "1-byte percentage" is selected, the user can specify the telegram values for switching the lights on and off with additional two parameters.
Delay before switching off	00:00:00…**00:01:00**…09:06:07	When the detection of a presence is finished, the controller sets a delay timer, and when its time is elapsed the controller switches the lights off. If a motion is detected while the timer is running, then the timer is restarted.
Enable "Set Delay Time" object	**No** Yes	This function is used to enable "Set Delay Time" object that is used to change the switching off delay time from an object.
Overwrite delay time value with download	No **Yes**	"No": the delay time that is used before the application download operation will be used after the download operation. "Yes": the delay time in "Delay before switching off" parameter will be used after the download operation.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 128

9.7.2 Presence Detector Controller Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Presence Detector Controller - Input*	1 Bit	1.001 Switch	C		W		

Table 129

The received on telegrams from this object indicate a motion is started, and the received off telegrams indicate the motion is stopped.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Presence Detector Controller – Output*	1 Bit 1 Byte	1.001 Switch 5.001 Percentage	C	R		T	

Table 130

This object DPT is specified with "Output object type" parameter. It is used to switch the lights on and off.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Presence Detector Controller – Set Delay Time*	2 Bytes	7.005 time (s)	C		W		

Table 131

This object is available if “Enable “Set Delay Time” object” parameter is set to “Yes”. It is used to change the switching off delay time. Maximum acceptable value is 32,767 seconds (equals to 09:06:07). The received values are saved to use them after bus voltage failures too.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Presence Detector Controller - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 132

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.8 Scene Actuator

Scene actuator function has one scene input and 8 outputs. It sends the configured output values when it receives the set scene number from its input object. The output values can be from different types.
Scene actuator values can be overridden with save feature.
A delay time can be set before sending the telegram of the first output, and another delay time between the outputs’ telegrams.

9.8.1 Scene Actuator Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Scene number for actuating	**1**…64	This parameter defines the scene number that will trigger the function to send its output telegrams.
Save output values with learn telegrams	**No** Yes	If “Yes” is selected, the output values can be overwritten with the last sent values to the output objects when a learn scene telegram is received. Note: Output values “Toggle”, “Up – Down”, “Decrease – Increase / Step up - Step down” for 1-bit objects and “Decrease – Increase” for 4-bit dimming control objects cannot be overwritten.

Stop sending telegrams if a scene with different number is called	**No** Yes	Due to the configurable time delays parameters of the scene actuator function, actuating a scene may take a long time and may not be executed immediately. If "Yes" is selected, the scene actuator will stop sending its output telegrams when it receives a different scene number than its scene number. For example: "Yes" should be selected when "Good bye" scene number is wanted to cancel "Welcome" scene number because they have opposite operations.
Overwrite output values at download	No **Yes**	If "No" is selected, the previously downloaded or learned output values will persist after the ETS download operation, and the new downloaded output values will be ignored. If "Yes" is selected, the downloaded output values will overwrite the previously downloaded or learned ones. Note: If "No" will be selected, the output types shouldn't be changed else the output values are undefined.
Enable actuating startup delay	**No** Yes	This parameter enables a delay before sending the first output value to the bus. If "Yes" is selected, two parameters will be shown to enter the unit and the value of the delay time.
Enable delay between output telegrams	**No** Yes	This parameter enables a delay between the output telegrams. If "Yes" is selected, two parameters will be shown to enter the unit and the value of the delay time.
Output x type	**Not used** 1-bit switch 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter defines the DPT of the x-th output object and according to its value additional parameter/s will be shown to specify the output values. If "Not used" is selected no telegram is sent for this output.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 133

9.8.2 Scene Actuator Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Scene Actuator - Scene Number Input*	1 Byte	18.001 Scene Control	C			T	

Table 134

Sending a scene number that matches the value of “Scene number for actuating” parameter to this object, triggers the scene actuator to start sending its output values.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Scene Actuator - Output x*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C		W	T	

Table 135

This object is available if the “Output x type” parameter is not set to “Not used”.
Its DPT is specified with “Output x type” parameter.
It sends the x-th output value when the scene actuator is triggered.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Scene actuator - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 136

This object is available if “Enable lock” parameter is set to “Yes”. It’s used to lock/unlock the auxiliary function.

9.9 Send After Delay

Send after delay function has one input and one output objects. It sends the received telegrams from its input object to its output object after a specific delay time. If a telegram is received before the delay time of a previous received telegram elapses, the previous telegram is ignored and the new telegram is sent to the output after the delay time.

9.9.1 Send After Delay Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Object type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter determines the DPT of the input and the output objects.
Delay time	00:00:00…**00:01:00**…09:06:07	This parameter determines the delay time before resending the received telegram from the input to the output.
Delay time is applied for	**All values** Specific value All values except specific value	This parameter determines which telegram values will be delayed. If “specific value” or “All values except specific value” is selected, an additional parameter will be available to enter the specific value. When the delay time is not applied for a telegram, it is redirected immediately to the output when it is received.
Enable “Set Delay Time” object	**No** Yes	This function is used to enable “Set Delay Time” object that is used to change the delay time from an object.
Overwrite delay time value with download	No **Yes**	“No”: the delay time that is used before the application download operation will be used after the download operation. “Yes”: the delay time in “Delay time” parameter will be used after the download operation.
Enable lock	**No** Yes	This function enables “Lock” object that is used to lock the auxiliary function.

Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 137

9.9.2 Presence Detector Controller Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send After Delay - Input*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C		W		

Table 138

The input object of the function. Its DPT is specified with "Object type" parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send After Delay – Output*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C			T	

Table 139

The output object of the function. Its DPT is specified with “Object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send After Delay– Set Delay Time*	2 Bytes	7.005 time (s)	C		W		

Table 140

This object is available if “Enable “Set Delay Time” object” parameter is set to “Yes”. It is used to change the delay time. Maximum acceptable value is 32,767 seconds (equals to 09:06:07). The received values are saved to use them after bus voltage failures too.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send After Delay - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 141

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.10 Send After Reset

Send after reset function can be used to:

1- Send a telegram with specific value or a read request when the device starts up.
2- Save the last sent value before an electric cut off and resend it when the bus voltage returns.
3- Send a read request or a telegram cyclically to the bus.

9.10.1 Send After Reset Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
After reset send	**Value telegram** Read request	This parameter defines what to send after reset.
Output object type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value	This parameter defines the DPT of the output object of this function.

	2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	
Output value	Parameter values are shown according to the selected object type.	This parameter is available if “After reset send” parameter is set to “Value telegram”. It defines the telegram value that will be sent after reset.
Overwrite output value when a telegram is received	**No** Yes	This parameter is available if “After reset send” parameter is set to “Value telegram”. If “Yes” is selected, receiving a telegram from the output object overwrites the output value. This feature can be used to save the last sent value before an electric cut off and resend it when the bus voltage returns.
Send output after delay	**No** Yes	This parameter enables a delay before sending the output value at startup.
Delay time unit	Second **Minute** Hour Day	This parameter defines the unit of the delay time before the output value is sent.
Delay time value	**1**…255	This parameter defines the value of the delay time before the output value is sent.
Send output cyclically	**No** Yes	This parameter enables sending the output value cyclically to the bus.
Cycle time unit	Second **Minute** Hour Day	This parameter defines the unit of the time period between the repeated output telegrams.
Cycle time value	**1**…255	This parameter defines the value of the time period between the repeated output telegrams.
Enable lock	**No** Yes	This function enables “Lock” object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function’s lock after bus voltage return. If “Read from bus” is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 142

9.10.2 Send After Reset Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send After Reset - Output*	1 Bit 1 Bit	1.001 Switch 1.008 Up / Down	C	R	W	T	U

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
		1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration					

Table 143

This object DPT is specified with “Output object type” parameter.
It sends the output value or read request after the device starts up according to the set parameters.
If “Overwrite output value when a telegram is received” parameter is set to “Yes”, receiving a telegram from this object overwrites the output value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send After Reset - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 144

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.11 Send Cyclically

Send cyclically function has one input and one output objects. It sends the received telegrams from its input object to its output object immediately then cyclically. If a telegram is received before the cycle time of a previous received telegram elapses, the previous telegram is ignored and the new telegram is sent to the output immediately then cyclically.

9.11.1 Send Cyclically Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Object type	**1-bit switch** 1-bit up/down 1-bit step	This parameter determines the DPT of the input and the output objects.

	1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	
Cycle time	00:00:00…**00:01:00**…09:06:07	This parameter determines the cycle time between the repeated sent telegrams.
Cycle time is applied for	**All values** Specific value All values except specific value	This parameter determines the telegram values that will be sent cyclically. If "specific value" or "All values except specific value" is selected, an additional parameter will be available to enter the specific value. When the cycle time is not applied for a telegram, it is redirected immediately to the output when it is received and not sent cyclically.
Enable "Set Cycle Time" object	**No** Yes	This function is used to enable "Set Cycle Time" object that is used to change the cycle time from an object.
Overwrite cycle time value with download	No **Yes**	"No": the cycle time that is used before the application download operation will be used after the download operation. "Yes": the cycle time in "Cycle time" parameter will be used after the download operation.
Limit cyclically sending count	**No** Yes	"No": the received telegrams from the input will be sent to the output cyclically forever. "Yes": the received telegrams from the input will be sent to the output cyclically a certain number of times.
Cycling count	1…**10**…65535	This parameter determines how many times a telegram will be sent cyclically.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 145

9.11.2 Send Cyclically Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send Cyclically - Input*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C		W		

Table 146

The input object of the function. Its DPT is specified with “Object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send Cyclically – Output*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C			T	

Table 147

The output object of the function. Its DPT is specified with “Object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send Cyclically – Set Cycle Time*	2 Bytes	7.005 time (s)	C		W		

Table 148

This object is available if “Enable “Set Cycle Time” object” parameter is set to “Yes”. It is used to change the cycle time. Maximum acceptable value is 32,767 seconds (equals to 09:06:07). The received values are saved to use them after bus voltage failures too.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Send Cyclically - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 149

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.12 Sequencer

The sequencer function sends the next/previous step value when it receives a valid telegram from its input object.

The number of steps is configurable. An additional 1-bit telegram can be sent with specific value for each step along side with output telegram. The output object type can be different for each step, or the same for all steps.

9.12.1 Sequencer Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Sequencing type	**Cyclic** Up-Down Two directions	This parameter defines the function behavior. Cyclic: The function goes up always and when it reaches its last step it returns to the first step. For example, in a sequencer with 3 steps the sequence goes as below: Step 1 – Step 2 –Step 3 – Step1 – Step 2… Up-Down: The function goes up at the beginning and then it changes the direction every time it reaches its limits. For example, in a sequencer with 3 steps the sequence goes as below: Step 1 – Step 2 –Step 3 – Step2 – Step 1 – Step 2 – Step 3 … Two directions: the user selects when the sequencer will go up, and when it will go down. For example, on telegrams for up and off telegrams for down.
Input object type	**1-bit** Scene number 1-byte unsigned value	This parameters defines the DPT of the input object.

Number of steps	**2**…5	This parameter defines how many step will be used in the sequencer function.
Number of output objects	**One output object for all steps** One output object for each step	This parameter defines how many object will be used for the steps. If “One output object for each step” is selected, each step can have an independent object with a different DPT.
Use additional 1-bit output object	**No** Yes	This parameter enables an additional 1-bit object that can be used to send a 1-bit telegram with a specific value for each step along side with the step output telegram. If it is enabled, further parameters will be shown to enable the user to enter the additional 1-bit object value for each step.
Output object type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter is available if “Number of output objects” parameter is set to “One output object for all steps”. It defines the DPT of the sequencer’s output object.
Step x output object type	**1-bit switch** 1-bit up/down 1-bit step 1-bit trigger 2-bit priority 4-bit dimming control Scene number 1-byte percentage 1-byte unsigned value 1-byte signed value HVAC mode 2-byte unsigned value 2-byte signed value 2-byte float 4-byte unsigned value 4-byte signed value 4-byte float	This parameter is available if “Number of output objects” parameter is set to “One output object for each step”. It defines the DPT of a step output object.
Step x output value	Parameter values are shown according to the selected object type.	This parameter defines the telegram value that will be sent through the step/s object when the step is reached.
Step x additional 1-bit object value	**Off** On	This parameter is available if “Use additional 1-bit output object” parameter is set to “Yes”. This parameter defines the telegram value that will be sent through the additional 1-bit object when the step is reached.

After bus return start from	**Step 1** Last sent step before bus failure	This parameter defines the behavior of the function after bus voltage return.
Send output value after delay	**No** Yes	This parameter enables a delay before sending the output value.
Delay time unit	Second **Minute** Hour Day	This parameter defines the unit of the delay time before the output value is sent.
Delay time value	**1**…255	This parameter defines the value of the delay time before the output value is sent.
Send output cyclically	**No** Yes	This parameter enables sending the output value cyclically to the bus.
Cycle time unit	Second **Minute** Hour Day	This parameter defines the unit of the time period between the repeated output telegrams.
Cycle time value	**1**…255	This parameter defines the value of the time period between the repeated output telegrams.
Enable lock	**No** Yes	This function enables “Lock” object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function’s lock after bus voltage return. If “Read from bus” is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 150

9.12.2 Sequencer Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Sequencer - Next Step Input* *Sequencer - Previous Step Input*	1 Bit 1 Byte 1 Byte	1.001 Switch 17.001 Scene Number 5.010 Counter Pulses	C		W		

Table 151

Previous Step Input object is available only if “Counting type” parameter is set to “Two directions”.
The DPT of these objects can be specified with “Object type” parameters.
These objects are used to trigger the sequencer to step when it receives a proper telegram according to “Next step at” and “Previous step at” parameters.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Sequencer - Output*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C	R		T	

Table 152

This object is available if “Number of output objects” parameter is set to “One output object for all steps”.
Its DPT is specified by “Output object type” parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Sequencer - Output x*	1 Bit 1 Bit 1 Bit 1 Bit 2 Bits 4 Bits 1 Byte 1 Byte 1 Byte 1 Byte 1 Byte 2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	1.001 Switch 1.008 Up / Down 1.007 Step 1.017 Trigger 2.001 Switch Control 3.007 Dimming Control 18.001 Scene Control 5.001 Percentage 5.010 Counter pulses 6.010 Counter pulses 20.102 HVAC mode 7.001 Pulses 8.001 Pulses Difference 9.001 Temperature 12.001 Counter Pulses (Unsigned) 13.001 Counter Pulses (Signed) 14.000 Acceleration	C	R		T	

Table 153

This object is available if “Number of output objects” parameter is set to “One output object for each step”.
Its DPT is specified by “Step x output object type” parameter.
It represents the output object of one step in the sequencer function.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Sequencer - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 154

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.13 Staircase Controller

Staircase controller function is used to switch the lights on when it is triggered, then switch them off automatically after a specific period. The received on telegrams from the input object triggers the function to switch the lights on. The function can give the user a warning before switching the lights off.

9.13.1 Staircase Controller Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Manual switch off with received off telegrams	**No** Yes	If “Yes” is selected, receiving off telegram from the input forces the function to switch off the lights immediately. If “No” is selected, off telegrams are ignored.
Output object type	**1-bit switch** 1-byte percentage	This parameter determines the DPT of the output object. If “1-bit switch” is selected, on telegrams will be sent to switch the lights on and off telegrams to switch them off. If “1-byte percentage” is selected, the user can specify the telegram values for switching the lights on and off with additional two parameters.
Light-on time	00:00:00…**00:02:00**…09:06:07	This parameter determines the time the lights will remain on after triggering the function.
Enable “Set Light-on” object	**No** Yes	This function is used to enable “Set Light-on Time” object that is used to change the light-on time from an object.
Staircase timer is retriggerable	**No** Yes	If “Yes” is selected, retriggering the function during the light-on time resets the timer. If “No” is selected, retriggering the function during the light-on time has no effect.
Overwrite delay time value with download	No **Yes**	“No”: the light-on time that is used before the application download operation will be used after the download operation. “Yes”: the light-on time in “Light-on time” parameter will be used after the download operation.
Switch off prewarning	**Disabled** Enabled	This parameter is used to warn the user that the lights will be switched off soon. Its behavior differs according to output object type. For 1-bit switch output object: Two additional times will be entered by the user “Prewarning off time” and “Prewarning on time”. When light-on time is elapsed, the lights are switched off for “Prewarning off time” then switched on for “Prewarning on time” then switched off permanently. If the function is triggered during “Prewarning off

		time" or "Prewarning on time" the lights are switched on and the timer is reset. For 1-byte percentage output object: One additional time "Prewarning time" and one dimming value "Prewarning value" will be entered by the user. When the light-on time is elapsed, the lights are dimmed to "Prewarning value" level for "Prewarning time" then they are dimmed to "switch off value". If the function is triggered during "Prewarning time" the lights are dimmed to "switch on value" and the timer is reset.
Enable lock	**No** Yes	This function enables "Lock" object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function's lock after bus voltage return. If "Read from bus" is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 155

9.13.2 Staircase Controller Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Staircase Controller - Input*	1 Bit	1.001 Switch	C		W		

Table 156

Receiving on telegrams from this object triggers the function, and the receiving off telegrams may or may not switch the lights off according to "Manual switch off with received off telegrams".

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Staircase Controller – Output*	1 Bit 1 Byte	1.001 Switch 5.001 Percentage	C	R		T	

Table 157

This object DPT is specified with "Output object type" parameter. It is used to switch the lights on and off.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Staircase Controller – Set Light-on Time*	2 Bytes	7.005 time (s)	C		W		

Table 158

This object is available if “Enable “Set Light-on Time” object” parameter is set to “Yes”. It is used to change the light-on time. Maximum acceptable value is 32,767 seconds (equals to 09:06:07). The received values are saved to use them after bus voltage failures too.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Staircase Controller - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

Table 159

This object is available if “Enable lock” parameter is set to “Yes”. It is used to lock/unlock the auxiliary function.

9.14 Working Time Counter

The working time counter allows to count the time that a KNX device remains on. Also, it can give an alarm when the working time reaches a specific value.

9.14.1 Working Time Counter Parameters

Name	Values	Description
Auxiliary function name		The user can give the auxiliary function a name for documentation purposes. This parameter value has no effect on the function work.
Input object value after bus return	**Off** On Read from bus As before bus failure	If “Read from bus” is selected, the function will send read request for the input object. If no response is received, the input value will be off.
Input polarity	**Normal** Inverted	This parameter is used to invert the input object value. “Normal”: The counter will count the time the input object value remains on. “Inverted”: The counter will count the time the input object value remains off.
Counting direction	**Decrement** Increment	“Decrement”: the counter will be loaded with “Counter start value” at the beginning and it will count down to 0. “Increment”: the counter will be loaded with 0 at the beginning and it will count up to “Counter limit value”.
Counter time unit	Second Minute **Hour**	This parameter determines the time unit of the counter.
Counter start value	1…**10000**…65535	This parameter is available if “Counting direction” parameter is set to “Decrement”.

		It determines the value that the timer will load it at the beginning.
Counter limit value	1…**1000**…65535	This parameter is available if “Counting direction” parameter is set to “Increment”. It determines the value that the timer will count up to it.
Counter value object type	2-byte unsigned value 4-byte unsigned value	This parameter defines the DPT of “Counter Value” object.
Counter value object is overwritable	**No** Yes	If “Yes” is selected, sending a value to “Counter Value” object will overwrite the current counter value.
Send counter value	Do not send, update only **When counter value is changed** At specific interval Cyclically only When counter value is changed and cyclically At specific interval and cyclically	This parameter defines when the function will send the counter value to the bus. If “Do not send, update only” is selected, the user can send read request to “Counter Value” object to get the current counter value. If the counter value is set to be sent at specific interval, an additional parameter will be shown to enter the interval value. The counter value will be sent to the bus if its new value is divisible by the interval value. If the counter value is set to be sent cyclically, additional parameters will be shown to enter the cycle time.
Send counter value after bus return	**No** Yes	This parameter is available if “Send counter value” parameter is not set to “Do not send, update only”. If “Yes” is selected, the function will send the loaded counter value at the start-up to the bus.
Enable “Reset Counter” object	**No** Yes	This parameter is used to enable “Reset counter” object that is used to reload the counter with its start value when “Counting direction” is “Decrement” or with 0 value when “Counting direction” is “Increment”.
Reset when received telegram is	**Off** On Off or on	This parameter defines the value that will reset the counter when it is received on “Reset Counter” object.
Enable “Alarm” object	**No** Yes	This parameter is used to enable “Alarm” object that is used to send alarm when counter value reaches a specific value.
Alarm is on when	**Counter time is elapsed** Specific value is reached	This parameter defines the value that will trigger the alarm when the counter reaches it. If “Specific value is reached” is selected, an additional parameter will be shown to enter the alarm value.
After bus return	**Reset** Load the reached value before bus failure	This parameter defines the behavior of the function after bus voltage return.
Overwrite counter value after download	No **Yes**	This parameter is available if “After bus return” parameter is set to “Load the reached value before bus failure”. If “No” is selected, the reached counter value before the application download operation will be loaded after the download operation.

		If “Yes” is selected, after the download operation the counter will be loaded with its start value when “Counting direction” is “Decrement” or with 0 value when “Counting direction” is “Increment.
Enable lock	**No** Yes	This function enables “Lock” object that is used to lock the auxiliary function.
Lock status after bus return	**Unlocked** Locked Read from bus As before bus failure	This parameter determines the status of the function’s lock after bus voltage return. If “Read from bus” is selected, the device will send a read request for the lock object of the function, if no response is received the function will be unlocked.

Table 160

9.14.2 Working Time Counter Group Objects

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Working Time Counter - Input*	1 Bit	1.001 Switch	C		W		

Table 161

The input object of working time counter function.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Working Time Counter - Counter value*	2 Bytes 2 Bytes 2 Bytes 4 Bytes 4 Bytes 4 Bytes	7.005 Time(s) 7.006 Time(m) 7.007 Time(h) 12.100 Counter Time (s) 12.101 Counter Time (min) 12.102 Counter Time (h)	C	R	W	T	

Table 162

The value of this object represents the reached counter value. Its DPT is specified by “Counter time unit” and “Counter value object type” parameters.
If “Counter value object is overwritable” parameter is set to “Yes”, writing a value to this object will overwrite the current counter value. If the written value is larger than the start or the limit value, the counter will use the start or the limit value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Working Time Counter - Reset Counter*	1 Bit	1.001 Switch	C		W		

Table 163

This object is available if “Enable “Reset Counter” object” parameter is set to “Yes”.

It's used to reload the counter with its start value if "Counting direction" is "Decrement" or with 0 value if "Counting direction" is "Increment" when it receives a telegram as specified in "Reset when received telegram is" parameter.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Working Time Counter - Alarm*	1 Bit	1.005 Switch	C	R		T	

Table 164

This object is available if "Enable "Alarm" object" parameter is set to "Yes".
It's used to send alarm when counter value reaches a specific value.

Object Name	Function	Size	Datapoint Type	Flags				
				C	R	W	T	U
AF n	*Working Time Counter - Lock*	1 Bit	1.003 Enable	C	R	W	T	U

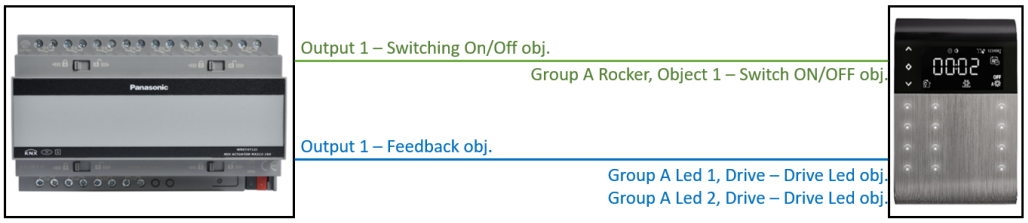
Table 165

This object is available if "Enable lock" parameter is set to "Yes". It is used to lock/unlock the auxiliary function.

10 Some examples of typical applications

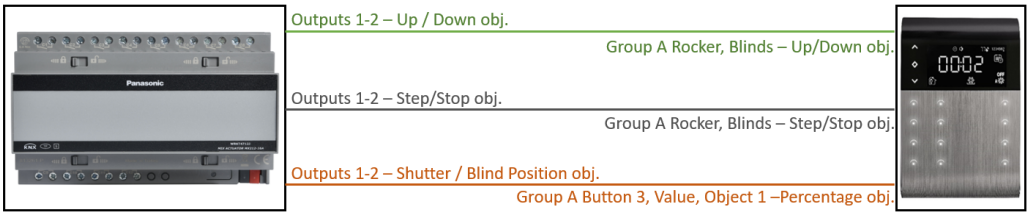
10.1 Lighting control with Mix Actuator

Lighting control can be done by using other devices with the mix actuator device. Next example shows how lighting control can be done with multi functional switch. The light is switched on/off with two buttons of multi functional switch.

Used devices	KNX Secure Mix Actuator MX312-16A-S (WRKT4812J) KNX Multi Functional Switch - MS104-D (WRKT62145FA)
Linking	
KNX Mix Actuator MX312-16A-S parameters	• Enable Outputs – Output 1 type: Switching (Lighting) • Output 1 Switching (Lighting) - Lighting Settings – Contact type: Normally open • Output 1 Switching (Lighting) - Lighting Settings – Function type: Switch on/off • Output 1 Switching (Lighting) - Lighting Settings – Feedback: Enabled • Output 1 Switching (Lighting) – Feedback – Feedback type: Normal • Output 1 Switching (Lighting) – Feedback – Send feedback telegrams: Send on change
KNX Multi Functional Switch - MS104-D parameters	• Touch buttons – Gang A: Enable • Touch buttons – Group A – Group Configuration: Rocker & Single button • Touch buttons – Group A – Rocker buttons: 1&2 • Touch buttons – Group A – Buttons – Rocker Button 1&2 – Rocker function: Switching • Touch buttons – Group A – Buttons – Rocker Button 1&2 – 1st object LEFT key action: ON • Touch buttons – Group A – Buttons – Rocker Button 1&2 – 1st object RIGHT key action: OFF • Touch buttons – Group A – Leds – Feedback led 1 – Led function: Display object value • Touch buttons – Group A – Leds – Feedback led 1 – Led behaviour: Object value 1 = Led On • Touch buttons – Group A – Leds – Feedback led 2 – Led function: Display object value • Touch buttons – Group A – Leds – Feedback led 2 – Led behaviour: Object value 0 = Led On
	The unmentioned parameters can be the default or user defined parameters

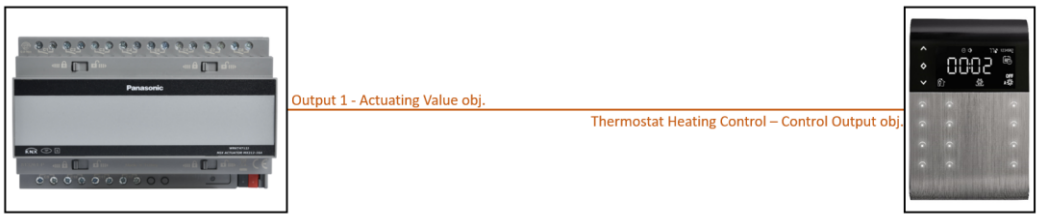
10.2 Blind control with Mix Actuator

Shutter\Blind control can be done by using other devices with the mix actuator device. In this example the operation of shutter function with multi functional switch is explained. The shutter is moved up and down in movement duration with two buttons of multi functional switch. Movement duration is the time between up and down position of shutter. The shutter takes the same time (60 s) to move between the upper position and the lower position. In addition, a previously defined shutter position can be set with another button.

Used devices	KNX Secure Mix Actuator MX312-16A-S (WRKT4812J) KNX Multi Functional Switch - MS104-D (WRKT62145FA)
Linking	
KNX Mix Actuator MX312-16A-S parameters	• Enable Outputs – Output 1 type: Shutter/Blind (AC) • Outputs 1-2 Shutter/Blind – Shutter/Blind Settings – Operation mode: Shutter (Without slats) • Outputs 1-2 Shutter/Blind – Driving Shutter / Blind – Movement duration to up and down: Same • Outputs 1-2 Shutter/Blind – Driving Shutter / Blind – Movement Duration: 60s
KNX Multi Functional Switch - MS104-D parameters	• Touch buttons – Gang A: Enable • Touch buttons – Group A – Group Configuration: Rocker & Single button • Touch buttons – Group A – Rocker buttons: 1&2 • Touch buttons – Group A – Buttons – Rocker Button 1&2 – Rocker function: Blinds • Touch buttons – Group A – Buttons – Rocker Button 1&2 – Blinds key action: Left = UP, Right = DOWN • Touch buttons – Group A – Buttons – Rocker Button 1&2 – Stop driving when: Short keystroke • Touch buttons – Group A – Buttons – Button 3 – Button function: Value • Touch buttons – Group A – Buttons – Button 3 – 1st object type: Percentage (1 byte) • Touch buttons – Group A – Buttons – Button 3 – Value: 30% • Touch buttons – Group A – Leds – Feedback led 1 –Led function: Button feedback • Touch buttons – Group A – Leds – Feedback led 1 – Led behaviour: Push/Release = Led On/Off • Touch buttons – Group A – Leds – Feedback led 2 –Led function: Button feedback • Touch buttons – Group A – Leds – Feedback led 2 – Led behaviour: Push/Release = Led On/Off • Touch buttons – Group A – Leds – Feedback led 3 –Led function: Button feedback • Touch buttons – Group A – Leds – Feedback led 3 – Led behaviour: Push/Release = Led On/Off
	The unmentioned parameters can be the default or user defined parameters

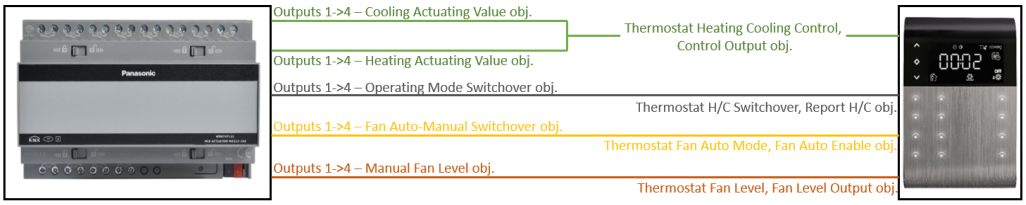
10.3 Heating control with Mix Actuator

Heating control can be done by using other devices with the mix actuator device. In this example the operation of heating function with multi functional switch is explained. Channels specified as heating in the mix actuator device can be switched with the control output value from the multi functional switch.

Used devices	KNX Secure Mix Actuator MX312-16A-S (WRKT4812J) KNX Multi Functional Switch - MS104-D (WRKT62145FA)
Linking	
KNX Mix Actuator MX312-16A-S parameters	• Enable Outputs – Output 1 type: Switching (Heating) • Output 1 Switching (Heating) - Heating Settings – Type of control: Continuous PWM (1-byte)
KNX Multi Functional Switch - MS104-D parameters	• Thermostat – Controller General - Temperature control function: Heating • Thermostat – Controller General - Controller type: Continuous
	The unmentioned parameters can be the default or user defined parameters

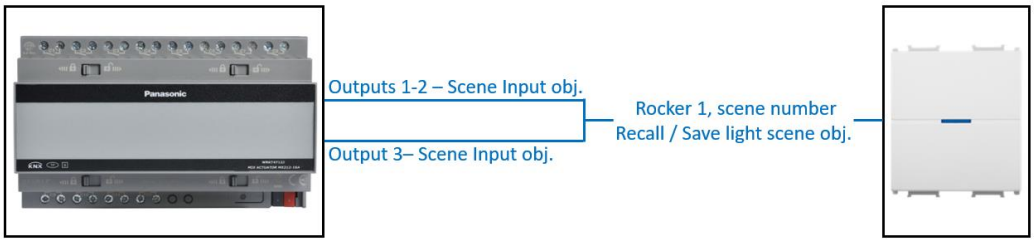
10.4 2 pipes fan coil system control with Mix Actuator

It is possible to control 2 pipes and 4 pipes fan coil system with the mix actuator device. In this example 2 pipes fan coil system control that has 3 fan levels is mentioned. Heating and cooling control can be done with the mix actuator's Fan coil (2-pipe) function.

Used devices	KNX Secure Mix Actuator MX312-16A-S (WRKT4812J) KNX Multi Functional Switch - MS104-D (WRKT62145FA)
Linking	
KNX Mix Actuator MX312-16A-S parameters	• Enable Outputs – Output 1 type: Fan coil (2-pipe) • Enable Outputs – Number of fan levels: 3 levels • Outputs 1->4 Fan Coil – Fan Coil Settings – Supported fan coil system: Cooling and Heating • Outputs 1->4 Fan Coil – Fan Coil Settings – Manual fan control: Enabled • Outputs 1->4 Fan Coil – Fan Settings – Fan level 1 threshold: 1 • Outputs 1->4 Fan Coil – Fan Settings – Fan level 2 threshold: 30 • Outputs 1->4 Fan Coil – Fan Settings – Fan level 3 threshold: 60 • Outputs 1->4 Fan Coil – Manual Fan Control – Fan auto-manual object value: Off=auto, On = manual • Outputs 1->4 Fan Coil – Manual Fan Control – Fan level changeover via: 1-byte object (0-255)
KNX Multi Functional Switch - MS104-D parameters	• Thermostat – Controller General - Temperature control function: Heating and Cooling • Thermostat – Controller General - Switchover object value = Heating = 1, Cooling = 0 • Thermostat – Controller General - HVAC system type: 1 control object (2 pipes system) • Thermostat – Controller General - Controller type: Continuous • Thermostat – Heating Control – Fan control: Enable • Thermostat – Cooling Control – Fan control: Enable • Thermostat – Fan Control – Number of fan level: 3 levels • Thermostat – Fan Control – Fan object type: 1-Byte object, configurable • Thermostat – Fan Control – Fan level off custom value: 0 • Thermostat – Fan Control – Fan level 1 custom value: 1 • Thermostat – Fan Control – Fan level 2 custom value: 2 • Thermostat – Fan Control – Fan level 3 custom value: 3 • Thermostat – Fan Control – Fan level level auto custom: No • Thermostat – Fan Control – Fan switchover auto to manual: Manual enable is 1, manual disable is 0 • Thermostat – Fan Control – Fan Heating – Fan Auto: Enable • Thermostat – Fan Control – Fan Heating: Fan level 1 threshold: 1 • Thermostat – Fan Control – Fan Heating: Fan level 2 threshold: 30 • Thermostat – Fan Control – Fan Heating: Fan level 3 threshold: 60 • Thermostat – Fan Control – Fan Cooling – Fan Auto: Enable • Thermostat – Fan Control – Fan Cooling: Fan level 1 threshold: 1 • Thermostat – Fan Control – Fan Cooling: Fan level 2 threshold: 30 • Thermostat – Fan Control – Fan Cooling: Fan level 3 threshold: 60
	The unmentioned parameters can be the default or user defined parameters

10.5 Scene control with Mix Actuator

Scene control can be done by using other devices with the mix actuator device. In the following example, scenario control in which curtain and lighting control is performed at the same time with a mix actuator and 1 gang switch is mentioned. When the up button of switch is pressed, the shutter will go up and the light will be turned off. When the bottom button is pressed, the shutter will go down and the light will be turned on.

Used devices	KNX Secure Mix Actuator MX312-16A-S (WRKT4812J) KNX Modular switch 1 gang (WRKT6121)
Linking	
KNX Mix Actuator MX312-16A-S parameters	• Enable Outputs – Output 1 type: Shutter/Blind (AC) • Enable Outputs – Output 3 type: Switching (Lighting) • Outputs 1-2 Shutter/Blind – Shutter/Blind Settings – Scenes: Enabled • Outputs 1-2 Shutter/Blind – Scenes – Scene 1 is assigned to: Scene number 1 • Outputs 1-2 Shutter/Blind – Scenes – Scene 1 – Action: Drive to top • Outputs 1-2 Shutter/Blind – Scenes – Scene 2 is assigned to: Scene number 2 • Outputs 1-2 Shutter/Blind – Scenes – Scene 2 – Action: Drive to bottom • Output 1 Switching (Lighting) - Lighting Settings – Scenes: Enabled • Output 1 Switching (Lighting) – Scenes – Scene 1 is assigned to: Scene number 1 • Output 1 Switching (Lighting) – Scenes – Scene 1 – State: Off • Output 1 Switching (Lighting) – Scenes – Scene 2 is assigned to: Scene number 2 • Output 1 Switching (Lighting) – Scenes – Scene 2 – State: On
KNX Modular switch parameters	• General – Function of Rocker 1: Scene • Rocker 1 – Scene number for Upper side: 1 • Rocker 1 – Scene number for Lower side: 2
	The unmentioned parameters can be the default or user defined parameters