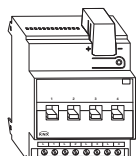


## Switch actuator REG-K/4x230/16 with current detection and manual mode

Product manual



Art. no. 647595

### Table of contents

For your safety .....	1
Getting to know the actuator.....	1
Connections, displays and operating elements .....	1
Mounting the actuator .....	1
Commissioning the actuator .....	2
Operating the actuator .....	2
What should I do if there is a problem? .....	2
Technical data .....	3
Settings in the KNX tool software (ETS) .....	4
Application overview .....	4
Application 4806/1.1 .....	
Switch logic current detection PWM .....	5

### For your safety

#### **DANGER**

##### **Risk of fatal injury from electrical current.**

The device may only be installed and connected by trained electricians. Observe the country-specific regulations as well as the valid KNX guidelines.

#### **WARNING**

Do not use the current detection function for applications relevant to safety.

#### **CAUTION**

##### **The device can be damaged.**

- Only operate the device in accordance with the specifications stated in the Technical Data.
- All devices that are installed next to the actuator must be equipped with at least basic insulation.
- Connect only pure ohmic loads to a channel with direct current (DC).

### Getting to know the switch actuator

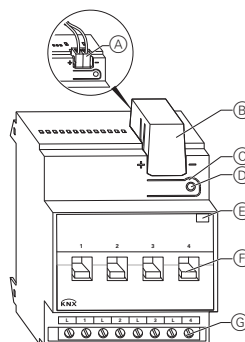
The switch actuator REG-K/4x230/16 with current detection and manual mode (referred to below as the **actuator**) can switch four loads via separate, floating make contacts.

You can also manually switch the connected loads with manual switches on the actuator without bus voltage.

The actuator has a bus coupler. It is installed on a DIN rail (DIN 60715), with the bus connection made via a bus connecting terminal. It is supplied with power from the bus voltage. A data rail is not required.

The actuator also has integrated current detection which measures the load current of each channel.

### Connections, displays and operating elements



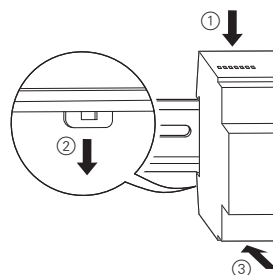
- Ⓐ Bus connecting terminal, max. 4 core pairs
- Ⓑ Cable cover
- Ⓒ Programming button
- Ⓓ Programming LED (red LED)
- Ⓔ Operational LED "RUN" (green LED)
- Ⓕ Manual switch
- Ⓖ Screw terminals

### Mounting the actuator

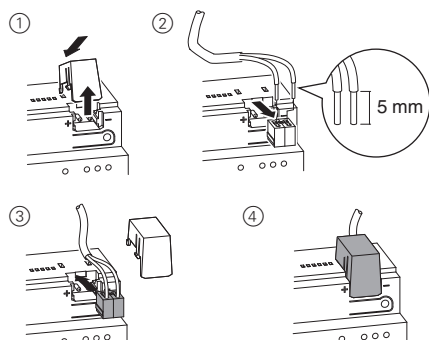
#### **CAUTION**

Strong magnetic fields can influence the current measurement. Install devices with a strong magnetic field (e.g. wound transformers such as bell transformers) at least 2 cm away from the actuator.

- ① Set the actuator onto the DIN rail.

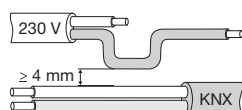


② Connect KNX.



**WARNING**  
**Risk of fatal injury from electrical current. The device can be damaged.**

Safety clearance must be guaranteed in accordance with IEC 60664-1. There must be at least 4 mm between the individual cores of the 230 V supply cable and the KNX line.



**DANGER**  
**Risk of fatal injury from electrical current.**

Voltage may be present at the outputs when the mains voltage is connected to the system. If subjected to strong vibrations during transportation, the switch contacts might change to the enabled state. After connecting the bus voltage, set the relays of the channels to the position desired simply by switching "On/Off" or by changing the manual switch to "OFF".

③ Connect the bus voltage.

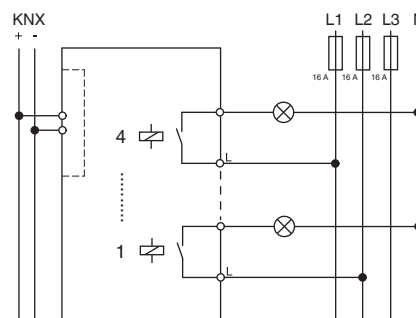
④ Wait at least 30 seconds.

⑤ Set the relays of the channels to the position desired simply by switching "On/Off" or by changing the manual switch to "OFF".

**CAUTION**  
**The actuator can be damaged.**

Protect the switch contacts with a series-connected 16 A circuit breaker.

⑥ Connect the load.



⑦ Connect the mains voltage.

You can now check the function of the connected load using the manual switch, without having to load the application from the ETS. (See the "Operating the actuator" section.)

**Commissioning the actuator**

① Press the programming button.

The programming LED lights up.

② Load the physical address and application into the device from the ETS.

The programming LED goes out.

The operational LED lights up: The application has been loaded successfully, the device is ready to be operated.

**Operating the actuator**

Connected devices are usually controlled using push-buttons or by remote control. However, you can manually switch each of the actuator's channels on and off directly at the manual switches.

**What should I do if there is a fault?**

**The green operational LED "RUN" is not lit.**

Cause	Solution
The bus voltage has failed.	Check bus voltage; only manual operation is possible.
The application was not loaded properly.	Load it again.

## Technical data

Power supply from KNX: DC 24 V, approx. 16 mA

### For alternating current (AC) per channel:

Nominal voltage: AC 230 V, 50/60 Hz  
 Nominal current: 16 A,  $\cos \varphi = 0.6$   
 Incandescent lamps: AC 230 V, max. 3600 W  
 Halogen lamps: AC 230 V, max. 2500 W  
 Fluorescent lamps: AC 230 V, max. 2500 VA, with parallel compensation  
 Capacitive load: AC 230 V, 16 A, max. 200  $\mu$ F  
 Motor load: AC 230 V, max. 1000 W  
 Switching frequency: max. 10x per minute at nominal load  
 Fuse: one 16 A circuit breaker connected upstream per channel  
 Current detection (load current):  
 Detection range (sine effective value): 0.1 A to 16 A  
 Sensing accuracy: +/- 8% from the existing current value (sine) and +/- 100 mA  
 Frequency: 50/60 Hz  
 Display: 100 mA  
 Sensing speed ( $\tau$ ): 200 ms

### For direct current (DC) per channel:

Nominal voltage: DC 12-24 V +10%, 0.1-16 A  
 Nominal current: 16 A  
 Switching frequency: max. 10x per minute at nominal load  
 Fuse: one circuit breaker capable of operating with direct current per channel, connected upstream  
 Current detection (load current):  
 Detection range: 0.1 A to 16 A  
**CAUTION:** Connect only pure ohmic loads to a channel with direct current (DC).  
 Sensing accuracy: +/- 8% from the existing current value (sine) and +/- 100 mA  
 Display: 100 mA  
 Sensing speed ( $\tau$ ): 200 ms

Ambient temperature


Operation: -5 °C to 45 °C  
 Environment: Can be used at up to 2000 m above mean sea level (MSL)  
 Max. humidity: 93% relative humidity, no moisture condensation

Operating elements: 1 programming button  
 1 manual switch per channel  
 Display elements: 1 red LED: programming check  
 1 green LED: ready for operation, "RUN"  
 KNX connection: Two 1 mm pins for bus connecting terminal  
 Load connection: one 2-gang screw terminal per channel for max. 2.5 mm<sup>2</sup> with one conductor or max. 1.5 mm<sup>2</sup> with two conductors  
 Device width: 4 modules = approx. 70 mm

## Settings in the KNX tool software (ETS)

### Selection in the product database

Manufacturer: Merten  
 Product family: 4.4 Switch actuator 4-gang  
 Product type: 4.4.01 DIN rail-mounted REG-K  
 Program name: Switch logic current detection  
 PWM 4806/1.1  
 Media type: Twisted Pair  
 Product name: Switch actuator REG-K/4x230/16  
 with current detection and  
 manual mode  
 Order number: 647595

 To guarantee the full functionality of the applications under ETS2, the ETS2 program from version 1.2 onwards and Service Release A or higher should be used.

### Application overview

Application	Ver s.	Functions
4806 Switch logic current detection PWM	1.1	Operation as break contact or make contact
		Staircase timer function
		Delay functions
		Scenes
		Logic operation functions
		Logic functions (limit value comparisons, gate functions, etc.)
		Blocking or priority control
		Central function
		Current detection (control with threshold values; power, operation and switch counter with limit value monitoring)
		PWM control

## 4806 Switch logic current detection PWM 1.1

### Table of contents

General .....	5
Application functions .....	5
Overview plan and signal routing .....	6
Switching settings .....	7
„Make contact“ relay mode .....	7
„Break contact“ relay mode .....	8
„Blinking“ relay mode .....	8
Central function .....	10
Time functions .....	12
Scenes .....	17
Higher priority functions .....	18
Status messages/ status response .....	20
Behaviour after bus voltage failure / after bus voltage recovery / after download .....	22
Safety function .....	24
Manual mode .....	25
Current detection .....	26
Counter .....	29
Logic functions .....	30
Heating function .....	45
Overview of parameters .....	48

### ● General

This software application enables you to program switch actuators using manual mode (referred to below as **actuator**), which switch loads via independent, floating make contacts.

You can also operate the relay of the actuator directly using the switch on the front. For more information see the "Manual operation" section.

All settings described refer to ETS version 3.

Maximum number of group addresses: 254

Maximum Assignments: 255



### Caution!

If you switch back to the preset values in either ETS 3 (by clicking on "Standard"), then all the values that you have changed so far will be deleted!



**Note:** Because various functions depend on other functions, these dependant functions are only visible and selectable in the ETS when the preceding function has been enabled. If you de-select functions or parameters, group addresses that have already been connected may be removed.



**Note:** The application files (vd2 and vd3) are configured in such a way that the application loading time is considerably reduced. When you convert an ETS 2 project to ETS 3, you lose this time saving. It therefore makes sense to load the vd3 file when using ETS 3.

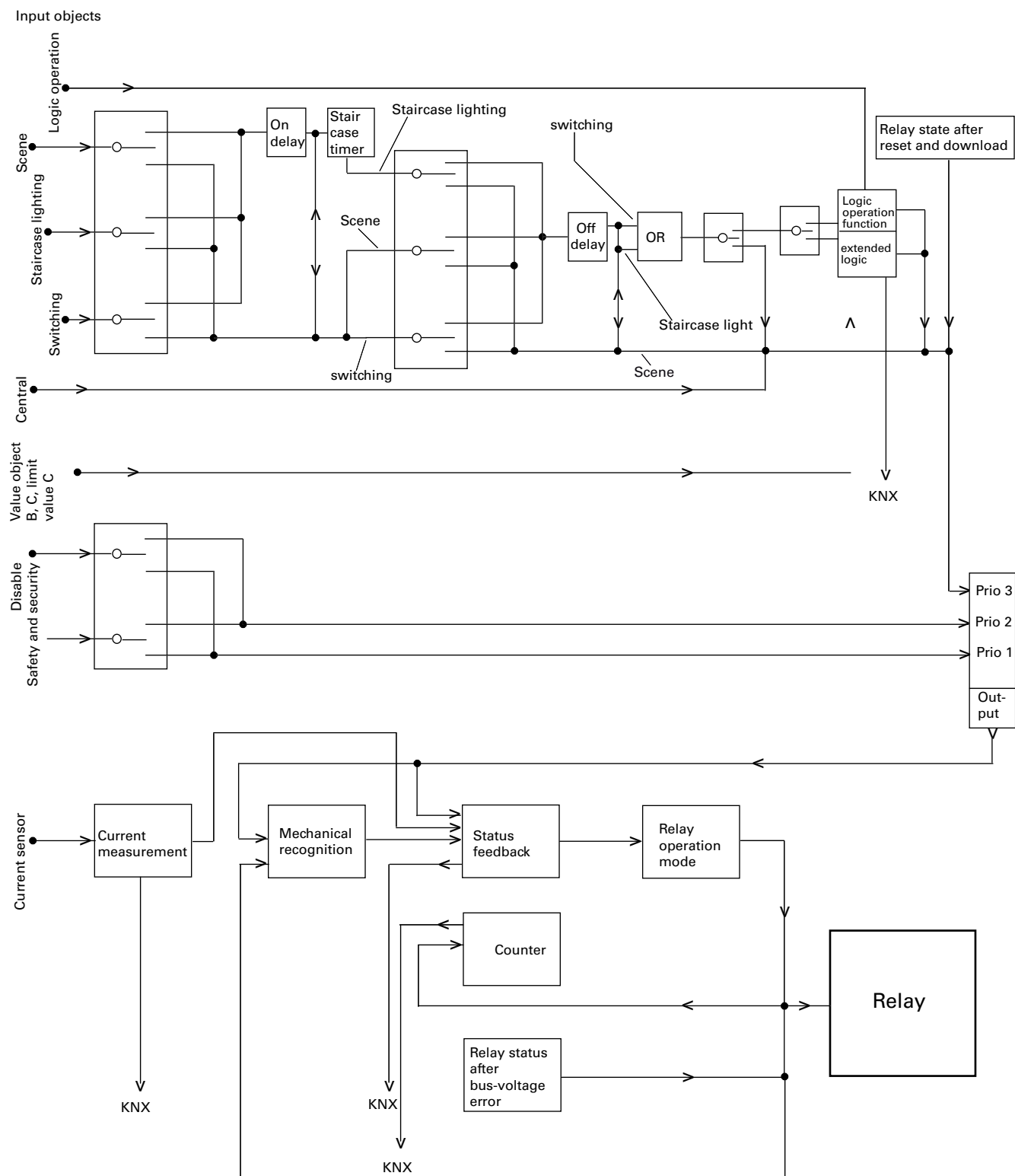
### ● Service life of the relay at different loads

DC 24 V, 16 A	20,000 switching cycles
AC 230 V, 16 A	30,000 switching cycles
AC 230 V, 1 A, $\cos \varphi = 1$	800,000 switching cycles
No load	1,000,000 switching cycles

### ● Application functions

The software application for the actuator provides you with numerous functions. The description of the functions specifies and explains the parameters that are relevant for these functions. You will find an overview of all the ETS application parameters for the actuator in the respective "Parameter overview" section.

## Overview plan and signal routing



## ● Explanations of the overview plan and signal routing

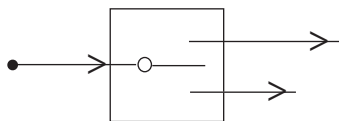


### Note:

The overview plan is read similar to a normal current circuit diagram. The KNX input objects are given on the left side. The signal flow is from left to right. The relay of the relevant switch channel is at the output. This means that the overview plan is viewed separately for each switch channel.

### Selection switch:

The following symbol represents a "selection switch":

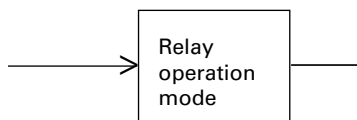


Depending on the parameterisation, this "selection switch" influences the further signal flow of the telegram within the current sensor actuator in the ETS software.

The "selection switch" functions like a switch.

### Processing

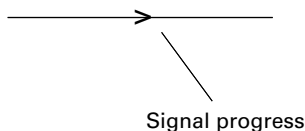
The following symbol represents a processing module:



The relay operation mode has been selected in this example. Detailed information about individual processing modules has been given below.

### Signal routing

The following symbol indicates the direction of signal (telegram) routing.



## Switching settings

To switch electronic loads on and off, you can set the output channels of the actuator to the "Switching" operating mode. You can select the operating mode for each channel in the "Channel configuration" tab:

### Parameter

Channel configuration	
Parameter	Setting
Device selection	<b>2 channels</b>
	4 channels
	8 channels
	12 channels
Channel X operation mode	<b>Switching</b>
	Heating

The descriptions below are for setting "Channel 1" and "Channel 2", but they apply to all channels.

If you select the operating mode "Switch" for two output channels, these output contacts are enabled. To activate these two outputs, the "Switch object - channel 1" and "Switch object - channel 2" communication objects (1 bit) appear.

You can use the "Switch channel 1" and "Switch channel 2" tabs to perform individual settings for each channel.

You can define the operation mode of the relay using the "Relay mode" parameter in the "Switch channel 1" and "Switch channel 2" tabs. You can use the relay as a "Make contact", a "Break contact" or for the "Blinking" function.

The relay is controlled using the "Switch object - channel 1" and "Switch object - channel 2" communication objects.

Telegram values have different effects, depending on the relay operating mode set:



When toggling the relay operation mode, parameter settings and linked group addresses are changed by the ETS program. For this reason, set the relay operation mode before parameterising the device.

### ● "Make contact" relay mode

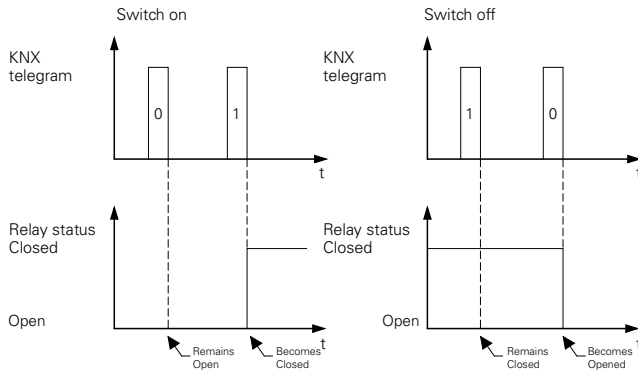
If the "Switching object" receives a telegram with the value "0", the contact is opened.

If a telegram value of "1" is received, the contact is closed.

The setting values "activated" and "not activated" are used for the different switching states of the output contacts. In the relay mode, "Make contact" means:

- Activated = pressed
- Not activated = released

## Switch (make function)



## Parameter

Channel X: Switching	
Parameter	Setting
Relay operation	Make contact

## Status feedback

Each channel can deliver status feedback, depending on the parameter settings. This always indicates the actual status of the relay. The following parameter settings are available for this:

- Pressed = 1; released = 0
- Pressed = 0; released = 1

## ● "Break contact" relay mode

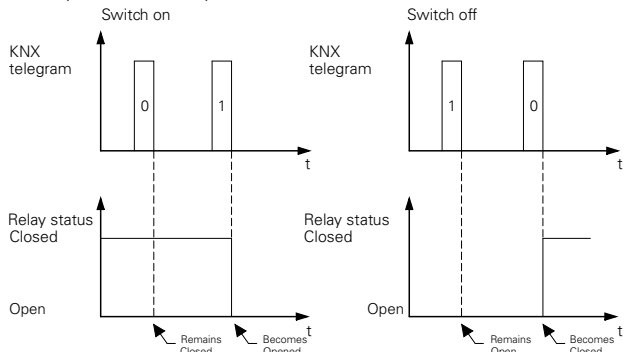
If the "Switching object" receives a telegram with the value "0", the contact is closed.

If a telegram value of "1" is received, the contact is opened.

The setting values "pressed" and "released" are used for the different switching states of the output contacts. In the relay mode, "Break contact" means:

- Activated = released
- Not activated = pressed

## Switch (break function)



## Parameter

Channel X: Switching	
Parameter	Setting
Relay operation	Break contact

## Status feedback

Each channel can deliver status feedback, depending on the parameter settings. This always indicates the actual status of the relay. The following parameter settings are available for this:

- Pressed = 1; released = 0
- Pressed = 0; released = 1

## ● "Blinking" relay mode

The "Blinking" function opens and closes the relay in alternation. You can define the blinking response for every channel using the "Blinking" tab. Define the blinking speed using a base Blinking interval \* factor. You can set whether the flash cycle begins with the relay closed or opened. You can also parameterise the ratio between the closed and opened relay during a blinking interval. You can reduce the flash impulses to a defined number in order to increase the service life of the relay. In addition, you can define the status that the relay should have after the defined number of flash impulses is reached.



### Important

Short switching times may be not parameterised under load (see the technical data for the switch output).

## Parameter

Channel X: Switching	
Parameter	Setting
Relay operation	Blinking

Channel X: Blinking (if the "Blinking" relay mode is selected)	
Parameter	Setting
Behaviour at pressed/released	<b>Blinking/relay opened</b>
	Blinking/relay closed
	Relay opened/Blinking
	Relay closed/Blinking
Basic blinking interval	<b>1 s</b> 1 min
Blinking interval factor	1-255, <b>5</b> Can be set in single steps
Ratio of closed/opened	10% / 90%
	20% / 80%
	30% / 70%
	40% / 60%
	<b>50% / 50%</b>
	60% / 40%
	70% / 30%
	80% / 20%
	90% / 10%
Blinking begins with	<b>Relay is closed</b>
	Relay is opened
Defined number of flash impulses (0 = permanent blinking)	0..100, <b>20</b>

Channel X: Blinking (if the "Blinking" relay mode is selected)	
Parameter	Setting
Status after running the defined number of flash impulses	Relay is closed
	Relay is opened

## Behaviour at pressed/released

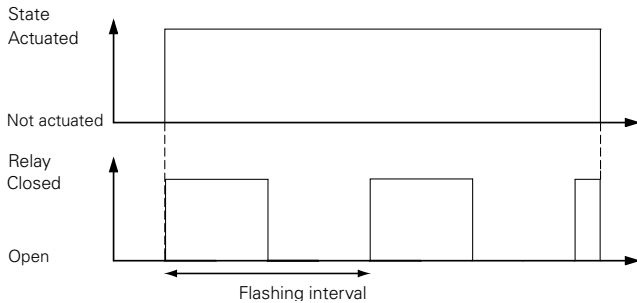
### – Blinking/relay opened

The relay starts blinking when "pressed".

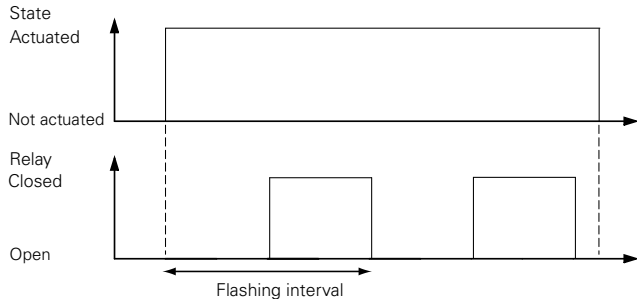
At "released", the relay stops blinking and is opened.

Define the relay status (closed or opened) at which blinking starts using the "Blinking begins with" parameter.

Blinking/relay opened  
(Flash cycle begins with relay closed)



Blinking/relay opened  
(Flash cycle begins with relay opened)



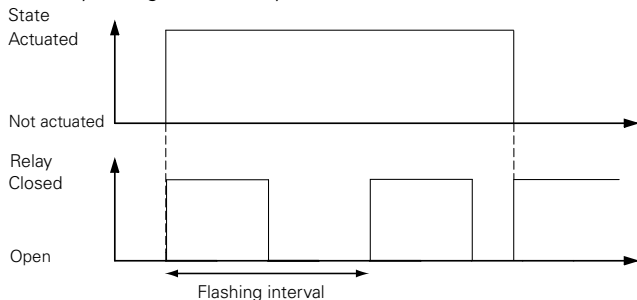
### – Blinking/relay closed

The relay starts blinking when "pressed".

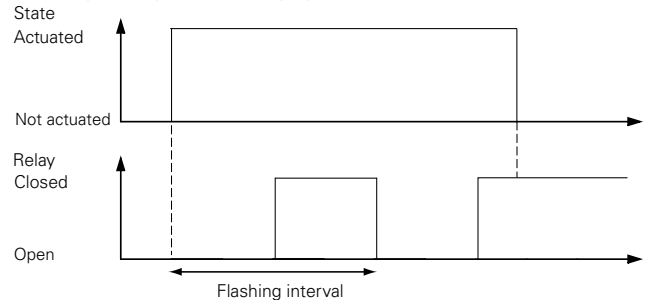
When "released", the relay stops blinking and is closed.

Define the relay status (closed or opened) at which blinking starts using the "Blinking begins with" parameter.

Blinking/relay closed  
(Flash cycle begins with relay closed)



Blinking/relay closed  
(Flash cycle begins with relay opened)



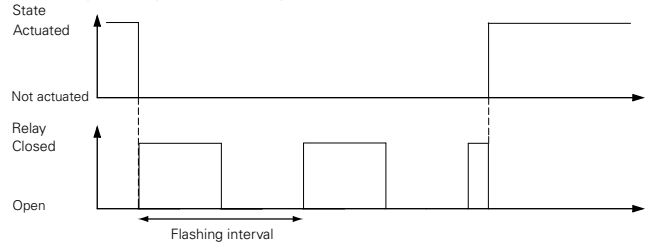
### – Relay opened/Blinking

When "pressed", the relay stops blinking and is opened.

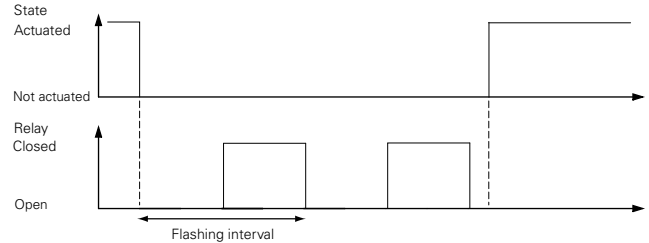
The relay starts blinking when "released".

Define the relay status (closed or opened) at which blinking starts using the "Blinking begins with" parameter.

Relay opened/Blinking  
(Flash cycle begins with relay closed)



Relay opened/Blinking  
(Flash cycle begins with relay opened)



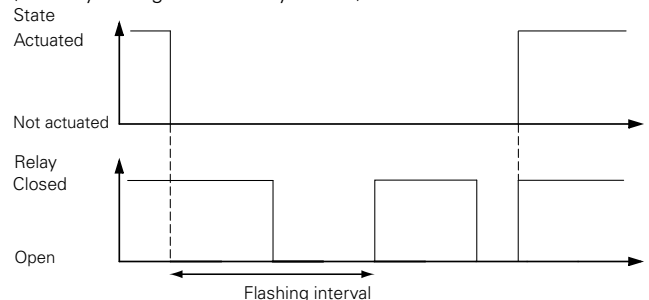
### – Relay closed/Blinking

When "pressed", the relay stops blinking and is closed.

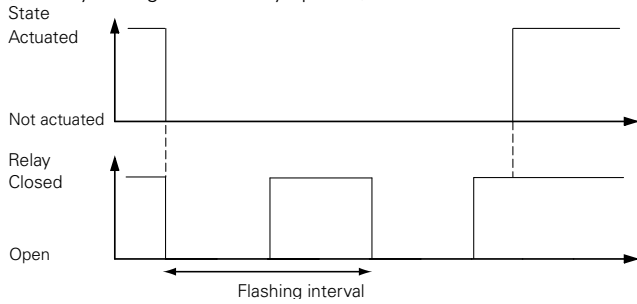
The relay starts blinking when "released".

Define the relay status (closed or opened) at which blinking starts using the "Blinking begins with" parameter.

Relay closed/Blinking  
(Flash cycle begins with relay closed)



Relay closed/Blinking  
(Flash cycle begins with relay opened)

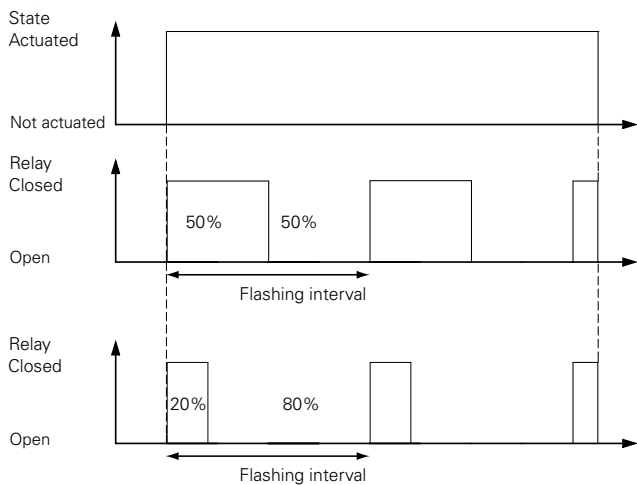


## Ratio of closed/opened

You can parameterise the percentage ratio between the closed and opened relay during a blinking interval.

Relay closed	Relay opened
10%	90%
20%	80%
30%	70%
40%	60%
50%	50%
60%	40%
70%	30%
80%	20%
90%	10%

Ratio of closed/opened  
(e.g. 1. 50%/50%; 2. 20%/80%)



## Defined number of blink impulses (0 = permanent blinking)

– 0..100

You can reduce the blink impulses to a defined number (1 to 100) in order to increase the service life of the relay. With "0", the number of blink impulses is unlimited; the relay thus blinks permanently.

## Status after running the defined number of blink impulses

- Relay closed
- Relay opened

In addition, you can define the status that the relay should have after the defined number of blink impulses is reached. Either relay closed or relay opened.

## Status feedback

Each channel can deliver status feedback, depending on the parameter settings. This always indicates the actual status of the relay. The following parameter settings are available for this:

- Pressed = 1; released = 0
- Pressed = 0; released = 1

## Note:

At the beginning of the Blinking interval, a 1-signal is sent once to the bus as status feedback. After the end of the Blinking interval, a 0-telegram is sent to the bus once.

## Communication objects

The switching commands for the output relay are received via the "Switch object" communication object.

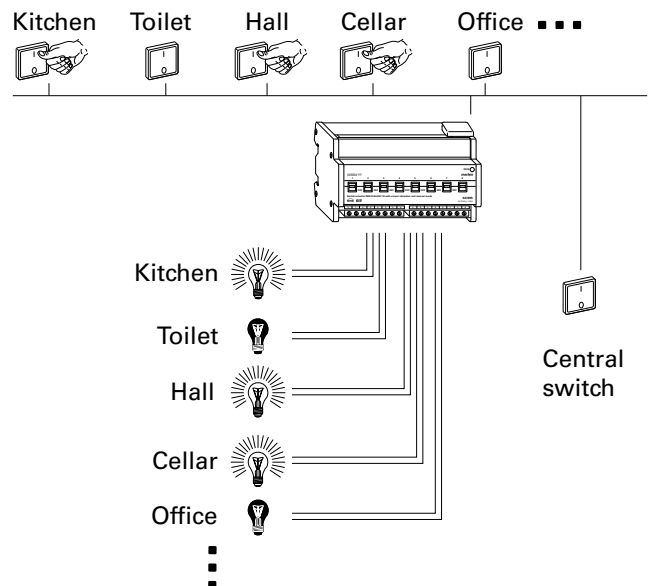
### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Switching	Switch object	1 bit	Low	CW	Receive

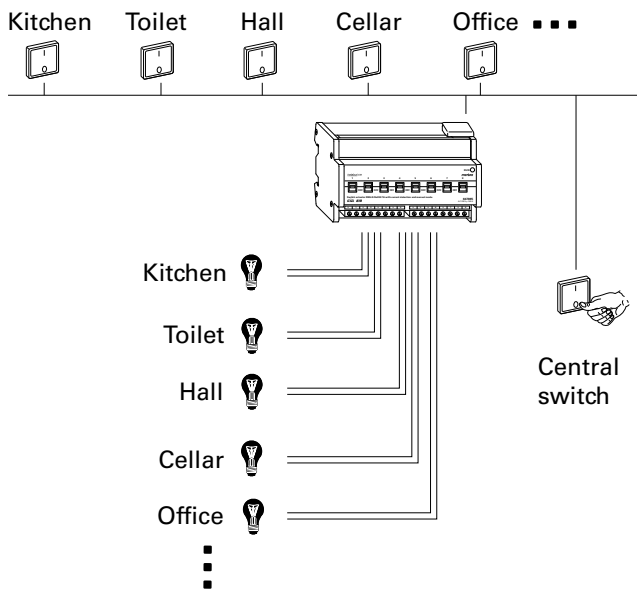
## Central function

You can use the central function to switch several output channels simultaneously with a telegram command. This function is useful, for example, when you wish to switch off all lights at the touch of a button before going to bed in the evening, or when you leave your apartment.

Decentral control without using the central function



## Central control using the central function



## Enabling the "General" central function

In order to be able to use the central function for the individual switching channels, you first have to enable the function in the "General" tab once: You can also select the "Save changes" and "Central function delay" options. You can save the changes made in the central function, for instance, using a separate push-button and the "Save changes" communication object (1-telegram saves the relay-positions in case of a 1-telegram at the central object, 0-telegram saves the relay-positions in case of a 0-telegram at the central object). You can parameterise the delay of the central function once for all channels and also set the time intervals between the central functions for every channel.

After enabling in the "General" tab, the "Central object" communication object (1 bit) is displayed. The "Save central object" communication object (1 bit) is also displayed if the "Save changes" function is selected.

### Parameter

General	
Parameter	Setting
Central function	Disabled
	Enabled
	Enabled/save changes
	Delayed/save changes

General (if "Delayed/save changes" is selected)	
Parameter	Setting
Delay of central function all channels 100 ms* factor (0-65535)	0-65535
Time between central functions per channel 100 ms* factor (0-255)	0-255, 2

## Enabling the "Channel" central function

After enabling the "Central function" parameter for every channel, you can set the function values in a new tab called "Channel X: Central function".

You can parameterise the allocation of switching states (activated, not activated) to telegram values.

### Parameter

Channel X: Switching	
Parameter	Setting
Central function	Disabled
	Enabled

Channel X: Central function (if "Enabled" is selected in the central function for the channel)	
Parameter	Setting
Overwrite runtime settings on download	Disabled
	Enabled
Central function	No reaction at "0" or "1"
	Released at "0" or "1"
	Pressed at "0" or "1"
	Pressed at "0" / released at "1"
	Released at "0" / pressed at "1"
	Released at "0" / no reaction at "1"
	Pressed at "0" / no reaction at "1"
	No reaction at "0" / released at "1"
	No reaction at "0" / pressed at "1"
	No reaction at "0" / pressed at "1"

## Overwrite runtime settings on download

– Disabled

Changes made in the central function during the operation, which you have saved using the "Save changes" communication object, are not overwritten on download.

– Enabled

All changes are overwritten on download.

## Priority

The central function has the same priority as the normal switching function. When a new object value is received via the "Central object", the effect is the same as when a new object value is received for the "Switch object" of the channel.

## Communication objects

The central control telegrams can be received via the "Central object" and "Save central object" communication objects:

Function	Object name	Type	Prio	Flags	Behaviour
Central function	Central object	1 bit	Low	CW	Receive
Central function	Save central object	1 bit	Low	CW	Receive

## Time functions

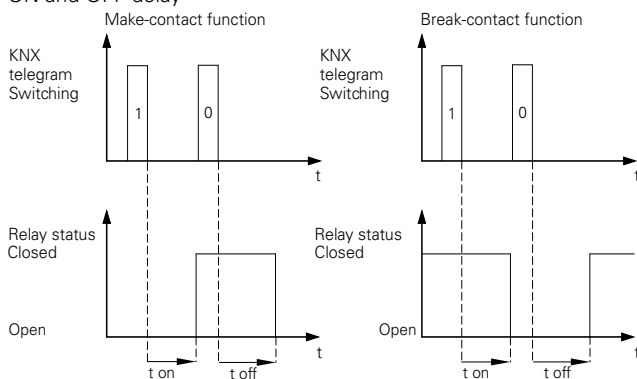
### ● ON and OFF delay

As a result of the delay function, the relay states are not altered immediately after a telegram is received, but only after the set delay time has elapsed:

- **On delay time** delays the changeover of the relay contact from the "not activated" state to the "activated" state after an object value "1" is received.
- **Off delay time** delays the changeover of the relay contact from the "activated" state to the "not activated" state after an object value "0" is received.

You can also use both functions together for one channel.

ON and OFF delay



For every channel, you can parameterise whether the delay affects the switch, staircase lighting or scene object or several objects in combination.

The delay time is made up of time base \* factor and you can parameterise it for every channel. You can define the set period as "retriggerable" or "not retriggerable" using parameters. In the case of a retriggerable on delay time, the delay time is restarted on receipt of a "1" telegram. In the case of a retriggerable Off delay time, the delay time is restarted on receipt of a "0" telegram.

You can also select "Adding" for the Off delay time. The delay time is added on receipt of the same telegram value, e.g. using a separate push-button. You can determine the maximum number of additions.

If the delay is not retriggerable, the relay is opened once the period has elapsed.

### Enabling delay times for every channel:

In order to use the delay functions with the switching channels, you must enable them separately for each switching channel. After the "On-Off delay time" function has been enabled, you can set the function values in a new tab called "Channel X: Delay times".

## Parameter

Channel X: Switching	
Parameter	Setting
On-Off delay time	Disabled
	Enabled

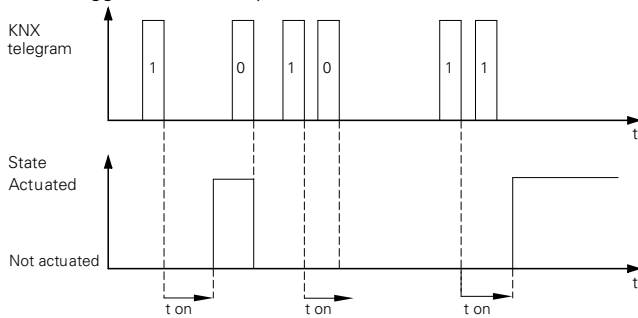
Channel X: On-Off-delay times (if "Enabled" is selected for the On-Off delay time)	
Parameter	Setting
<b>ON delay</b>	
Affects	<b>None of the objects</b> Switch object Staircase lighting time object Scene object Switch and staircase lighting object Switch and scene object Staircase lighting and scene object Switch, staircase lighting and scene object
ON delay	<b>Not retriggerable</b> Retriggerable
Time base for ON delay	100 ms <b>1 s</b> 1 min
Factor for on delay time (1-255)	1-255, <b>3</b>
<b>OFF delay</b>	
Affects	<b>None of the objects</b> Switch object Staircase lighting time object Scene object Switch and staircase lighting object Switch and scene object Staircase lighting and scene object Switch, staircase lighting and scene object
OFF delay	<b>Not retriggerable</b> Retriggerable Retriggerable and adding
Max. number of additions (if "Retriggerable and adding" Off delay time is selected)	2 3 4 <b>5</b>
Time base for OFF delay	100 ms <b>1 s</b> 1 min
Factor for Off delay time (1-255)	1-255, <b>120</b>

### Restarting On delay time/Off delay time

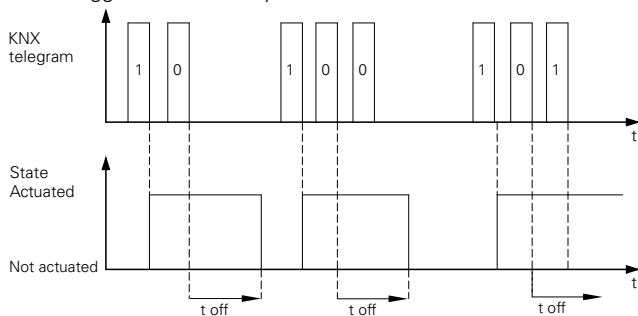
- Not retriggerable

When a telegram value is first received, the delay function is started. After the delay time has elapsed, the output relay is switched, regardless of whether further telegrams with the same value have been received during the delay time.

## Non-retriggerable ON delay



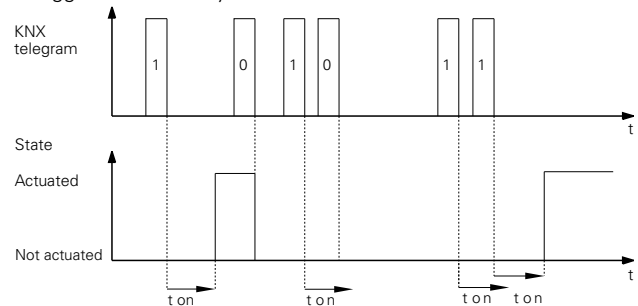
## Non-retriggerable OFF delay



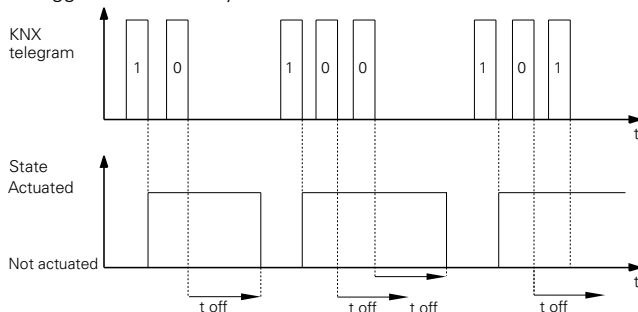
### – Retriggerable

The delay function is started while a telegram value is being received. If the "Object" receives the same telegram value again while a delay time is running, the delay time is re-started from the beginning.

## Retriggerable ON delay



## Retriggerable OFF delay



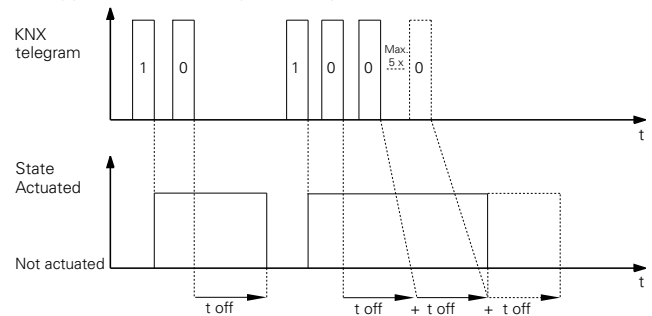
## Restarting and adding the Off delay time

### – Retriggerable and adding

The delay function is started while a telegram value is being received. If the "Object" receives the same telegram value once or several times while a delay time is running, the delay time is added. You can add the delay

time up to five times, e.g. by pressing a separate push-button multiple times.

## Retriggerable and adding Off delay time



## Interrupting the delay function

If a delay function is started when a new object value is received, and if the output channel receives a telegram with the opposite object value while the delay time is running, the delay function is interrupted. The relay is not switched:

- When the object value "0" is received, any ON delay that is running is interrupted.
- When the object value "1" is received, any OFF delay that is running is interrupted.

## Priority

If the output of the actuator is forced into a new switching position by a higher priority function while a delay is still running, the relay is switched without a delay.

## Communication objects

On and Off delay time telegrams can be received via the "Switch object" communication object:

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
On-Off delay time	Switch object	1 bit	Low	CW	Receive

## ● Staircase lighting time (staircase timer)

As the name already suggests, this function is used to switch on a load, such as the light in a staircase, via a bus telegram, and then to switch it back off after an adjustable period of time has elapsed. Switching off therefore requires no manually or automatically generated bus telegram. The actuator executes the OFF procedure independently in a time-controlled manner.

Two types of staircase lighting time are available:

- You can use the **Staircase lighting time fixed** to parameterise a fixed staircase time per channel. The fixed staircase lighting time is made up of time base \* factor. This function provides you with the "Staircase lighting time fixed" object (1 bit).
- You can determine only the time base using the **Staircase lighting time variable** function. The factor is defined using the "Staircase lighting time variable" object (2 byte integer without sign) e.g. using a push-button. You can thus specify the staircase lighting time length from different positions depending on the required situation.

You can cancel the staircase lighting time early in both of these staircase lighting functions.

You must select the "Retriggerable" or "Adding" staircase lighting time if you want to restart the staircase lighting time when it is still running or add the staircase lighting time. The staircase lighting time is then restarted or added using another "1"-telegram.

If the staircase lighting time is not retriggerable, the relay is opened immediately after the period has elapsed.

For staircase lighting time variable, you also have the option of retriggering to the higher value.

You can send a defined number of warnings that can be parameterised in terms of time, so that you are warned before the staircase lighting time elapses.

### Enabling the staircase lighting time

In order to use the staircase lighting time with the switching channels, you must select it separately for each switching channel.

After selecting the staircase lighting time fixed function, you can set the function values in a new tab called "X staircase lighting time fixed".

After selecting staircase timer variable, you can set the function values in a new tab called "X staircase lighting time variable".

### Parameter

Channel X: Switching	
Parameter	Setting
Staircase lighting function	<b>Disabled</b>
	Staircase lighting function fixed
	Staircase lighting function variable

Channel X: Staircase lighting time function fixed (if "Staircase lighting function fixed" function is selected)	
Parameter	Setting
Staircase lighting time	<b>With anticipated truncation (0-telegram)</b> Without early cancellation
Staircase lighting time is	<b>Not retriggerable</b> Retriggerable Retriggerable and adding Retrigger to the higher value (only stair. timer variable)
Max. number of additions (if "Retriggerable and adding" staircase lighting time is selected)	2 3 4 <b>5</b>
Time base staircase lighting time fixed	1 s <b>1 min</b>
Factor staircase lighting time (1-65535)	1-65535, <b>3</b>
Number of warnings before end of staircase lighting time	0 1 2 3
Warning starts at a staircase lighting rest time of (1 s * factor)	1-255, <b>30</b>

Channel X: Staircase lighting time function variable (if "Staircase lighting function variable" is selected)	
Parameter	Setting
Staircase lighting time	<b>With anticipated truncation (0-telegram)</b> Without early cancellation
Staircase lighting time is	<b>Not retriggerable</b> Retriggerable Retriggerable and adding Retrigger to the higher value (only stair. timer variable)
Max. number of additions (if "Retriggerable and adding" staircase lighting time is selected)	2 3 4 <b>5</b>
Base time for staircase lighting time variable object	1 s <b>1 min</b>
Number of warnings before end of staircase lighting time	0 1 2 3
Warning starts at a staircase lighting rest time of (1 s * factor)	1-255, <b>30</b>

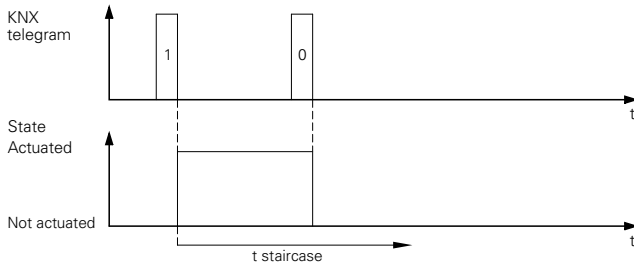
## Cancel staircase lighting time early

If the "Staircase lighting time object" receives a new telegram with the object value "0" while a staircase lighting time is running, the reaction of the switching channel will depend on your setting for the "Staircase lighting time" parameter:

- With early cancellation (0-telegram)

After the object value "0" is received, the output is switched to the "not activated" position. In this case, an active Off delay time delays the transfer of the relay states from "pressed" to "released".

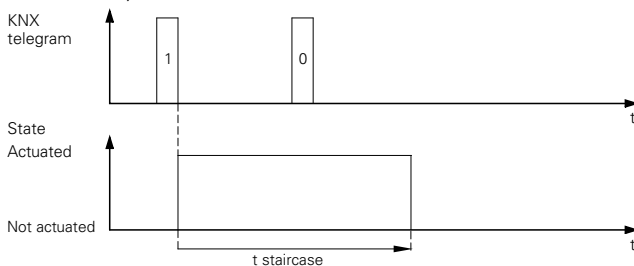
With early cancellation (0-telegram)



- Without early cancellation

A telegram with the object value "0" has no effect. The staircase lighting time set continues to run normally until the end. An OFF delay cannot be set.

Without early cancellation



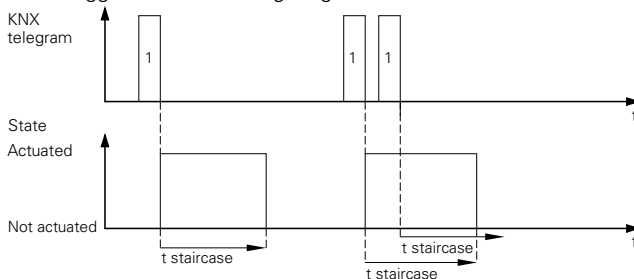
## Restart staircase lighting time

If the "Staircase lighting time object" receives a new telegram with the object value "1" while a staircase lighting time is running, the reaction of the switching channel will depend on your setting for the "Actual staircase lighting time" parameter:

- Not retriggeable

A new telegram received with the object value "1" has no effect. The function continues to run normally.

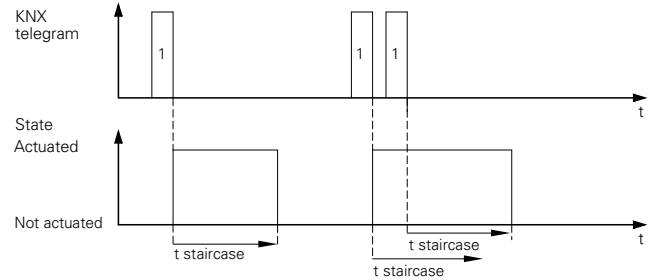
Not retriggeable staircase lighting time



- Retriggerable

After a new telegram with the object value "1" has been received, the staircase lighting time is restarted.

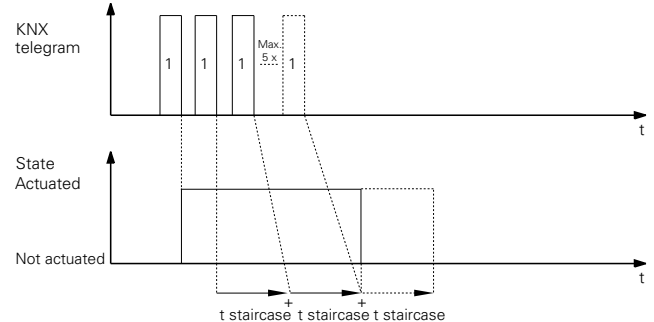
Retriggerable staircase lighting time



- Retriggerable and adding

After a new telegram or multiple telegrams with the object value "1" are received, the staircase lighting time is added to the present staircase lighting time. The number of additions can be set. You can parameterise a maximum of five staircase lighting time additions. You can add the staircase lighting time, for instance, by pressing a separate push-button multiple times.

Retriggerable and adding staircase lighting time

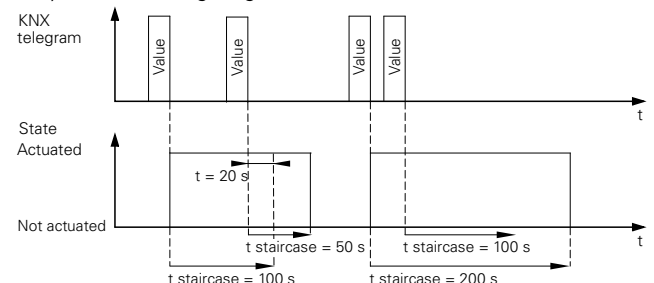


## Restarting the staircase lighting time variable with a higher value

- Retrigger to the higher value (only stair. lighting time variable)

After a new telegram has been received, the staircase lighting time is restarted with the higher value.

Retrigger to the higher value  
(only for staircase lighting time variable)



## Warning before end of staircase lighting time

With the warning, the user can be informed that the function is due to end soon when the lighting system switches off just before the staircase lighting time elapses. The user can restart (retrigger) the staircase lighting time by pressing a push-button. If the user does nothing, the function will continue to run normally.

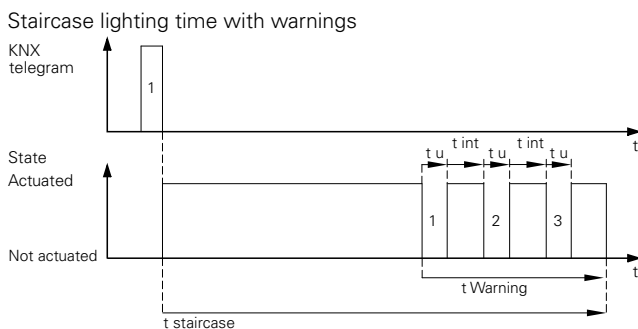
You can set the number of warnings using the "Number of warnings before end of staircase lighting time" parameter. The warning function is disabled if this parameter is set to „0“. Select one, two or three warning impulses in order to enable the warnings. The first warning starts at the set staircase lighting rest time, which you can derive from a time base (1 s) \* factor (1-255).

For each warning, the output contact for the fixed time frame of 500 ms (t<sub>u</sub>) is switched to the "not activated" state. If you have activated more than one warning, the waiting time (t<sub>del</sub>) between the warning impulses is calculated as per the following formula:

$$\text{Waiting time between warnings (t}_{\text{del}}) = \frac{\text{Staircase lighting rest time (t}_{\text{warning}}) - (\text{Number of warnings} * 500 \text{ ms (t}_{\text{u}}))}{\text{Number of warnings}}$$

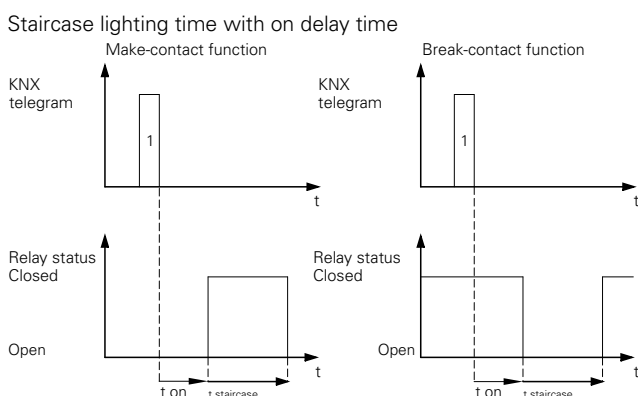


If a running staircase lighting time is interrupted by early cancellation, no warning is given.



## Staircase lighting time in combination with delay functions

The combination of a staircase lighting time with an on delay time results in the delayed start of the staircase lighting time.

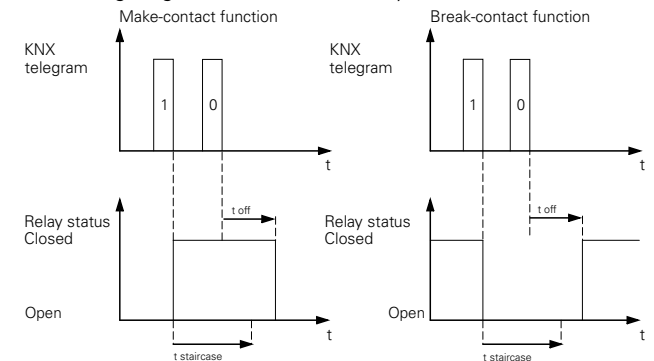


The combination of the staircase lighting time with an Off delay time depends on how you have defined the staircase lighting time:

- In case of the staircase lighting time **with anticipated truncation (0-telegram)**, the Off delay time is started when an anticipated off telegram is received on the "Staircase lighting time object". The output is switched off once the OFF delay has expired (not activated).

- For the staircase lighting time **without early cancellation**, the receipt of an Off telegram does not affect the "Staircase lighting time object". The staircase lighting time runs through to the end and switches the output relay directly into the "released" state. An OFF delay cannot be set.

### Staircase lighting function with OFF delay



For staircase lighting time with the option of early cancellation and active warning, the staircase lighting time with warning is immediately deactivated on receipt of an Off telegram. The OFF delay elapses. No warning is given.

## Priority

If the output of the actuator is brought into a new switching position by a function with a higher priority while the staircase lighting time is running, the relay is immediately switched to this position. The last switching telegram is stored and any delay periods or staircase lighting time continue to run.

## Communication objects

Telegrams for the staircase lighting time can be received via the "Staircase lighting time fixed" and "Staircase lighting time variable" communication objects:

### Per channel:

(depending on the selected staircase lighting function)

Function	Object name	Type	Prio	Flags	Behaviour
Staircase lighting time fixed	Staircase lighting time fixed	1 bit	Low	CW	Receive
Staircase lighting time variable	Staircase lighting time variable	2 byte	Low	CW	Receive

## Scenes

### ● Scene

The scene function can be used when multiple room functions need to be changed simultaneously with a push-button action or an operating command. Calling up a scene allows you, for instance, to switch on the room lighting, set the heating control to daytime operation and switch on the power supply to the socket-outlets in a room.

The scene function allows you to integrate multiple channels into one scene control. There are up to eight different scenes for each output channel. One of the 64 possible scene addresses 0 to 63 (which correspond to the telegram values 0 to 63) can be assigned to each of these scenes. You can store the switching states (activated, not activated) as scene values for every output channel.

You can parameterise a time delay for every actuator channel in order to minimise starting current when switching on a complex scene.

### Telegram format

Telegrams for the scene function have the following data format: L X D D D D D.

- L = learning bit
- X = is not used
- DDDDDD = retrieved scene address

If the learn bit in a telegram has the value "0", the relay states stored for this scene address are retrieved and set.

If the learn bit has the value "1", then the current output states are saved as new scene values for the scene address received.

Examples:

- Telegram value 57  
Binary 0011 1101  
Hexadecimal 39  
Retrieve scene address 57
- Telegram value 29  
Binary 0001 1101  
Hexadecimal 1D  
Retrieve scene address 29
- Telegram value 157  
Binary 1001 1101  
Hexadecimal 9D  
Learn scene address 29
- Telegram value 141  
Binary 10001101  
Hexadecimal 8D  
Learn scene address 13

### Retrieving and saving scenes

The "Scene object" object allows you to retrieve scene values for the output relay. After a scene telegram is received, the actuator evaluates the transmitted scene

address and switches the outputs to the stored scene values.

If the "Scene object" receives a scene telegram with learn bit "1", then the current switching state is saved as the new scene value in all the scenes that have been assigned to the scene address received.



**Note:** If a scene address is assigned to multiple scenes within a channel (incorrect parameterisation), only the scene last found with this scene address is retrieved or saved. You can avoid this by assigning different scene addresses within a channel.

### Activating the scene function

After you enable the "Scenes" parameter, the "Channel X: Scene" tab appears in the respective channel. You can specify the switching state of the output relay when a scene is retrieved in the tab. You can activate each of the eight scenes individually.

After you enable the function, the communication object "Scene object" appears. This object receives the scene telegrams.

### Parameter

Channel X: Switching	
Parameter	Setting
Scenes	Disabled
	Enabled

Channel X: Scenes (if "Enabled" is selected for scenes)	
Parameter	Setting
Overwrite scene values in actuator during download	Disabled Enabled
Time delay for the actuator channel (base 100 ms)	0-255 (0 = disabled)
Scene 1	Disabled Enabled
Scene 1 scene address (0-63) (if "Enabled" is selected for scene 1)	0-63, 0
Scene 1 switching state (if "Enabled" is selected for scene 1)	Released Pressed
Scene 2	Disabled Enabled
Scene 2 scene address (0-63) (if "Enabled" is selected for scene 2)	0-63, 1
Scene 2 switching state (if "Enabled" is selected for scene 2)	Released Pressed
Scene 3	Disabled Enabled
Scene 3 scene address (0-63) (if "Enabled" is selected for scene 3)	0-63, 2
Scene 3 switching state (if "Enabled" is selected for scene 3)	Released Pressed
Scene 4	Disabled Enabled

Channel X: Scenes (if "Enabled" is selected for scenes)	
Parameter	Setting
Scene 4 scene address (0-63) (if "Enabled" is selected for scene 4)	0-63, <b>3</b>
Scene 4 switching state (if "Enabled" is selected for scene 4)	<b>Released</b> Pressed
Scene 5	<b>Disabled</b> Enabled
Scene 5 scene address (0-63) (if "Enabled" is selected for scene 5)	0-63, <b>4</b>
Scene 5 switching state (if "Enabled" is selected for scene 5)	<b>Released</b> Pressed
Scene 6	<b>Disabled</b> Enabled
Scene 6 scene address (0-63) (if "Enabled" is selected for scene 6)	0-63, <b>5</b>
Scene 6 switching state (if "Enabled" is selected for scene 6)	<b>Released</b> Pressed
Scene 7	<b>Disabled</b> Enabled
Scene 7 scene address (0-63) (if "Enabled" is selected for scene 7)	0-63, <b>6</b>
Scene 7 switching state (if "Enabled" is selected for scene 7)	<b>Released</b> Pressed
Scene 8	<b>Disabled</b> Enabled
Scene 8 scene address (0-63) (if "Enabled" is selected for scene 8)	0-63, <b>7</b>
Scene 8 switching state (if "Enabled" is selected for scene 8)	<b>Released</b> Pressed

## Overwriting scene values on download

If you enabled the parameter "Overwrite scene values in actuator during download", then your default values overwrite the scene values saved in the device on download. If you don't want to overwrite the values in the device, disable this parameter. In this case, only those scene values for which the parameters were set during the first download are written into the device memory. If an application is then downloaded, the scene values will remain in the device memory.

## Priority

The scene function has the same priority as the normal switching function via the "Switch object". This should be taken into account with regard to the priority of the higher-level functions.

## Communication objects

The scene telegrams can be received via the "Scene object" communication object:

Function	Object name	Type	Prio	Flags	Behaviour
Scene	Scene object	1 byte	Low	CW	Receive

## Higher priority functions

### ● Lock function

You can use the Lock function to systematically press/release a channel, and to disable it in this position. The state of the output channel cannot be changed by other control commands as long as locking is active. You can activate the Lock function for each switching channel individually.

After the Lock function is enabled, a new communication object called "Lock object" and a new tab called "Channel X: Lock function" are available for the switching function. You can use "Lock object" to activate and deactivate a channel block.

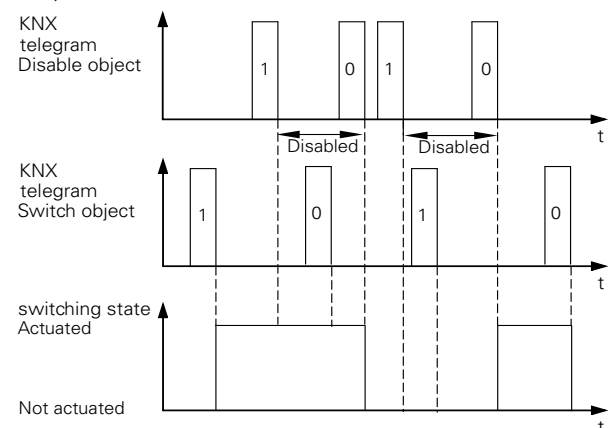
If the "Lock object" receives a telegram with the object value which you have specified for the "Lock" parameter, then all other channel functions are blocked. You can define the reaction via the "Behaviour at start of locking" parameter:

If the disable object receives a telegram with the opposite object value to the activation, the lock is removed and the output relay adopts the state which you have specified in the "Behaviour at end of locking" parameter:

If a higher priority function is active after the Lock function has ended, the "Behaviour after end of locking" parameter is ineffective. The state of the higher priority function is immediately adopted.

**i** The Lock function always switches immediately. The last switching telegram is saved during a Lock function and any delay periods or staircase timers continue to run.

Lock for object value "1"  
Behaviour at start of locking = no reaction;  
Behaviour at end of locking = follows current value;  
Relay mode: make contact



## Parameter

Per channel: Select the disable function.

Channel X: Switching	
Parameter	Setting
Higher priority function	Locking operation
	Priority function

Per channel: Set the disable function.

Channel X: Disable	
Parameter	Setting
Lock	At object value "0" At object value "1"
Behaviour at start of locking	No reaction
	Pressed
	Released
Behaviour at end of locking	No reaction
	Follows current value
	Pressed
	Released
Lock behaviour at download (only if a higher priority function is inactive)	Deactivated
	Activated
	As before download
Behaviour of the locking after bus voltage recovery	Deactivated
	Activated
	As before bus voltage failure

## Locking behaviour after download

After downloading, the disable function is also set as it was after a bus voltage recovery. The "Relay state after download" parameter in the "Channel X: Failure behaviour" parameter tab determines the state that is set.

If the "Relay state after download" parameter is set to the value "As before download", the disable function is activated as previously specified, and the relay is switched accordingly.

## Behaviour of locking after bus voltage recovery

- Deactivated  
The disable function is not activated after bus recovery, regardless of its status prior to bus voltage failure.
- Activated  
The disable function is activated after bus recovery and the output is switched back to the state you specified in the "Behaviour at start of locking" parameter. If you have set the "no reaction" value here, the output is blocked in its current state.
- As before bus voltage failure  
The disable function is switched to the state which was active before the bus voltage failure. If the disable function was active, the output is controlled by its settings in the "Behaviour at start of locking" parameter.

## Communication objects

The following communication objects can be selected:

Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Channel X	Lock object	1 bit	Low	CW	Receive

## ● Priority function

If you have selected priority function as the higher priority function for a switching channel, a new communication object called "Priority object" is available for this channel.

The object values of the priority object have the following meaning:

Value Bit 1	Value Bit 2	Behaviour of the output
1	1	Activate priority function, "pressed" switching state
0	1	Deactivate priority function, switching state depends on the "Behaviour after the end of priority function" parameter
1	0	Activate priority function, "released" switching state
0	0	End of priority function, switching state depends on the "Behaviour after the end of priority function" parameter

Priority function is activated when the value "1" is received on bit 1. The assigned output relay is then switched to "pressed" (bit 2 = "1") or "released" (bit 2 = "0"), depending on bit 2.

An active priority function is then enabled again by a new telegram with the value "0" on bit 1. As long as priority function is active, the relevant channel cannot be controlled via the "Switch object" or the advanced functions (central function, time functions, scene function).

After the end of a priority function, the behaviour of the output relay is determined by the "Behaviour at end of priority function" parameter:

## Parameter

Per channel: Select the priority function.

Channel X: Switching	
Parameter	Setting
Higher priority function	Lock function
	Priority function

Per channel: Set the behaviour of the priority function.

Channel X: Disable	
Parameter	Setting
Behaviour after end of priority function	Follows current value
	Pressed
	Released
Behaviour of priority function after bus voltage recovery	Disabled
	Enabled, released
	Enabled, pressed
	As before bus voltage failure

The setting values lead to the following reactions after a priority function is lifted:

- Follows current value  
During the priority function, all switching commands of subordinate functions (basic function and advanced functions) are followed by the application and the switch state is updated internally. In this way, at the end of a priority function, the switching state can be set which it would currently have had without the priority function.

## Behaviour of priority function after bus voltage recovery

You can use the "Behaviour of priority function after bus voltage recovery" parameter to specify the reaction of the channel after bus recovery and the switching state:

- Locked  
The priority function remains deactivated. The switching state of the channel is derived from other higher priority functions or from the set switching behaviour after bus recovery.
- Enabled, released  
The priority function is automatically activated when the bus voltage recovers, and the switching state is switched to the "released" state.
- Enabled, pressed  
The priority function is automatically activated when the bus voltage recovers, and the switching state is switched to the "pressed" state.
- As before bus voltage failure  
The priority control function is set to the status which it had before the bus voltage failure. If the priority control was previously active, the output relay is switched to the state which it had before.

## Communication objects

The following communication objects can be selected:

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Channel X	Priority object	2 bit	Low	CW	Receive

## Status messages/status response

### ● Status messages/status response

The software application provides you with options for reporting the current device status and the status of output channels via communication objects.

Depending on these status-/status response, you can trigger certain control processes or enable and disable functions in the system.


The status-/status response of the output channels are made available through communication objects. You can set the transmission characteristics of these objects to one of the following values using parameters:

- **Active status response object**  
The new value is automatically sent to the bus after the switching state of the output relay (e.g. relay pressed or relay released) is changed.
- **Passive status object**  
The status object does not transmit its values. However, the current status is available and can be read out by other bus devices. The status messages are also suitable for displaying current operating statuses by means of visualisation software.

You can parameterise the status-/status response object such that only certain changes are reported. Three variants are available:

Reports status-/status response object:

- At changes by bus and manipulation
- At changes by bus
- At changes by manipulation

 Manual operation and status-/status response are described in the Manual operation chapter.

You can assign two different telegram values to the status-/status feedback objects:

- Pressed = 1; Released = 0
- Pressed = 0; Released = 1

Use this to determine the value that is sent as status-/status response objects.

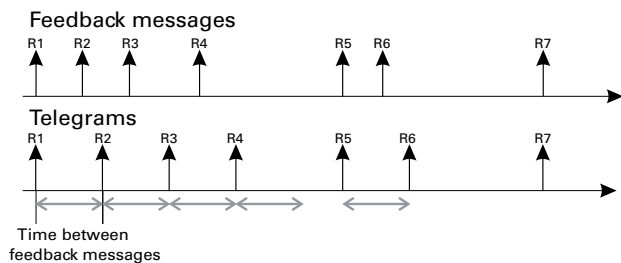
## Delayed sending behaviour for status responses

You can set the delay of status feedbacks from this device and the interval between multiple status feedback telegrams with a higher priority in the "General" tab.

### Parameter

General	
Parameter	Setting
Delay status response all channels, 100 ms * factor (0-65535)	0-65535
Time between status response per channel, 100 ms * factor (0-255)	0-255, <b>2</b>

If there is only one pending transmission job, the message is sent as set in the "Delay status response all channels" parameter. If more than one status response is pending, the other status responses are sent using the pattern of the delay time determined by the "Time between status response per channel" parameter.



## Activate status-/status response

If you want to activate the status-/status response for an output channel, you must select the corresponding "Status-/status response" for every channel. The "Active status response object" or the "Passive status object" is available to you.

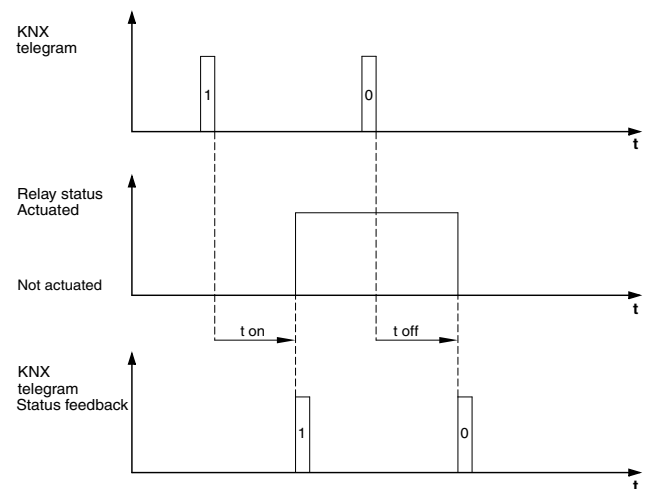
### Parameter

Channel X: Switching	
Parameter	Setting
Status/status response	Disabled
	Active status response object
	Passive status object
Status/status response object	For bus and manual mode change
	For bus change
	For manual mode change
Value status/status response object	Pressed = 1, Released = 0
	Pressed = 0, Released = 1

### Use status-/status feedback

- Disabled  
The status-/status response object is not active.
- Active status response object  
If you have set the "Status-/status response" parameter to the "Active status response object" value, the switching status of the output relay (e.g. Relay activated or Relay not activated) is transmitted each time the state is changed via the "Status response object" object.

Example: Active status response with an On and Off delay



- Passive status object  
If you have specified the object behaviour as "passive status object", the output status is not transmitted when the status changes, but is simply updated in the "Status object" object. The value of the "Status object" can then be read out by another bus device, e.g. by a visualisation software program.

## Communication objects

The following communication objects are available:

### Per channel:

(depending on the selected status-/status response)

Function	Object name	Type	Prio	Flags	Behaviour
Status response	Status response object	1 bit	Low	CRT	Transmit/read out
Status report	Status object	1 bit	Low	CR	Read out

### ● Collected status/status feedback object

You can send status-/status response with a time delay using 4-byte objects coded by bit using the collected status/status response object.

Up to 16 channels are integrated into one collected status/status feedback in order to save group addresses and to reduce the bus load, e.g. for a central off telegram.

The 4-byte object is constructed as follows. The upper 2 bytes show which condition bit is valid ("1" = valid, "0" = invalid). The lower 2 bytes show the conditions (actuated or not actuated) of the channels.

**For example:** 12-gang actuator, channels 2 and 6 are actuated

00001111 11111111 00000000 00100010

You can parameterise the collected status/status response objects such that only certain changes are reported. Three variants are available:

Reports collected status/status response object:

- At changes by bus and manipulation
- At changes by bus
- At changes by manipulation

You can define the value (pressed = 1, released = 0 or pressed = 0, released = 1) of the collected status/status response using the "Assignment channel status to value of bit" parameter.

In case of a status response object, you can also parameterise a delay time for the status response telegrams to be sent.

## Activate collected status/status response

If you want to activate the collected status/status response for the device, you must select collected status/status response in the "General" tab. The "Status response object" or the "Status object" is available to you.

### Parameter

General	
Parameter	Setting
Collected status/status response	<b>Disabled</b>
	Status response object
	Status object
Collected status/status response object	<b>At changes by bus and manipulation</b>
	At changes by bus
	At changes by manipulation
Assignment channel status to bit value	<b>Pressed = 1, Released = 0</b>
	Pressed = 0, Released = 1
Delay time sending (1 s * factor; 0-255) (if "Status response object" is selected for collected status/status response)	0-255, <b>60</b>
Behaviour after bus voltage recovery	<b>No reaction</b>
	Sends current status

## Behaviour after bus voltage recovery

- No reaction  
The collected status/status response remains unchanged in its current position.
- Sends actual status  
The current status of the output channels is sent to the bus after the set sending delay time elapses. However, only those messages are sent which you have defined as "status response object".

## Communication objects

The following communication objects are available for the device:

Function	Object name	Type	Prio	Flags	Behaviour
Collected status/status response	Collected status/status response object	4 byte	Low	CRT	Transmit/read out

## Behaviour after bus voltage failure / after bus voltage recovery / after download

If you want the actuator to respond in a certain way if the supply voltage fails unexpectedly during operation, then you can configure failure behaviour for each channel.

If you have enabled the failure behaviour parameter for a channel, an additional tab called "Channel X Failure behaviour" in which you can set other parameters appears.

### Parameter

Channel X: Switching	
Parameter	Setting
Failure behaviour	<b>Disabled</b>
	Enabled

For the device, you can set a global transmission delay of all telegrams after bus recovery. You can specify the delay time under the "Sending delay after bus voltage recovery in s" parameter in the "General" tab.

### Parameter

General	
Parameter	Setting
Sending delay after bus voltage recovery in s	0-255, <b>30</b>

## ● Behaviour when bus voltage fails

When the bus voltage drops below 18 V, the relay can adopt a parameterised state. The relay state can be either "pressed" or "released", or remain in the state it was in before the failure. At the same time, the current switching position of the relay is stored in the device. Active time delays are not saved. No further status feedback telegrams are sent.

The RUN LED is not lit when bus voltage fails.

## Activate behaviour after bus voltage failure

If you want to activate the behaviour after bus voltage failure for an output channel, you must select a "Relay state after bus voltage failure" for every channel. You can set the failure behaviour to "no reaction", "pressed" or "released".

### Parameter

Channel X: Failure	
Parameter	Setting
Relay state after bus voltage failure	<b>No reaction</b>
	Pressed
	Released

## Possible settings:

- No reaction  
The relay contact remains unchanged in its current position. If time functions (staircase lighting function, ON delay, OFF delay) are currently active, these are interrupted.
- Pressed  
The relay is closed with a make contact and opened with a break contact. Current time functions are deactivated.
- Released  
The relay is opened with a make contact, and closed with a break contact. Current time functions are deactivated.

## ● Behaviour on bus voltage recovery

The relay can adopt a parameterised state after bus voltage recovery.

The RUN LED is lit, thus indicating that bus voltage is present again.

Thirty seconds after bus voltage is once again present and stable, the device starts transmitting the activated messages of all the channels in succession.

## Activate behaviour after bus voltage recovery

If you want to activate the behaviour after bus voltage recovery for an output channel, you must select a "Relay state after bus voltage recovery" for every channel. You can set the behaviour to "no reaction", "pressed" or "released" or "as before bus voltage failure".

## Parameter

Channel X: Failure	
Parameter	Setting
Relay state after bus voltage recovery	<b>No reaction</b>
	Pressed
	Released
	As for bus voltage failure
Status feedback behaviour after bus voltage recovery	<b>No reaction</b>
	Sends curr. status

## Possible settings:

- No reaction  
The relay contact remains unchanged in its current position.
- Pressed  
The relay is closed with a make contact and opened with a break contact.
- Released  
With a make contact, the relay is opened, and with a break contact, it is closed.
- As for bus voltage failure  
In the "as for bus voltage failure" parameter, the relay adopts the state which was saved in the device at the time of bus voltage failure. Any manual operations that occur in the meantime are overwritten.

## Status feedback behaviour after bus voltage recovery

You can set the behaviour of status response objects after bus voltage recovery as follows:

- No reaction  
No status response is sent.
- Sends actual status  
The actual state of the output channels is sent to the bus after the set transmission delay time elapses. However, only those messages you have defined as "active status response object" are transmitted.

## Priority:

The reaction to the behaviour set here when the bus voltage returns has a lower priority. If a higher-level function is activated for the switching channel directly after the bus voltage returns, the following settings apply for these functions.



### Important:

Short switching times may be not parameterised under load (see the technical data for the switch output).



### Important:

Relay states that are caused by functions with a higher priority (higher priority function) take precedence over the behaviour after bus voltage recovery.

Example: OR logic operation with parameterised value of the logic object after bus voltage recovery = 1 takes priority and switches the output.

## ● Behaviour after ETS download

Before the first download after commissioning, all relays are switched off (default setting).

The relay can adopt a parameterised state after the ETS download.

If an internal error or a defective download results in a status in which the application is no longer operational, the device does not react. The output relays remain in their last position. The RUN-LED does not light up.

## Activate the behaviour after ETS download

If you want to activate the behaviour after ETS download for an output channel, you must select a "Relay state after download" for every channel. You can set the download behaviour of the output channels to "no reaction", "pressed", "released" or "as before download".

## Parameter

Channel X: Failure	
Parameter	Setting
Relay status after download	<b>No reaction</b>
	Pressed
	Released
	As before download

### Possible settings:

- No reaction  
After a download, the relays remain in their current status. This does not result in the relays being switched. Any manual operation that occurs in the meantime is not overwritten.  
This function is conceived for the reconfiguration of switch actuators in occupied buildings so that the outputs do not switch in the event of a download. The device does not know the status of the channel at this point and therefore no status signal can take place. The status is only available following a switching action of the channel.
- Pressed  
The relay is closed with a make contact and opened with a break contact.
- Released  
The relay is opened with a make contact, and closed with a break contact.
- As before download  
The relays retain the behaviour that was set before the download. Any manual operation that occurs in the meantime is overwritten. If a higher-level function (logic operation, priority control or lock) is active, the behaviour is executed which you have specified for these functions.



### Important:

Relay states that are caused by functions with a higher priority (higher priority function), take precedence over the behaviour after ETS download.  
Example: OR logic operation with parameterised value of the logic object after bus voltage recovery = 1 takes priority and switches the output.

## Safety function

The safety function is a 1 bit communication object with high priority. This means that the object has priority **over** the following communication objects:

- Logic object A (logic function)
- Value objects B and C (extended logic function)
- Scene object
- Switch object
- Staircase lighting object
- Central object



### Note:

The safety object is a "general object" and is visible only as an object in the ETS software. In individual channels, the safety object must be enabled again.

The safety object is enabled in the "general parameters". In the relevant switch channel, the safety object must be enabled or disabled again.

## Parameter

General	
Parameter	Setting
Safety function	<b>Disabled</b>
	Safety at "1"
	Safety at "0"

- Safety at "1": The communication object switches the safety function on at logical "1" and switches it off at logical "0".
- Safety at "0": The communication object switches the safety function on at logical "0" and switches it off at logical "1".

Channel X switching	
Parameter	Setting
Safety function	Enabled
	<b>Disabled</b>

## ● Channel X safety function

The following selection can be made in the ETS software in the relevant channel of the safety function:

## Parameter

Channel X: Safety function	
Parameter	Setting
Behaviour at safety state	<b>No reaction</b>
	Pressed
	Released
	Blinking, 1 s frequency
Behaviour at end of safety state	<b>Follows the currently valid status</b>
	Pressed
	Released
	No reaction

Channel X: Safety function	
Parameter	Setting
Behaviour at exceeding cycle time	<b>No reaction</b>
	Pressed
	Released
	Blinking, 1 s frequency

## Behaviour at the beginning and end of safety and when the cycle is exceeded

- Relay activation can be parameterised at the beginning and end of safety as well as when the cycle is exceeded.



### Note:

How the relay switches (break contact or make contact) depends on the operation mode.

## Communication objects

The following communication objects can be selected:

Function	Object name	Type	Prio	Flags	Behaviour
Safety function	Safety object - general	1 bit	Low	CW	Receive

## Manual mode

Channel buttons for the mechanical actuation of every channel are provided on the front side of the device. The manual buttons switch the relay directly and mechanically.



### Caution!

Loads could be damaged. Higher-level safety functions do not work (priority operation, etc.) when loads are operated using the channel buttons. To prevent damage, you should therefore be very careful when operating manually.

After applying the mains voltage, you can also control the connected loads without a bus connection via the channel buttons. The current switching state can be read at the switch position.

## ● Automatic cancelling of manual mode

This switching actuator follows the manual buttons in accordance with the switching states specified by the device and always checks whether the relay corresponds to the bus switching state.

If not, the device interprets this as manual mode.

If manual mode has been carried out, the channel remains in its new state until a new telegram comes via the bus (e.g. switch object, staircase lighting object, etc.), or the channel has been parameterised to reset after a specified time or has been parameterised to reset via object.

Manual operation can be reset via the "Reset manual operation" object. The relay again follows the bus switch commands.

You can also parameterise a time after which the relay again follows the bus switch commands, thus when manual operation is also automatically reset to the current object value.



**Risk of fatal injury!** Persons could be injured and loads could be damaged.

When parameterising "Reset manipulation" in the ETS, the relay could switch unintentionally and provide current to the output after a manual operation.

Take special care when parameterising "Reset manipulation" to prevent injuries and damage. Put up the following warning note at the appropriate channels on the switching actuator as well as the connected devices:

### Caution!

This current connection can switch automatically even when the switching actuator is in manual operation.

## Parameter

General	
Parameter	Setting
Autom. reset of manual operation to actual object value	
Time until reset 1s*factor (0-65535, 0=not reset)	0 - 65535
Status response manual operation	<b>Disabled</b>
	At manual mode (A)
	At failed autom. reset (B)
	At (A) OR (B)

Channel X:	
Parameter	Setting
Reset manipulation	<b>Disabled</b>
	After time
	If Reset manual operation object = 1

## Status response manual operation

The "Status response manual operation" object can be used to transmit the manual mode status (of all channels with a 4 byte object). You can parameterise the time for transmitting the status:

- Disabled. No status response for manual operation.
- For manual mode. As soon as a channel button is actuated.
- For failed autom. reset. If the automatic reset has not been executed correctly (e.g. if the channel button has been mechanically blocked).
- For manual operation or if the automatic reset has not been executed correctly.

**i** Note that when the bus load is high (many telegrams in a short time) the status feedback of the manual mode can be delayed or even impossible.

## Reset manipulation

Use the "Automatic reset of manual operation to actual object value" parameter to set the time at which a manual mode should be automatically reset by the device for every channel separately:

- Disabled. No automatic manual operation reset.
- After time. The manual operation is reset after the time set in the "Time until reset" parameter in the "General" parameter tab (after the channel button is pressed).
- If the Reset manual operation object=1. As soon as value "1" is received via the "Reset manual operation=1" object.

## Communication objects

The following communication objects are available:

Function	Object name	Type	Prio	Flags	Behaviour
Reset	Reset manual operation	1 bit	Low	CW	Receive

Function	Object name	Type	Prio	Flags	Behaviour
Status response	Status response manual operation channel	4 byte	Low	CRT	Transmit

## Current detection

The actuator has hall sensors which can detect and analyse the load current of each switch channel. These analyses can be used for additional operations and functions.

Currents higher than 100 mA and a minimum duration of approx 200 ms (duration of the measuring cycle at 50 Hz; approx 120 ms at 45 Hz) are detected. These values are derived from the internal filters and error corrections (temperature compensation, etc.).

For channels for which current measurement has been parameterised, the current is measured in succession.

**i** Non-sinusoidal currents (e.g. phase-controlled) or alternating currents with overlapping direct current shares (e.g. using current converters) lead to measuring errors and can thus also lead to malfunctions if current analysis continues to be used.

## Current detection areas of application

- Alternating currents or direct currents (can be selected per channel via parameters)
- Detection range: 0.1 A to 16 A
- Frequency range AC: 50 Hz/60 Hz (sinusoidal).
- Min duration of the load current for correct detection: 200 ms
- Max error in the current value detection: 8%
- Max error in the internal filters: 100 mA

**i** A measuring cycle always lasts 200 ms for every channel. Every channel for which the current measurement has been parameterised is measured in succession.

## Detection/measurement of direct current

Channels for detecting and analysing direct currents are calibrated during the initial measurements. The measurements are within the specified error limits only after the channels have been operated with maximum load for a longer period (switch on and switch off again).

**i** Measurements of a non-calibrated channel with direct current can lead to an error of up to 1.5 A.

To obtain reliable measurements immediately after installation, you can load all channels with the maximum possible current for approx 2.5 hours after connection (e.g. by switching on all downstream loads):

- Load all channels with the maximum expected load.
- Switch off the load only after approx 2.5 hours.

- Cool down the switch actuator for at least 2 hours (without actuating the relay).



The allowed connected loads may not be exceeded.

The following table contains the probable increase in temperature of the switch actuator (or the relay), depending on the load current for all channels.

Load current (all channels)	Temperature increase
4A	1.4 °C
8A	6.0 °C
12A	12.0 °C
16A	22.0 °C

## Error analysis



The probable errors (current value detection errors) must be taken into account when continuing to use the current value. In principle, both errors are added.

Example:

A detected current value of 400 mA can actually be within the range of 268 mA to 532 mA due to the natural error.

This is derived from 8% of 400 mA = 32 mA and 100 mA filter error. The inaccuracy is thus plus/minus 132 mA, i.e. 400 mA minus 132 mA and 400 mA plus 132 mA.

The error values are important for further processing of the detected current value.



If the currents flowing through the channels are different (direct current/alternating current/different frequencies), the error correction no longer functions reliably. The measured values can fluctuate and are inaccurate.

## ● Setting the current type

A current type must be set for every channel for correct current detection and analysis.

Current detection	
Parameter	Setting
Current type	Alternating current
	Direct current



For direct current, 12-24 V +10% is permissible.

## ● Transmitting the current value

The current value can be sent to the bus via the "Current value" object.

Current detection	
Parameter	Setting
Object current value	1 byte (base = 100 mA)
	2 byte fixed decimal (base = 1mA)
	2 byte floating point

Current detection	
Parameter	Setting
Transmitting the current value	Not (read only)
	At changes
	Cyclic
	Cyclic and at changes
At change of	10%.... <b>20%</b> ....50%
Base cyclical interval	0.1 s to <b>1.0 s</b>
Cyclical interval factor	2 to <b>10</b> to 65535

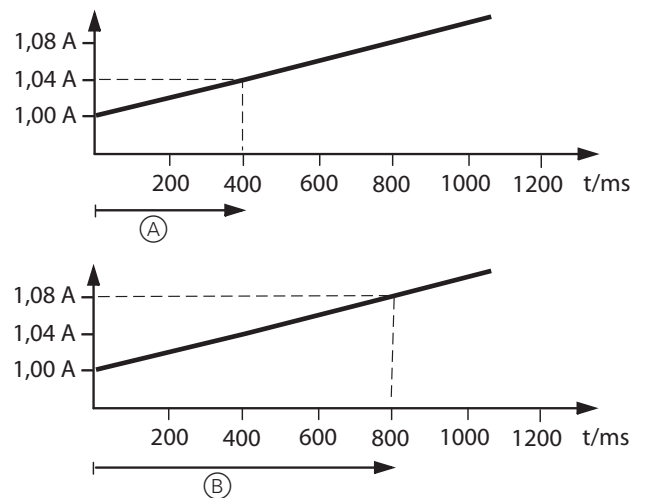
## Send current value on change

The current value change can be analysed only within the current detection application range.

For channels for which current measurement has been parameterised, the current value is measured for the respective duration of a measuring cycle in succession.

This means the intervals between measurements used for determining the change increases when the number of measured channels increases. This must be taken into account when parameterising the change value.

Example:



A load at a channel of a 12-gang switch actuator has a linear increase in the current as shown in the image (change of 0.2 A/200 ms). You parameterised that when the value is exceeded by 5%, a telegram is sent. In the first case, the current is measured for two channels only. For current measurement for two channels, the measuring cycle takes 400 ms (2x200 ms; (A)) for one channel. After 400 ms, the current has increased from 1.0 A to 1.04 A, i.e. by 4%. No telegram is sent. In the second case, the current is measured for four channels. The measuring cycle now takes 800 ms (4x200 ms; (B)) per channel. After 800 ms, the current increases from 1.0 A to 1.08 A, i.e. by 8%. Now, a telegram is sent.

## Sending current value cyclically

You can define a time interval for sending the current value.

The measuring cycle is always 200 ms per channel. The channels for which current measurement has been parameterised are always measured in succession. The measurements of the channel are saved (max 16 measurements) and used to calculate an average value until the value is actually sent (cyclic interval). This means, the longer the cyclic interval, the more accurate the value sent. The more measurements recorded, the effect of short-term current fluctuations and natural measuring errors decreases.

Also adjust the cyclic interval to the number of channels for which current measurement has been parameterised. The lower the number of measured channels, the higher the number of measurements that are recorded and analysed for every channel per cyclic interval.

## Communication objects

The following communication objects can be selected:

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Current detection	Current value	1 byte, 2 byte FK, 2 byte GK	Low	CTR	Transmit

For 1 byte telegrams, the base is 100 mA,  
For 2 byte telegrams, 1 mA.

### ● Analysis of current values

The current value for every channel can be analysed internally by the device. You can define up to two limit values (current values) to compare the latest respective current value to. You can also parameterise a reaction if the limits are exceeded and/or fallen short of.

As a reaction, you can also parameterise that a value is sent via the "Status value X" objects.



16 measurements are carried out for every channel and an average value is calculated from them for limit value comparison. This value is compared to the limit value.

The higher the number of channels parameterised for current measurement, the longer it takes to carry out 16 measurements (200 ms) for every channel and compare the limit values. If there are twelve channels, it can take up to 38,4 s.



Selecting "None" for the "Reaction if exceeded.../Reaction if undercut..." parameters is practical if the exceeding/undercut is used only for the counter.

You can also set a hysteresis (in mA) for every limit value in order to prevent undesired reactions when current values are around the limit value.



The value for the hysteresis is symmetrical. A telegram is sent at 900 if the limit value is 1000 and the hysteresis value is 100.

Furthermore, you can also delay analysis after closing the relay in order to hide any short-term current spikes.

Current detection	
Parameter	Setting
Analysis	No
	With 1 limit value
	With two limit values
Limit valueX	
Limit valueX 1mA * factor	200 to <b>1000</b> to 65535
Hysteresis, 1mA * factor	100 to <b>200</b> to 8000
Delay of analysis 100ms * factor	0 to <b>5</b> to 255
Reaction if exceeded Device/Object status current limit valueX	None
	Released/None
	None/"0"
	None/"1"
	Released/"0"
	Released/"1"
Reaction if undercut Device/Object status current limit valueX	None
	Released/None
	None/"0"
	None/"1"
	Released/"0"
	Released/"1"

## Communication objects

The following communication objects can be selected:

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Current detection	Status current limit value1	1 bit	Low	CTR	Transmit
Current detection	Status current limit value2	1 bit	Low	CTR	Transmit

### ● Energy consumption calculation

In addition to current values, the energy consumption can also be sent as a value (via the "Energy consumption" object).

You can send the calculated energy consumption either in 1 kWh steps, or cyclically or in both ways.

Furthermore you can parameterise a limit, where a 1-bit telegram will be sent if the limit is reached. If the limit is reached the energy consumption will be reset to 0 (for that the limit can be reached again). So you can use this parameter also to prevent an „overflow“ of the energy consumption value.

**i** At limit 0 no telegram will be send over the object Energy limit value reached and of course there is no reset of the energy consumption value. The energy consumption value will be calculated up to the possible maximum of the telegram type.

**i** Up to 32 measurements are carried out per channel and used to calculate an average value for calculating the energy consumption. The accuracy of the current measurement increases with the number of measurements. Inductive and capacitive loads increase the inaccuracy of the measurements.

Current detection	
Parameter	Setting
Voltage for calculation 1V * factor	12 to <b>230</b> to 253
Transmitting consumption	<b>Not (read only)</b>
	All 1 kWh
	Cyclic
	Cyclic and all 1 kWh
Base cyclical interval	1 min to <b>1h</b>
Cyclical interval factor	<b>1</b> to 255
Limit (at limit reset consumption to 0) 1kWh * Faktor (0-65535)	<b>0</b> ..65535
Format energy consumption	Wh
	Ws (J)

For calculating the energy consumption, you must know the voltage of the connected circuit and must enter it via the relevant parameter. The time is internally recorded. The energy consumption is then calculated using these values.

**i** Energy consumption is a calculated value. The values are thus comparison values only.

**i** Energy consumption can be sent in Wh or Ws(J). 4 byte without a sign is used for Wh and 4 byte with a floating point object is used for Ws(Y).

## Communication objects

The following communication objects can be selected:

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Current detection	Energy usage	4 byte	Low	CTW(R)	Transmit
Current detection	Energy limit value reached	1 bit	Low	CTW(R)	Transmit

## Counter

You can count and analyse the switching of the relay, the number of times the limit value was exceeded and the operating hours.

Counter	
Parameter	Setting
Counter	<b>None</b>
	Switch counter
	Operating hours counter
	Switch and operating hours counter
	Combined counter (= 1 value)
Object switch counter	<b>4 byte fixed decimal</b>
	2 byte fixed decimal
Switch counter condition	<b>if relay is closed</b>
	if (current value) limit is exceeded
Switch counter limit	<b>0 to 65536</b>
Object operating hours counter	<b>4 byte fixed decimal</b>
	2 byte fixed decimal
Operating hours counter condition	<b>if relay is closed</b>
	if (current value) limit 1 is exceeded
Operating hours counter limit	<b>0 to 65536</b>
Combined counter limit (combined counter=x*switching+y*hours)	<b>0 to 999999</b>
Factor x	0 to 255
Factor y	<b>0</b> to 255

### Switch counter condition/operating hours if (current) limit value is exceeded

The number of times the current limit values (parameter "Limit value1" in the Current detection tab) have been exceeded/fallen below is counted. The counting depends on the conditions that have been parameterised in the "Current detection" tab.

Current detection	
Parameter	Setting
Analysis	None
	With 1 limit value
	With two limit values
Limit valueX	
Limit valueX 1mA * factor	0.2 to <b>1.0</b> to 1.2
Hysteresis	5% to <b>20%</b> to 50%
Delay of analysis 100ms * factor	0 to <b>5</b> to 255
Reaction at exceeding Device/Object status current value limitX	None
	Not actuated/none
	Actuated/none
	None/"0"
	None/"1"
	Not actuated/"0"
	Actuated/"0"
	Not actuated/"1"
	Actuated/"1"

Current detection	
Parameter	Setting
Reaction for falling below Device/Object status current value limitX	None
	Not actuated/none
	Actuated/none
	None/"0"
	None/"1"
	Not actuated/"0"
	Actuated/"0"
	Not actuated/"1"
	Actuated/"1"

## Switch counter limit value

If this value is exceeded, a telegram is send via the "Switch counter limit reached" object. When the value is reached, the counter is reset to zero.

## Combined counter

In addition to the switch and operating hours counter, you can calculate a combined value and even define a limit value for this. The combined value is calculated according to the following formula:  

$$\text{Factor } x * \text{switch counter} + \text{factor } y * \text{operating hours counter}$$

A weighting can be defined using both these factors.

The combined value is particularly applicable for devices whose service life depends on the operating time as well as the initial frequency.

The values of the combined counter and the limit value excess is transmitted via the "Operating hours counter" and "Operating hours counter reached" objects.



Due to the calculation, the combined counter has greater values. Therefore, 4 byte fixed decimal should be parameterised as object type.

## Communication objects

The following communication objects can be selected:

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Counter	Switching counter	4 byte, 2 byte	Low	CW	Transmit
Counter	Switch counter limit reached	1 bit	Low	CW	Transmit
Counter	Operating hours counter	4 byte, 2 byte	Low	CW	Transmit
Counter	Operating hours counter limit reached	1 bit	Low	CW	Transmit

## Logic functions

Two logic functions respectively are available for every **switching channel**:

- Logic operation
- or
- Extended logic operation

The logic operation always activates the relay in the switch actuator. What this means: a logic operation for channel "1" activates the relay of the switching channel "1".

The extended logic operation can activate the corresponding relay of the switch actuator or can be used independently.

You can evaluate, process and write values on the KNX bus within your building function with the help of logic gates and input objects.

Example: You can connect two wind values (2 bytes each) from different house facades. The result of the AND logic operation is forwarded to the KNX bus and the roller shutters of the building are moved using the logic operation result.

The following parameter setting shows how to select the logic function.

### Parameter

Switching channel X	
Parameter	Setting
Logic function	<b>Disabled</b>
	Logic operation
	Extended logic operation

## ● Logic operation

You can implement simple logic gates using the logic operation. For every channel, the following objects are available as input objects:

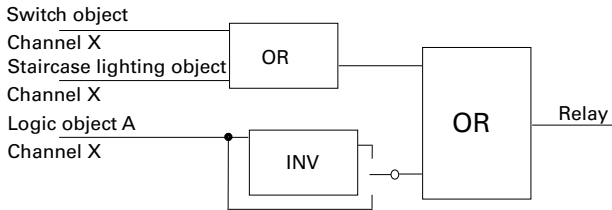
- Switch object channel X
- Staircase lighting object channel X
- Logic object A channel X

The output of the logic function can be processed only internally in the switch actuator. This means that the logic operation result is forwarded to the relay of the respective channel and not sent to the bus. The "Switch object" and "Staircase lighting object" input objects are always connected with an OR gate for the base logic function. The output of this OR gate is sent to a freely parameterisable gate: AND, OR, XOR. The following figures show the selection procedure of the logic operation.

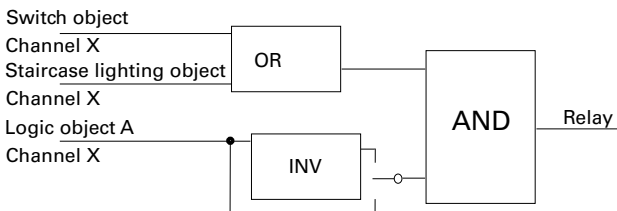
## Parameter

Logic operation: channel X	
Parameter	Setting
Type of logic operation	OR
	AND
	XOR

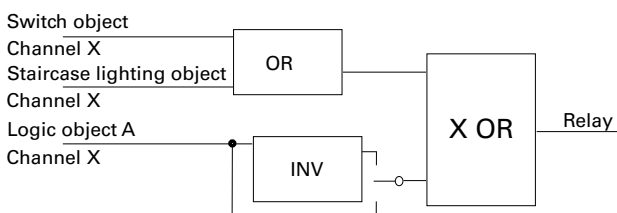
## Logic operation type: OR



## Logic operation type: AND



## Logic operation type: XOR



## Parameter

Base logic function channel X	
Parameter	Setting
Logic object A	Not inverted
	Inverted
Value of logic object A after bus voltage recovery and download	0
	1

Logic object A can be connected "Inverted" or "Not inverted". The value of logic object A can be parameterised after download.



## Note:

Note that the output of the logic operation (logic operation result) does not necessarily switch the relay directly. Other parameters such as central function, priority selection, etc. are also decisive for the switching state of the relay. For more details, refer to the overview plan.

## Communication objects

The following communication objects can be selected:

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Logic operation	Logic object A channel X	1 bit	Low	CW	Transmit
Logic operation	Switch object channel X	1 bit	Low	CW	Transmit
Logic operation	Staircase timer object fixed channel X	1 bit	Low	CW	Transmit
Logic operation	Staircase timer object variable channel X	2 byte	Low	CW	Transmit

## ● Extended logic operation

More complex logic operations can be realised with the help of the "Extended logic operation". For every channel, the following objects are available as input objects:

- Logic object A channel X
- Value object B channel X
- Value object C channel X
- Switch object channel X
- Staircase lighting object channel X

The value objects B and C are special. Here, even different byte values can be processed or compared with each other instead of 1-bit values. A filter connection and a gate function can also be integrated into the "Extended logic operation". The following gates can be selected as function modules: AND, OR, XOR.

The output of the "Extended logic operation" can be processed internally in the switch actuator in different ways:

- The logic operation result is sent to the relay of the corresponding channel.
- The logic operation result is sent to the bus as a telegram.
- The logic operation result is sent internally to another logic function. The output of this second logic function is sent to the relay of the corresponding channel.
- The logic operation result is sent internally to another logic function. The output of this second logic function is sent internally to the bus as a telegram.



**Note:**

The result of the extended logic function can be sent internally to the KNX bus. If this function is used, the switch object and the staircase lighting object cannot be used simultaneously to switch the relay of the corresponding channel.

**Structure of the "Extended logic operation"**

The "Extended logic operation" has been divided in **two parts**:

- Logic 1
- Logic 2

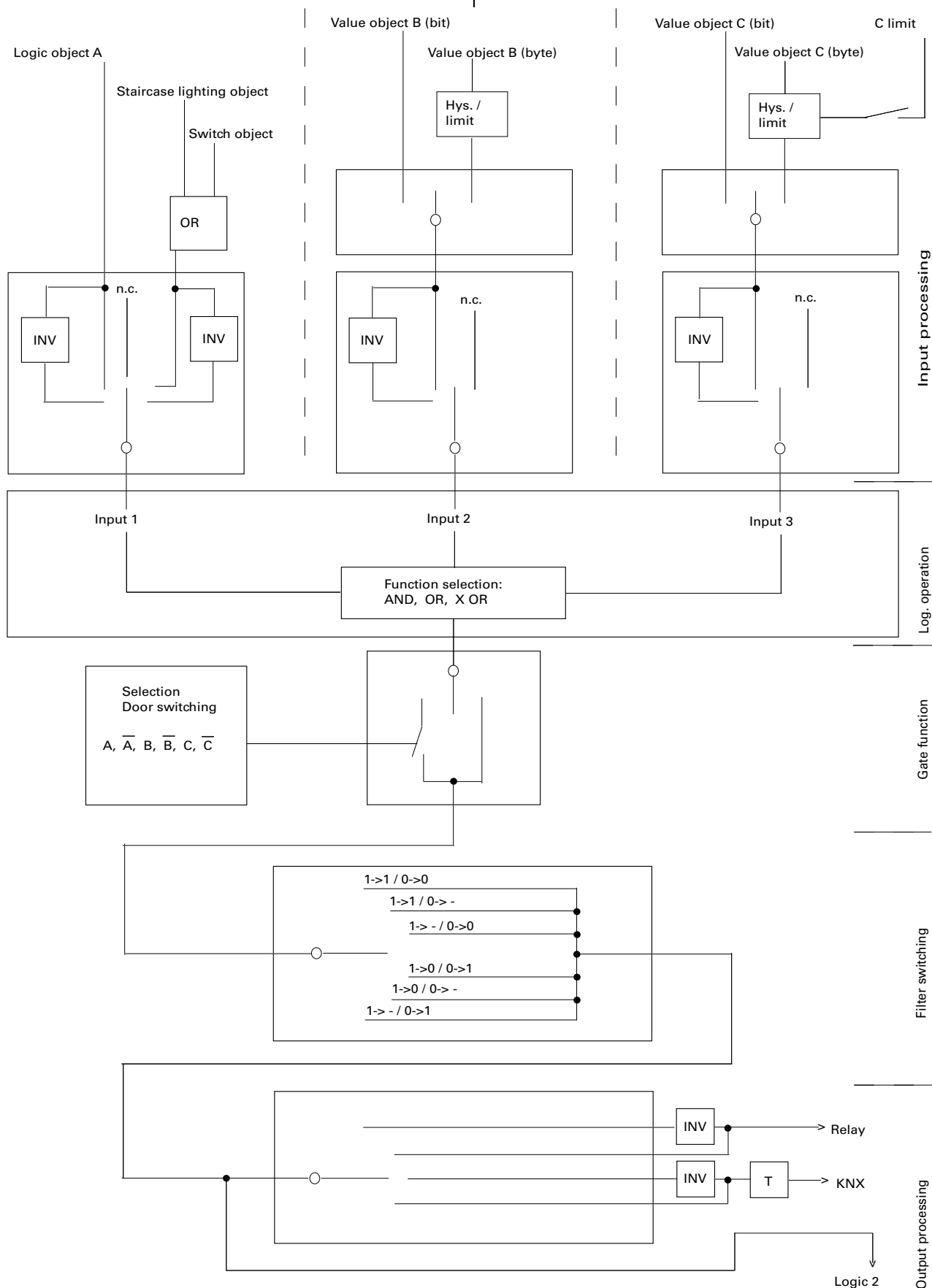
Every part of the logic function comprises

**five sections**:

- Input processing
- Logic function
- Gate function
- Filter connection
- Output processing

The individual sections have been set up like a series connection. Every section must be parameterised in the ETS software, so that the output signal can be used. The signal flow of the telegram is read from top to bottom. For more details, refer to the overview plan.

## Overview plan of the "Extended logic operation" as a block diagram (logic 1)



## "Input processing" (logic 1)

The input variables that affect logic 1 and the values of these objects are determined in the "input processing" section. The format for the B and C input objects can be selected (1 bit, 1 byte, etc.). For formats greater than

1 bit, comparison values can also be used as parameters.

## "Logic object A channel X, switch object channel X, Staircase lighting object channel X" (logic 1/input 1)

### Parameter

Extended logic operation	
Parameter	Setting
Value of logic object A after bus voltage recovery and download	<b>0</b>
	1

The value of logic object A can be parameterised after bus recovery and download.

### Parameter

Extended logic operation	
Parameter	Setting
Logic function 1 input 1	<b>Not used</b>
	Switch/staircase timer object
	Switch/staircase timer object inverted
	Logic object A
	Logic object A inverted

The "Logic object A channel X" and the "Switch object channel X" each have a value of 1 bit. The "Staircase lighting object channel X" has a value of 1 bit in case of a "fixed" staircase timer and of 2 bytes in case of a "variable" staircase timer. The switch object and the Staircase lighting object are always connected with an OR gate. The following options can be selected in the ETS software:

Parameter: Logic 1/Input 1

- Logic object A channel X not inverted
- Logic object A channel X inverted
- Staircase lighting object OR switch object not inverted
- Staircase lighting object OR switch object inverted
- Input 1 not used

## "Logic object B" (logic 1/input 2)

### Parameter

Extended logic operation	
Parameter	Setting
Logic object B and limit values	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign

Extended logic operation	
Parameter	Setting
	2-byte fixed point with sign
	2-byte floating decimal point
	4-byte fixed point without sign
Logic object B with (visible only if logic object B > 1 bit)	<b>1 limit value</b>
	2 limit values
Hysteresis value object B (symmetrically around limit value 1) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object B (symmetrically around limit value 2) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison B after bus recovery and download	<b>0</b>
	1
Limit value comparison B = 1 if logic object (visible only if logic object B > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)
Limit value comparison B limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison B limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 1 input 2	<b>Not used</b>
	Logic object B/limit value comparison B
	Logic object B/limit value comparison B inverted

The value of the "Logic object B channel X" can be 1 bit or higher. You have the following options in the ETS software:

Parameter: Logic 1/Input 2

- 1 bit
- 1 byte
- 2-byte fixed point without sign
- 2-byte fixed point with sign
- 2-byte floating decimal point
- 4-byte fixed point without sign

When 1 bit is selected, only the logical values "0" or "1" are processed for the input.

If byte is selected, other parameters in the ETS software are displayed. The logic object B can be compared with one or two limit values. The setting of the hysteresis in percent refers symmetrically to the respective limit value. It is also parameterised when the logic object B is defined as "1". The input value in bytes is thus again converted into a 1-bit value. The logical elements can process only binary states.

The "value of limit value comparison B" can be parameterised by the ETS software after bus recovery or download.

The following options can be selected in the ETS software:

Parameter: Logic 1/Input 2

- Logic object B with one or two limit values. (visible only if a value greater than 1 bit is set)
- Limit value comparison B limit value 1 or limit value 2, can be set depending on the selected data type. (visible only if a value greater than 1 bit is set)
- Hysteresis (symmetrically around the limit value/s) in percent. (visible only if a value greater than 1 bit is set)
- Determination, when the limit value comparison is defined as logical "1": Limit value comparison B = 1 if logic operation result . . . (visible only if a value greater than 1 bit is set)
- Value of limit value comparison B after bus recovery and download: Logical "0" or "1"
- Logic object B/limit value comparison B not inverted
- Logic object B/limit value comparison B inverted
- Input 2 not used

## "Logic object C" (logic 1/input 3)

### Parameter

Extended logic operation	
Parameter	Setting
Logic object C, limit values and object limit value comparison	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point
Logic object C with (visible only if logic object C > 1 bit)	4-byte fixed point without sign
	<b>1 limit value</b>
	2 limit values
Hysteresis value object C (symmetrically around limit value 1) (visible only if logic object C > 1 bit)	Object limit value logic operation C
	0-30%, <b>10 %</b>
Hysteresis value object C (symmetrically around limit value 2) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison C after bus recovery and download	<b>0</b>
	1
Limit value comparison C = 1 if logic object (visible only if logic object C > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)
Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison C limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting

Object limit value logic operation C	
Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting

Extended logic operation	
Parameter	Setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Value of limit value logic operation C after bus voltage recovery	Like parameter setting
	As before bus voltage failure
Value of limit value logic operation C after bus voltage recovery	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 1 input 3	<b>Not used</b>
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted
	Gate function (0 = lock 1 = release)
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted

The value of the "Logic object C channel X" can be 1 bit or higher. You have the following options in the ETS software:

Parameter: Logic 1/Input 3

- 1 bit
- 1 byte
- 2-byte fixed point without sign
- 2-byte fixed point with sign
- 2-byte floating decimal point
- 4-byte fixed point without sign

When 1 bit is selected, only the logical values "0" or "1" are processed for the input.

If byte is selected, other parameters in the ETS software are displayed. The logic object C can be compared with one or two limit values. In addition to the logic object B, another object can be parameterised for the logic object C instead of the two limit values. This object now forms the limit value. This limit value is determined through the KNX bus using a separate object. The setting of the hysteresis in percent refers symmetrically to the respective limit value. It is also parameterised when the logic object C is defined as "1". The input value in bytes is thus again converted into a 1-bit value. The logical elements can process only binary states.

The "value of limit value comparison C" can be parameterised by the ETS software after bus recovery or download.

The following options can be selected in the ETS software:

Parameter: Logic function 1 input 3

- Logic object C with one or two limit values, or setting: "Object limit value - logic operation C". (visible only if a value greater than 1 bit is set)
- Limit value comparison C limit value 1 or limit value 2, can be set depending on the selected data type. (visible only if a value greater than 1 bit is set and if this is selected with limit values)
- Hysteresis (symmetrically around the limit value/s) in percent. (visible only if a value greater than 1 bit is set)
- Determination, when limit value comparison C is defined as logical "1": Limit value comparison C = 1 if logic operation result . . . (visible only if a value greater than 1 bit is set)
- Value of limit value comparison C after bus recovery and download: Logical "0" or "1"
- Logic object C/limit value comparison C not inverted
- Logic object C/limit value comparison C inverted
- Input 3 not used

## "Logic operation" (logic 1)

The method of linking inputs 1 to 3 logically is determined in the "Logic operation" section. The following options can be selected in the ETS software:

### Parameter

Extended logic operation	
Parameter	Setting
Logic function 1	<b>OR</b>
	AND
	XOR

Parameters: Logic 1/Logic function 1

- OR (gate)
- AND (gate)
- XOR (gate)

## "Gate function" (logic 1)

The gate function affects the further signal flow of the output of the logic operation. The logic operation result can be forwarded or is sent through a gate. This gate is enabled or disabled by the logic values A, A inverted, B, B inverted, C and C inverted. The logic values deal with the processed input objects as described in the "Input processing for logic 1" chapter.

Example: Logic objects A, B and C switch the inputs 1 to 3 respectively. An OR gate is selected as a logical operation. The gate connection is enabled by the logic object A.

Result: A or B or C must be logical "1", so that the logic operation result becomes logical "1". Moreover, the logic object A must always be logical "1", so that the gate function enables the signal.

## Parameter

Extended logic operation	
Parameter	Setting
Gate function 1 control input	<b>Not used (gate open)</b>
	Logic object A
	Logic object A inverted
	Logic object B/limit value comparison B
	Logic object B inverted/limit value comparison B inverted
	Logic object C/limit value comparison C
	Logic object C inverted/limit value comparison C inverted
Gate function 1	<b>Gate function inverted (1 = lock 0 = release)</b>
	Gate function (0 = lock 1 = release)

The following options can be selected in the ETS software:

Parameter: Logic 1/gate function 1 control input

- Not used (gate open, signals are forwarded)
- Logic object A not inverted
- Logic object A inverted
- Logic object B not inverted / limit value comparison B not inverted
- Logic object B inverted/limit value comparison B inverted
- Logic object C not inverted/limit value comparison C not inverted
- Logic object C inverted/limit value comparison C inverted

The following can be parameterised in addition to the selected control input:

Parameter: Gate function 1

- Gate function inverted (1 = lock , 0 = release)
- Gate function not inverted (0 = lock, 1 = release)

## "Filter connection" (logic 1)

The filter connection influences the binary states in the signal flow. Telegrams with the logical "0" or "1" can be forwarded without check or inverted. There is another way of filtering certain telegrams.

## Parameter

Extended logic operation	
Parameter	Setting
<b>Filter 1</b>	
Filter 1	<b>1 -&gt; 1 / 0 -&gt; 0</b>
	1 -> 1 / 0 -> -
	1 -> - / 0 -> 0
	1 -> 0 / 0 -> 1
	1 -> 0 / 0 -> -
	1 -> - / 0 -> -

The following options can be selected in the ETS software:

Parameter: Logic 1/Filter 1

- 1->1 / 0->0
- 1->1 / 0->-
- 1->- / 0->0
- 1->0 / 0->1
- 1->0 / 0->-
- 1->- / 0->1

Example 1: Parameterisation 1->1 / 0->0 means, all incoming telegrams with logical "1" are displayed with logical "1" at the output of the filter connection. All incoming telegrams with logical "0" are displayed with logical "0" at the output of the filter connection.

Example 2: Parameterisation 1->1 / 0->- means, all incoming telegrams with logical "1" are displayed with logical "1" at the output of the filter connection. All incoming telegrams with logical "0" are filtered out. This means, a "0" telegram is not forwarded.

## "Output processing" (logic 1)

The following can be selected in the output processing section:

- Telegram of logic 1 switches the relay of the corresponding switching channel
- Telegram of logic 1 is sent to the KNX bus
- Telegram of logic 1 is sent to input 1 of logic 2



### Note: Output processing for the relay

Please note that the output of the logic operation (logic operation result) does not necessarily switch the relay directly. Other parameters such as central function, priority selection, etc. are also decisive for the switching state of the relay. For more details, refer to the overview plan.



### Note: Output processing for the KNX bus

Please note that the actual telegram is resent to the bus whenever the input is updated. Cyclical telegrams are thus retained.

## Parameter

Extended logic operation	
Parameter	Setting
<b>Time 1</b>	
Time 1 output	<b>Controls relay directly</b>
	Controls relay inverted
	Object logic output
	Object logic output inverted
	No reaction
Common time base	100 ms
	<b>1 s</b>
	1 Min
Factor for delay of 1-telegram	0-65536, <b>30</b>
Factor for delay of 0-telegram	0-65536, <b>30</b>

The following options can be selected in the ETS software:

Parameter: Logic 1/Output 1

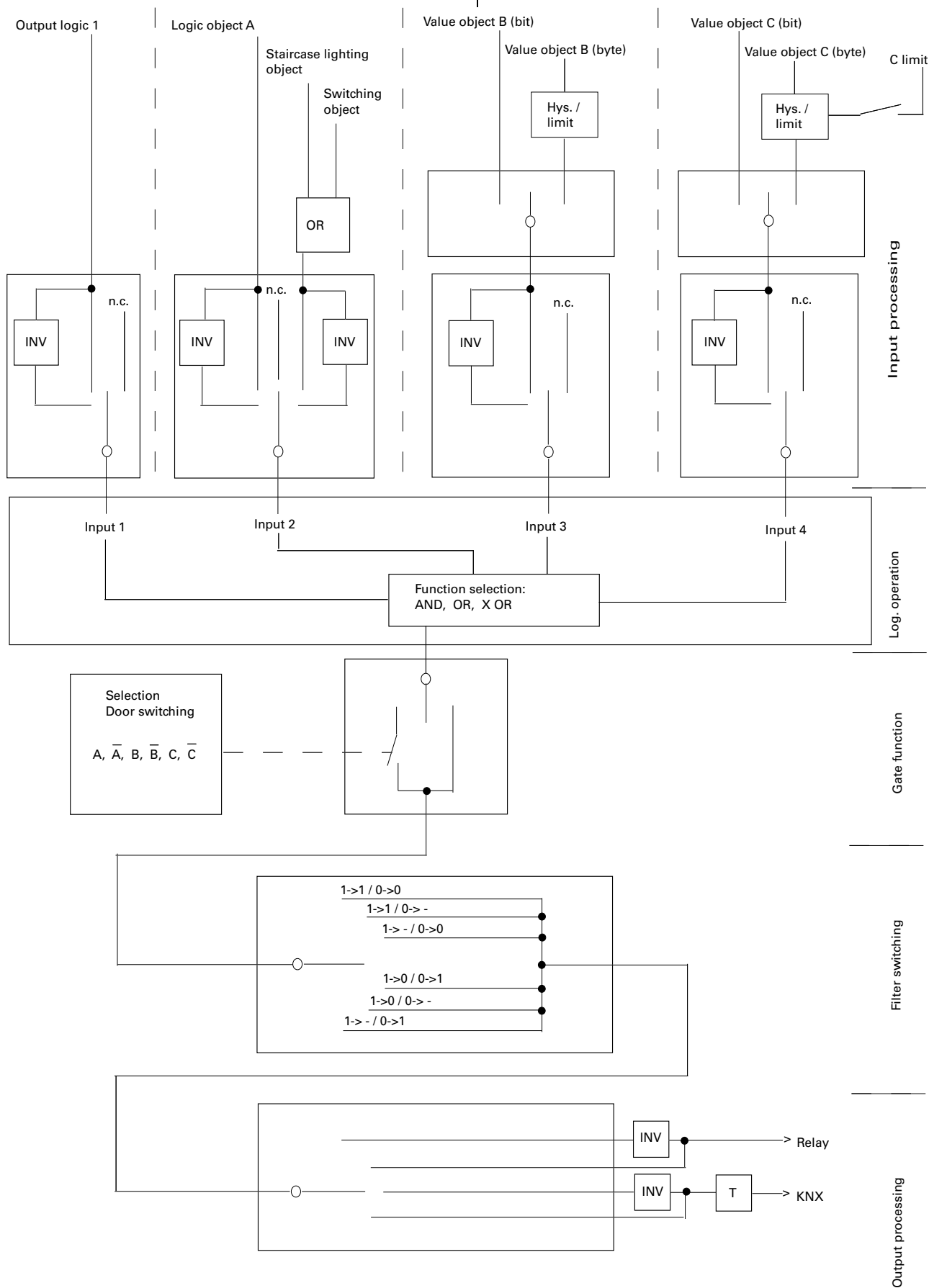
- Controls relay directly
- Controls relay inverted
- Object logic output (telegram is sent to the KNX bus)
- Object logic output inverted (telegram is sent inverted to the KNX bus)
- No reaction (direct connection to logic 2)
- A delay time can also be parameterised for logical "1" and logical "0" telegrams each.



### Note:

The time delay can be parameterised only for telegrams that are sent to the KNX bus. Switching the relay or sending information to logic 2 is performed immediately.

## Overview plan of the "Extended logic operation" as a block diagram (logic 2)



## "Input processing" (logic 2)

The input variables that affect logic 2 and the values of these objects are determined in the "input processing" section. Here it is determined, whether the input has a 1-bit value or a byte value.

The **same** communication objects as those for input processing of logic 1 are available for inputs 2 to 4. In addition to logic 1, one more input is connected to logic 2. This input is the output of logic 1, provided that the output behaviour of logic 1 has been parameterised: "to logic 2".

## "Logic object output of logic 1" (logic 2/input 1)

The value of the "Logic object output of logic 1" is 1 bit.

### Parameter

Extended logic operation	
Parameter	Setting
Logic function 2 input 1	<b>Not used</b>
	Time 1 output
	Time 1 output inverted

The following options can be selected in the ETS software:

Parameter: Logic 2/Input 1

- Time 1 output not inverted
- Time 1 output inverted
- Input 1 not used

## "Logic object A channel X, switch object channel X, Staircase lighting object channel X" (logic 2/input 2)

The "Logic object A channel X" and the "Switch object channel X" each have a value of 1 bit. The "Staircase lighting object channel X" has a value of 1 bit in case of a "fixed" staircase timer and of 2 bytes in case of a "variable" staircase timer. The switch object and the Staircase lighting object are always connected with an OR gate.

### Parameter

Extended logic operation	
Parameter	Setting
Logic function 2 input 2	<b>Not used</b>
	Switch/staircase timer object
	Switch/staircase timer object inverted
	Logic object A
	Logic object A inverted

The following options can be selected in the ETS software:

Parameter: Logic 2/Input 2

- Logic object A channel X not inverted
- Logic object A channel X inverted

- Staircase lighting object OR switch object not inverted
- Staircase lighting object OR switch object inverted
- Input 1 not used



### Note:

The staircase timer variable is a special case. This object is a 2-byte object and can also be used as an input object. The value "0" is evaluated as a logical "0" and all values greater than "0" are evaluated as logical "1".

## "Logic object B" (logic 2/input 3)

The value of the "Logic object B channel X" can be 1 bit or value of one bit or byte.

### Parameter

Extended logic operation	
Parameter	Setting
Logic object B and limit values	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point
	4-byte fixed point without sign
Logic object B with (visible only if logic object B > 1 bit)	<b>1 limit value</b>
	2 limit values
Hysteresis value object B (symmetrically around limit value 1) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object B (symmetrically around limit value 2) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison B after bus recovery and download	<b>0</b>
	1
Limit value comparison B = 1 if logic object (visible only if logic object B > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)

Limit value comparison B limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison B limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting

Extended logic operation	
Parameter	Setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 2 input 3	<b>Not used</b>
	Logic object B/limit value comparison B
	Logic object B/limit value comparison B inverted

You have the following options in the ETS software:  
Parameter: Logic 2/Input 3

- 1 bit
- 1 byte
- 2-byte fixed point without sign
- 2-byte fixed point with sign
- 2-byte floating decimal point
- 4-byte fixed point without sign

When 1 bit is selected, only the logical values "0" or "1" are processed for the input.

If byte is selected, other parameters in the ETS software are displayed. The logic object B can be compared with one or two limit values. The setting of the hysteresis in percent refers symmetrically to the respective limit value. It is also parameterised when the logic object B is defined as "1". The input value in bytes is thus again converted into a 1-bit value. The logical elements can process only binary states.

The "value of limit value comparison B" can be parameterised by the ETS software after bus recovery or download.

The following options can be selected in the ETS software:

Parameter: Logic 2/Input 3

- Logic object B with one or two limit values. (visible only if a value greater than 1 bit is set)
- Limit value comparison B limit value 1 or limit value 2, can be set depending on the selected data type. (visible only if a value greater than 1 bit is set)
- Hysteresis (symmetrically around the limit value/s) in percent. (visible only if a value greater than 1 bit is set)
- Determination, when the limit value comparison is defined as logical "1": Limit value comparison B = 1 if logic operation result . . . (visible only if a value greater than 1 bit is set)
- Value of limit value comparison B after bus recovery and download: Logical "0" or "1"
- Logic object B/limit value comparison B not inverted
- Logic object B/limit value comparison B inverted
- Input 2 not used

## "Logic object C" (logic 2/input 4)

The value of the "Logic object C channel X" can be 1 bit or byte.

### Parameter

Extended logic operation	
Parameter	Setting
Logic object C, limit values and object limit value comparison	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point
Logic object C with (visible only if logic object C > 1 bit)	4-byte fixed point without sign
	<b>1 limit value</b>
	2 limit values
Hysteresis value object C (symmetrically around limit value 1) (visible only if logic object C > 1 bit)	Object limit value logic operation C
	0-30%, <b>10 %</b>
Hysteresis value object C (symmetrically around limit value 2) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison C after bus recovery and download	<b>0</b>
	1
Limit value comparison C = 1 if logic object (visible only if logic object C > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)

Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison C limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
<b>Object limit value logic operation C</b>	
Limit value comparison C limit value 1	0-255 for the 1 byte setting

Extended logic operation	
Parameter	Setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Value of limit value logic C after bus recovery	As parameter setting
	As before bus voltage failure
Value of limit value logic C after bus recovery	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 2 input 4	<b>Not used</b>
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted
	Gate function (0 = lock 1 = release)
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted

The following options can be selected in the ETS software:

Parameter: Logic 2/Input 4

- 1 bit
- 1 byte
- 2-byte fixed point without sign
- 2-byte fixed point with sign
- 2-byte floating decimal point
- 4-byte fixed point without sign

When 1 bit is selected, only the logical values "0" or "1" are processed for the input.

If byte is selected, other parameters in the ETS software are displayed. The logic object C can be compared with one or two limit values. In addition to the logic object B, another object can be parameterised for the logic object C instead of the two limit values. This object now forms the limit value. This limit value is determined through the KNX bus using a separate object. The setting of the hysteresis in percent refers symmetrically to the respective limit value. It is also parameterised when the logic object C is defined as "1". The input value in bytes is thus again converted into a 1-bit value. The logical elements can process only binary states.

The "value of limit value comparison C" can be parameterised by the ETS software after bus recovery or download.

The following options can be selected in the ETS software:

Parameter: Logic 2/Input 4

- Logic object C with one or two limit values, or setting: "Object limit value - logic operation C". (visible only if a value greater than 1 bit is set)
- Limit value comparison C limit value 1 or limit value 2, can be set depending on the selected data type. (visible only if a value greater than 1 bit is set and if this is selected with limit values)
- Hysteresis (symmetrically around the limit value/s) in percent. (visible only if a value greater than 1 bit is set)
- Determination, when limit value comparison C is defined as logical "1": Limit value comparison C = 1 if logic operation result . . . (visible only if a value greater than 1 bit is set)
- Value of limit value comparison C after bus recovery and download: Logical "0" or "1"
- Logic object C/limit value comparison C not inverted
- Logic object C/limit value comparison C inverted
- Input 4 not used

## "Logic operation" (logic 2)

The method of linking inputs 1 to 4 logically is determined in the "Logic operation" section.

### Parameter

Extended logic operation	
Parameter	Setting
Logic function 2	<b>OR</b>
	AND
	XOR

The following options can be selected in the ETS software:

Parameter: Logic 2/Logic function 2

- OR (gate)
- AND (gate)
- XOR (gate)

## "Gate function" (logic 2)

The gate function affects the further signal flow of the output of the logic operation. The logic operation result can be forwarded or is sent through a gate. This gate is enabled or disabled by the logic values A, A inverted, B, B inverted, C and C inverted. The logic values deal with the same input objects as described in the "Input processing for logic 1" chapter.

Example: Logic objects A, B and C switch the inputs 1 to 3 respectively. An OR gate is selected as a logical operation. The gate connection is enabled by the logic object A.

Result: A or B or C must be logical "1", so that the logic operation result becomes logical "1". Moreover, the logic object A must always be logical "1", so that the gate function enables the signal.

## Parameter

Extended logic operation	
Parameter	Setting
Gate function 2 control input	<b>Not used (gate open)</b>
	Switch object A
	Switch object A inverted
	Limit value comparison B
	Limit value comparison B inverted
	Limit value comparison C
	Limit value comparison C inverted
Gate function 2	Gate function inverted (1 = lock 0 = release) Gate function (0 = lock 1 = release)

The following options can be selected in the ETS software:

Parameter: Logic 2/gate function 2 control input

- Not used (gate open, signals are forwarded)
- Logic object A not inverted
- Logic object A inverted
- Limit value comparison B not inverted
- Limit value comparison B inverted
- Limit value comparison C not inverted
- Limit value comparison C inverted

The following can be parameterised in addition to the selected control input:

Parameter: Gate function 1

- Gate function inverted (1 = lock , 0 = release)
- Gate function not inverted (0 = lock, 1 = release)

## "Output processing" (logic 2)

The following can be selected in the output processing section:

- Telegram of logic 2 switches the relay of the corresponding switching channel
- Telegram of logic 2 is sent to the KNX bus



### Note: Output processing for the relay

Please note that the output of the logic operation (logic operation result) does not necessarily switch the relay directly. Other parameters such as central function, priority selection, etc. are also decisive for the switching state of the relay. For more details, refer to the overview plan.



### Note: Output processing for the KNX bus

Please note that the actual telegram is resent to the bus whenever the input is updated. Cyclical telegrams are thus retained.

## Parameter

Extended logic operation	
Parameter	Setting
<b>Time 2</b>	
Time 2 output	<b>Controls relay directly</b>
	Controls relay inverted
	Object logic output
	Object logic output inverted
Common time base	100 ms
	<b>1 s</b>
	1 Min
Factor for delay of 1-telegram	0-65536, <b>30</b>
Factor for delay of 0-telegram	0-65536, <b>30</b>

The following options can be selected in the ETS software:

Parameter: Logic 2/Output 2

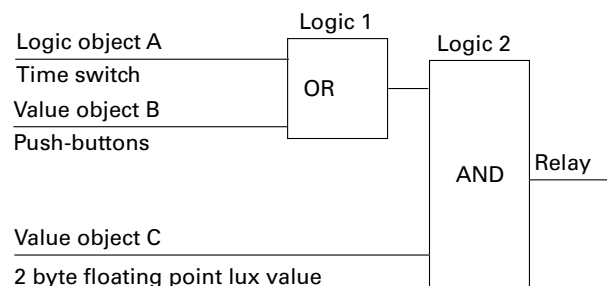
- Controls relay directly
- Controls relay inverted
- Object logic output (telegram is sent to the KNX bus)
- Object logic output inverted (telegram is sent inverted to the KNX bus)
- A delay time can also be parameterised for logical "1" and logical "0" telegrams each.

## Exemplary application for logic 1 and logic 2

Requirements:

The working hours of the employees of an assembly shop are from 7:00 hrs to 16:00 hrs. The timer keeps the lights on during this time. The lights can be switched on using a push-button if anyone enters the assembly before 7:00 hrs or after 16:00 hrs. This push-button is parameterised such that a "0" signal is sent after one hour. This means, if lights are switched on using the push-button beyond the regular working hours, they remain on for a maximum period of one hour. The lights are switched off if the external brightness exceeds 30,000 lux.

Logical elements:



### Note:

You require a KNX movement detector, a KNX timer and a lux value sensor in order to implement the aforementioned exemplary application.

The timer sends a "0" signal at 4:00 pm. The timer sends a "1" signal at 7:00 am. At this time, the external brightness is normally less than 30,000 lux; the relay thus switches on the lights. If the external brightness exceeds 30,000 lux, the AND logic operation is no longer fulfilled and the relay switches off the lights. The telegrams of the push-button are relevant in the time from 4:00 pm to 7:00 am. When pressed, the push-button sends a "1" signal. If the external brightness is less than 30,000 lux at this time, the relay switches on the lights. The relay switches off the lights if the external brightness increases, or after one hour.

Parameter settings in the ETS software:

Extended logic operation channel X

- The gate function and the filter connection are not required for this application.
- The KNX signal of the timer is linked with the logic object A (1 bit).
- The KNX signal of the push-button is linked with the value object B (1 bit).
- The aforementioned input objects are parameterised as an OR gate; this gate forms logic 1.
- Only the logic object A and value object B inputs are parameterised for logic 1.
- The output of logic 1 is forwarded to logic 2.
- The KNX signal of the weather station (lux value measurement) is linked with the value object C (2-byte floating decimal point).
- Limit value comparison C limit value 1 = 30,000 (lux)
- Limit value comparison C = logical "1" if limit value 1 shortfall (< limit value 1).
- Hysteresis: 10 %
- The value object C and the logical output of logic 1 are parameterised as an AND gate; this gate forms logic 2.
- The output of logic 2 switches the relay of the switch actuator.



**Note:**

The project planning described above refers to an exemplary application. The parameters in the ETS software must suit your system.

## Communication objects

The following communication objects can be selected:

**Per channel:**

Function	Object name	Type	Prio	Flags	Behaviour
Extended logic operation	Logic object A channel X	1 bit	Low	CW	Receive
Extended logic operation	Switch object channel X	1 bit	Low	CW	Receive
Extended logic operation	Staircase timer object fixed channel X	1 bit	Low	CW	Receive
Extended logic operation	Staircase timer object variable channel X	2 byte	Low	CW	Receive
Extended logic operation	Value object B channel X	1 bit	Low	CW	Receive
Extended logic operation	Value object C channel X	1 bit	Low	CW	Receive
Extended logic operation	Value object B channel X	1 byte	Low	CW	Receive
Extended logic operation	Value object C channel X	1 byte	Low	CW	Receive
Extended logic operation	Value object B channel X	2 byte	Low	CW	Receive
Extended logic operation	Value object C channel X	2 byte	Low	CW	Receive
Extended logic operation	Value object B channel X	4 byte	Low	CW	Receive
Extended logic operation	Value object C channel X	4 byte	Low	CW	Receive
Extended logic operation	Object limit value logic operation C	1 byte	Low	CW	Receive
Extended logic operation	Object limit value logic operation C	2 byte	Low	CW	Receive
Extended logic operation	Object limit value logic operation C	4 byte	Low	CW	Receive
Extended logic operation	Logic output object channel X	1 bit	Low	CT	Transmit

## Heating function

The current sensor actuator can be used for switching functions and as a heating actuator. The heating actuator activates electro-thermal/magnetic valve drives of heaters and cooling ceilings. One heating channel can be parameterised for every switch channel.

### Parameter

Channel configuration	
Parameter	Setting
Device selection	<b>2 channels</b>
	4 channels
	8 channels
	12 channels
Channel X operation mode	<b>Switching</b>
	Heating

### ● Summer - Winter - operation

#### Parameter

General	
Parameter	Setting
Summer/winter mode	<b>Summer = 1, Winter = 0</b>
	Summer = 0, Winter = 1

- In the "General" tab, you can parameterise the bit value at which the summer and/or winter mode should be activated.
- The summer - winter mode parameter is relevant only for heating/cooling operation.
- Connect the "Summer - Winter - General" object to a year time switch. You can also switch the summer - winter mode using a push-button.

### ● Delay time before the object "All valves are closed" is sent

#### Parameter

General	
Parameter	Setting
Object "All valves are closed" sends with a delay of ... min.	30.. <b>60</b> ..255

- In the "General" card, parameterise the delay time before the object "All valves are closed" is sent.
- For this, connect the object "All valves closed - General".

### ● Priorities of communication objects

The communication objects of the heating actuator have different priorities. The order of priority is fixed and cannot be modified in the software:

- 1st Priority: Locking object 1

- 2nd Priority: Object valve protection
- 3rd Priority: Locking object 2
- 4th Priority: All other heating objects

### ● Channel X heating

#### Reset of manual operation ( to relay state as before manual operation)

**i** The word „manipulation“ in some parameters is used instead of „manual operation“ during the restricted length of the parameter wording.

The "Reset manipulation (relay state as before manual operation)" parameter is relevant only in connection with the general "Automatic reset of manual operation to the current object value" parameter.

#### Parameter

General: Automatic reset of manual operation to the current object value	
Parameter	Setting
Time until reset 1 s x factor (0-65535, 0 = no reset)	<b>0</b> - 65535

Channel X heating	
Parameter	Setting
Reset manipulation (relay state as before manipulation)	<b>Disabled</b>
	After a time
	If object Reset manual operation = 1

- Reset manipulation (manual operation):  
After a time (the duration is parameterised in the "General" tab).
- Reset manipulation (manual operation):  
If object Reset manual operation = 1 (if the manual operation object - general is switched with logic "1")

### Channel-related parameters for heating:

The following selection can be made in the ETS software in the relevant channel of the heating function:

#### Parameter

Channel X heating	
Parameter	Setting
Control output for heating object	<b>1 byte</b>
	1 bit

- Select the object type in this parameter:  
1 bit or 1 byte.
- Connect the "Channel X heater on-off" to the relevant controller.

## Parameter

Channel X heating	
Parameter	Setting
Valve status 1 bit	<b>Disabled</b> Active status feedback object Passive status object
Valve status 1 byte	<b>Disabled</b> Active status feedback object Passive status object
Status/status response object	<b>At bus and manual operation change</b> At bus change At manual operation change
Status/status response object	<b>Pressed = 1; released= 0</b> Pressed = 0; released = 1

- The 1 bit status object can be parameterised only if 1 bit has been selected as the control output for heating.
- The 1 bit/1 byte valve status can be disabled or can be parameterised as an active status feedback object or a passive status object.
- If the status/status feedback object is active, you can parameterise the change that transmits the object.
- As a default setting, logical value "1" for pressed and logical value "0" for released is transmitted to the bus as a status/status feedback object. This value can be inverted.

## Parameter

Channel X heating	
Parameter	Setting
Valve type	<b>De-energized closed</b> De-energized opened

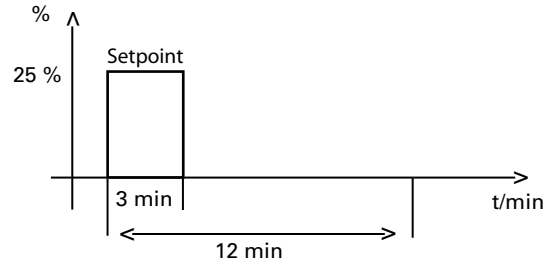
- Set the correct valve type depending on the connected valve.

## Parameter

Channel X heating	
Parameter	Setting
PWM cycle time (1 min x factor)	1 - 60, <b>15</b>

In PWM regulation, the control values calculated by the controller (e.g. multifunction push-button with RTR) are converted into pulse-width modulation (PWM). Within a parameterisable cycle time, the control actuator is opened ("1") and then closed again ("0") for the calculated percentage duration. For example, when a control value of 25% has been calculated and a cycle time of 12 minutes parameterised, a "1" is transmitted at the beginning of the cycle time and a "0" is transmitted after three minutes (= 25% of 12 minutes).

tes) have elapsed.  
Example:



## Parameter

Channel X heating	
Parameter	Setting
Valve protection (prio. 2)	
Activate valve protection	<b>Cyclic</b> Never With telegram Cyclic and with telegram
Cycle time (1 day x factor)	1 - 255, <b>10</b>
Duration (1 min x factor)	1 - 255, <b>3</b>
Valve status	<b>No reaction</b> Active status response object Passive status response object

- A valve protection can be activated so that the valves do not gum even if they are not used for a long time.
- Valve protection - cyclical: Set the cycle time and the duration of the open valve.
- Valve protection - with telegram: The valve protection is triggered via a separate object (1 bit) (start channel X for valve protection).
- A status/status feedback object can be transmitted as soon as valve protection is triggered.

## Parameter

Channel X heating	
Parameter	Setting
Failure behaviour	
Valve position in case of bus voltage failure	<b>No reaction</b> Opened Closed
Valve position upon bus voltage recovery	<b>Unchanged</b> Opened Closed

- The failure behaviour of the connected valve can be parameterised for bus voltage failure and bus voltage recovery.

## Parameter

Channel X heating	
Parameter	Setting
Locking behaviour	
Lock 1 (prio. 1)	<b>For object value "1"</b> For object value "0"
Lock 1 valve position for summer mode in %	0 - 100%, <b>30</b>
Lock 1 valve position for winter mode in %	0 - 100%, <b>50</b>
Lock 2 (prio. 3)	<b>For object value "1"</b> For object value "0"
Lock 2 valve position in summer mode in %	0 - 100%, <b>30</b>
Lock 2 valve position in winter mode in %	0 - 100, <b>50</b>

- Two locking objects can be activated for every heating channel.
- The object value (logical "1" or "0") at which the relevant lock function should be activated can be parameterised.
- Locking object 1 has a higher priority than locking object 2.
- A valve position for the summer and winter mode can be parameterised for each locking behaviour.

## Parameter

Channel X heating	
Parameter	Setting
Cyclical monitoring of the control output	
Send alarm	<b>Disabled</b> For missing cyclical telegrams
Surveillance time 1 min x factor (0 = off)	1-255, <b>30</b>
Send emergency mode object cyclically 1 min x factor (0= no cyclical sending)	<b>0-255</b>
Valve position at exceeding in cycle time in the summer mode	0 - 100%, <b>50</b>
Valve position at exceeding in cycle time in the winter mode	0 - 100%, <b>30</b>

- The heating actuator can monitor the cyclical control output of the controller (e.g. multifunction push-button with RTR).
- If cyclical transmission of the control output is interrupted, an alarm can be sent to the bus either cyclically or just once via the "Channel X heating alarm" object.
- The monitoring time of the cyclical control output can be parameterised to between 1 minute and 255 minutes.
- The valve position for the summer and winter mode can be parameterised if the cycle of the control output is exceeded.

## Communication objects

The following communication objects can be selected:

### Cross-channel:

Function	Object name	Type	Prio	Flags	Behaviour
General	Summer - Winter - operation	1 bit	Low	CW	Receive
General	Reset manual operation	1 bit	Low	CW	Receive
General	All valves are closed	1 bit	Low	CRT	Read/send

### Per channel:

Function	Object name	Type	Prio	Flags	Behaviour
Heating	Heating on/off - channel X	1 bit	Low	CW	Receive
Heating	Continuous heating - channel X	1 byte	Low	CW	Receive
Heating	Lock 1 heating - channel X	1 bit	Low	CW	Receive
Heating	Lock 2 heating - channel X	1 bit	Low	CW	Receive
Heating	Start valve protection - channel X	1 bit	Low	CW	Receive
Heating	Feedback heating on/off - channel X	1 bit	Low	CR	Read
Heating	Feedback continuous heating - channel X	1 byte	Low	CR	Read
Heating	Heating alarm - channel X	1 bit	Low	CRT	Read/send
Heating	Valve protection feedback - channel X	1 bit	Low	CT	Transmit

## Overview of parameters

### ● Overview of switching parameters

Channel configuration	
Parameter	Setting
Device selection	<b>2 channels</b>
	4 channels
	8 channels
	12 channels
Channel X operation mode	<b>Switching</b>
	Heating

Channel X: Switching	
Parameter	Setting
Relay operation	<b>Make contact</b>
	Break contact
	Blinking

Channel X: Blinking (if the "Blinking" relay mode is selected)	
Parameter	Setting
Behaviour at pressed/released	<b>Blinking/relay opened</b>
	Blinking/relay closed
	Relay opened/Blinking
	Relay closed/Blinking
Basic Blinking interval	<b>1 s</b>
	1 min
Blinking interval factor	1-255, <b>5</b>
	Can be set in single steps
Ratio of closed/opened	10% / 90%
	20% / 80%
	30% / 70%
	40% / 60%
	<b>50% / 50%</b>
	60% / 40%
	70% / 30%
	80% / 20%
	90% / 10%
Blinking begins with	<b>Relay is closed</b>
	Relay is opened
Defined number of flash impulses (0 = permanent Blinking)	0..100, <b>20</b>
Status after running the defined number of flash impulses	<b>Relay is closed</b>
	Relay is opened

General	
Parameter	Setting
Central function	<b>Disabled</b>
	Enabled
	Enabled/save changes
	Delayed/save changes

General (if "Delayed/save changes" is selected)	
Parameter	Setting
Delay of central function all channels 100 ms* factor (0-65535)	<b>0-65535</b>
Time between central functions per channel 100 ms* factor (0-255)	0-255, <b>2</b>

Channel X: Switching	
Parameter	Setting
Central function	<b>Disabled</b>
	Enabled

Channel X: Central function (if "Enabled" is selected in the central function for the channel)	
Parameter	Setting
Overwrite runtime settings on download	<b>Disabled</b>
	Enabled
Central function (if "Enabled" is selected for "Overwrite runtime settings on download")	<b>No reaction at "0" or "1"</b>
	Released at "0" or "1"
	Pressed at "0" or "1"
	Pressed at "0" / released at "1"
	Released at "0" / pressed at "1"
	Released at "0" / no reaction at "1"
	Pressed at "0" / no reaction at "1"
	No reaction at "0" / released at "1"
	No reaction at "0" / pressed at "1"

Channel X: Switching	
Parameter	Setting
On-Off delay time	<b>Disabled</b>
	Enabled

Channel X: On- Off delay time	
Parameter	Setting
<b>ON delay</b>	
Affects	<b>None of the objects</b>
	Switch object
	Staircase lighting time object
	Scene object
	Switch and staircase lighting object
	Switch and scene object
	Staircase lighting and scene object
	Switch, staircase lighting and scene object
ON delay	<b>Not retriggerable</b>
	Retriggerable
Time base for ON delay	100 ms
	<b>1 s</b>
	1 min
Factor for on delay time (1-255)	1-255, <b>3</b>
<b>OFF delay</b>	

Channel X: On- Off delay time	
Parameter	Setting
Affects	<b>None of the objects</b>
	Switch object
	Staircase lighting time object
	Scene object
	Switch and staircase lighting object
	Switch and scene object
	Staircase lighting and scene object
OFF delay	Switch, staircase lighting and scene object
	<b>Not retriggerable</b>
	Retriggerable
Max. number of additions (if "Retriggerable and adding" Off delay time is selected)	Retriggerable and adding
	2
	3
	4
Time base for OFF delay	<b>5</b>
	100 ms
	<b>1 s</b>
Factor for Off delay time (1-255)	1 min
	1-255, <b>120</b>

Channel X: Switching	
Parameter	Setting
Staircase lighting time	<b>Disabled</b>
	Staircase lighting time fixed
	Staircase lighting time variable

Channel X: Staircase lighting time (fixed)	
Parameter	Setting
Staircase lighting time	<b>With anticipated truncation (0-telegram)</b>
	Without early cancellation
Staircase lighting time is	<b>Not retriggerable</b>
	Retriggerable
	Retriggerable and adding
	Retrigger to the higher value (only stair. timer variable)
Max. number of additions (if "Retriggerable and adding" staircase lighting time is selected)	2
	3
	4
	<b>5</b>
Time base staircase lighting time fixed	1 s
	<b>1 min</b>
Factor staircase lighting time (1-65535)	1-65535, <b>3</b>
Number of warnings before end of staircase lighting time	<b>0</b>
	1
	2
	3
Warning starts at a staircase lighting rest time of (1 s * factor)	1-255, <b>30</b>

Channel X: Staircase lighting time (variable)	
Parameter	Setting
Staircase lighting time	<b>With anticipated truncation (0-telegram)</b>
	Without early cancellation
Staircase lighting time is	<b>Not retriggerable</b>
	Retriggerable
	Retriggerable and adding
	Retrigger to the higher value (only stair. timer variable)
Max. number of additions (if "Retriggerable and adding" staircase lighting time is selected)	2
	3
	4
	<b>5</b>
Base time for staircase lighting time variable object	1 s
	<b>1 min</b>
Number of warnings before end of staircase lighting time	<b>0</b>
	1
	2
	3
Warning starts at a staircase lighting rest time of (1 s * factor)	1-255, <b>30</b>

Channel X: Switching	
Parameter	Setting
Scenes	<b>Disabled</b>
	Enabled

Channel X: Scene	
Parameter	Setting
Override scene values in actuator on download	<b>Disabled</b>
	Enabled
Time delay for the actuator channel (base 100 ms)	<b>0-255 (0 = disabled)</b>
Scene 1	<b>Disabled</b>
	Enabled
Scene 1 scene address (0-63) (if "Enabled" is selected for scene 1)	0-63, <b>0</b>
Scene 1 switching state (if "Enabled" is selected for scene 1)	<b>Released</b>
	Pressed
Scene 2	<b>Disabled</b>
	Enabled
Scene 2 scene address (0-63) (if "Enabled" is selected for scene 2)	0-63, <b>1</b>
Scene 2 switching state (if "Enabled" is selected for scene 2)	<b>Released</b>
	Pressed
Scene 3	<b>Disabled</b>
	Enabled
Scene 3 scene address (0-63) (if "Enabled" is selected for scene 3)	0-63, <b>2</b>
Scene 3 switching state (if "Enabled" is selected for scene 3)	<b>Released</b>
	Pressed
Scene 4	<b>Disabled</b>
	Enabled
Scene 4 scene address (0-63) (if "Enabled" is selected for scene 4)	0-63, <b>3</b>
Scene 4 switching state (if "Enabled" is selected for scene 4)	<b>Released</b>
	Pressed

Channel X: Scene	
Parameter	Setting
Scene 5	<b>Disabled</b> Enabled
Scene 5 scene address (0-63) (if "Enabled" is selected for scene 5)	0-63, <b>4</b>
Scene 5 switching state (if "Enabled" is selected for scene 5)	<b>Released</b> Pressed
Scene 6	<b>Disabled</b> Enabled
Scene 6 scene address (0-63) (if "Enabled" is selected for scene 6)	0-63, <b>5</b>
Scene 6 switching state (if "Enabled" is selected for scene 6)	<b>Released</b> Pressed
Scene 7	<b>Disabled</b> Enabled
Scene 7 scene address (0-63) (if "Enabled" is selected for scene 7)	0-63, <b>6</b>
Scene 7 switching state (if "Enabled" is selected for scene 7)	<b>Released</b> Pressed
Scene 8	<b>Disabled</b> Enabled
Scene 8 scene address (0-63) (if "Enabled" is selected for scene 8)	0-63, <b>7</b>
Scene 8 switching state (if "Enabled" is selected for scene 8)	<b>Released</b> Pressed

Per channel: Select the Locking function.

Channel X: Switching	
Parameter	Setting
Higher priority function	<b>Locking operation</b> Priority function

Per channel: Set the Lock function.

Channel X: Disable	
Parameter	Setting
Lock function	<b>At object value "0"</b> At object value "1"
Behaviour at start of Locking	No reaction Pressed <b>Released</b>
Behaviour at start of Locking	No reaction Follows currently valid status Pressed <b>Released</b>
Locking behaviour at download (only if a higher priority function is inactive)	<b>Disabled</b> Enabled As before bus voltage failure
Behaviour of the locking after bus voltage recovery	<b>Disabled</b> Enabled As before bus voltage failure

Per channel: Select the priority function.

Channel X: Switching	
Parameter	Setting
Higher priority function	Locking operation <b>Priority function</b>

Per channel: Set the behaviour of the priority function.

Channel X: Disable	
Parameter	Setting
Behaviour after end of priority function	Follows currently value Pressed <b>Released</b>
Priority function behaviour after bus voltage recovery	<b>Disabled</b> Enabled, released Enabled, pressed As before bus voltage failure

General	
Parameter	Setting
Delay status feedback all channels, 100 ms * factor (0-65535)	<b>0-65535</b>
Time between status feedback per channel, 100 ms * factor (0-255)	0,255, <b>2</b>

Channel X: Switching	
Parameter	Setting
Status/status response	<b>Disabled</b> Active status response object Passive status object
Status/status response object	<b>For bus and manual mode change</b> For bus change For manual mode change
Value status/status response object	<b>Pressed = 1, Released = 0</b> Pressed = 0, Released = 1

General	
Parameter	Setting
Collected status/status response	<b>Disabled</b> Status response object Status object
Collected status/status response object	<b>At changes by bus and manipulation</b> At changes by bus At changes by manipulation
Assignment channel status to bit value	<b>Pressed = 1, Released = 0</b> Pressed = 0, Released = 1
Delay time sending (1 s * factor; 0-255) (if "Status response object" is selected for collected status/status response)	0-255, <b>60</b>
Behaviour after bus voltage recovery	<b>No reaction</b> Sends current status

Channel X: Switching	
Parameter	Setting
Failure behaviour	<b>Disabled</b>
	Enabled

General	
Parameter	Setting
Sending delay after bus voltage recovery in s	0-255, <b>30</b>

Channel X failure behaviour	
Parameter	Setting
Relay state after bus voltage failure	<b>No reaction</b>
	Pressed
	Released

Channel X failure behaviour	
Parameter	Setting
Relay state after bus voltage recovery	<b>No reaction</b>
	Pressed
	Released
	As for bus voltage failure
Status response behaviour after bus voltage recovery	<b>No reaction</b>
	Sends curr. status

Channel X: General	
Parameter	Setting
Relay status after download	<b>No reaction</b>
	Pressed
	Released
	As before download

## ● Parameter of safty function

General	
Parameter	Setting
Safety function	<b>Disabled</b>
	Safety at "1"
	Safety at "0"
Safety function	Enabled
	<b>Disabled</b>
Behaviour at safety state	<b>No reaction</b>
	Pressed
	Released
	Blinking, 1 s frequency
Behaviour at end of safety state	<b>Follows the currently valid status</b>
	Pressed
	Released
	No reaction
Behaviour at exceeding cycle time	<b>No reaction</b>
	Pressed
	Released
	Blinking, 1 s frequency

## ● Parameter of manual mode

General	
Parameter	Setting
Autom. reset of manual operation to actual object value	
Time until reset 1s*factor (0-65535, 0=not reset)	0 - 65535
Status response manual operation	<b>Disabled</b>
	At manual mode (A)
	At failed autom. reset (B)
	At (A) OR (B)

Channel X:	
Parameter	Setting
Reset manipulation	<b>Disabled</b>
	After time
	If Reset manual operation object = 1

## ● Parameter of current detection

Current detection	
Parameter	Setting
Current type	<b>Alternating current</b>
	Direct current

Current detection	
Parameter	Setting
Object current value	<b>1 byte (base = 100 mA)</b>
	2 byte fixed decimal (base = 1mA)
	2 byte floating point
Transmitting the current value	<b>Not (read only)</b>
	At changes
	Cyclic
	Cyclic and at changes
At change of	10%.... <b>20%</b> ....50%
Base cyclical interval	0.1 s to <b>1.0 s</b>
Cyclical interval factor	2 to <b>10</b> to 65535

Current detection	
Parameter	Setting
Analysis	No
	With 1 limit value
	With two limit values
Limit valueX	
Limit valueX 1mA * factor	200 to <b>1000</b> to 65535
Hysteresis, 1mA * factor	100 to <b>200</b> to 8000
Delay of analysis 100ms * factor	0 to <b>5</b> to 255
Reaction if exceeded Device/Object status current limit valueX	None
	Released/None
	None/"0"
	None/"1"
	Released/"0"
	Released/"1"

Current detection	
Parameter	Setting
Reaction if undercut	None
Device/Object status current limit valueX	Released/None
	None/"0"
	None/"1"
	Released/"0"
	Released/"1"

Current detection	
Parameter	Setting
Voltage for calculation 1V * factor	12 to <b>230</b> to 253
Transmitting consumption	<b>Not (read only)</b>
	All 1 kWh
	Cyclic
	Cyclic and all 1 kWh
Base cyclical interval	1 min to <b>1h</b>
Cyclical interval factor	<b>1</b> to 255
Limit (at limit reset consumption to 0) 1kWh * Faktor (0-65535)	<b>0</b> ...65535
Format energy consumption	Wh
	Ws (J)

## ● Parameter of counter

Counter	
Parameter	Setting
Counter	<b>None</b>
	Switch counter
	Operating hours counter
	Switch and operating hours counter
	Combined counter (= 1 value)
Object switch counter	<b>4 byte fixed decimal</b>
	2 byte fixed decimal
Switch counter condition	<b>if relay is closed</b>
	if (current value) limit is exceeded
Switch counter limit	<b>0 to 65536</b>
Object operating hours counter	<b>4 byte fixed decimal</b>
	2 byte fixed decimal
Operating hours counter condition	<b>if relay is closed</b>
	if (current value) limit 1 is exceeded
Operating hours counter limit	<b>0 to 65536</b>
Combined counter limit (combined counter=x*switching+y*hours)	<b>0 to 999999</b>
Factor x	0 to 255
Factor y	0 to 255

## ● Parameter of logic

Switching channel X	
Parameter	Setting
Logic function	<b>Disabled</b>
	Logic operation
	Extended logic operation

Logic operation: channel X	
Parameter	Setting
Type of logic operation	<b>OR</b>
	AND
	XOR

Base logic function channel X	
Parameter	Setting
Logic object A	<b>Not inverted</b>
	Inverted
Value of logic object A after bus voltage recovery and download	<b>0</b>
	1

Extended logic operation	
Parameter	Setting
Value of logic object A after bus voltage recovery and download	<b>0</b>
	1

Extended logic operation	
Parameter	Setting
Logic function 1 input 1	<b>Not used</b>
	Switch/staircase timer object
	Switch/staircase timer object inverted
	Logic object A
	Logic object A inverted

Extended logic operation	
Parameter	Setting
Logic object B and limit values	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point
	4-byte fixed point without sign
Logic object B with (visible only if logic object B > 1 bit)	<b>1 limit value</b>
	2 limit values
Hysteresis value object B (symmetrically around limit value 1) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object B (symmetrically around limit value 2) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison B after bus recovery and download	<b>0</b>
	1
Limit value comparison B = 1 if logic object (visible only if logic object B > 1 bit)	> Limit value 1 (limit value exceeded)

Extended logic operation	
Parameter	Setting
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND <limit value 2 (range maintained)
Limit value comparison B limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison B limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 1 input 2	<b>Not used</b>
	Logic object B/limit value comparison B
	Logic object B/limit value comparison B inverted

Extended logic operation	
Parameter	Setting
Logic object C, limit values and object limit value comparison	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point
	4-byte fixed point without sign
Logic object C with (visible only if logic object C > 1 bit)	<b>1 limit value</b>
	2 limit values
	Object limit value logic operation C
Hysteresis value object C (symmetrically around limit value 1) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object C (symmetrically around limit value 2) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison C after bus recovery and download	<b>0</b>
	1
Limit value comparison C = 1 if logic object (visible only if logic object C > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)

Extended logic operation	
Parameter	Setting
	> Limit value 1 AND <limit value 2 (range maintained)
Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison C limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting

Object limit value logic operation C	
Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Value of limit value logic operation C after bus voltage recovery	Like parameter setting
	As before bus voltage failure
Value of limit value logic operation C after bus voltage recovery	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 1 input 3	<b>Not used</b>
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted
	Gate function (0 = lock 1 = release)
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted

Extended logic operation	
Parameter	Setting
Logic function 1	<b>OR</b>
	AND
	XOR

Extended logic operation	
Parameter	Setting
Gate function 1 control input	<b>Not used (gate open)</b>
	Logic object A
	Logic object A inverted
	Logic object B/limit value comparison B
	Logic object B inverted/limit value comparison B inverted
	Logic object C/limit value comparison C
	Logic object C inverted/limit value comparison C inverted
Gate function 1	<b>Gate function inverted (1 = lock 0 = release)</b>
	Gate function (0 = lock 1 = release)

Extended logic operation	
Parameter	Setting
<b>Filter 1</b>	
Filter 1	<b>1 -&gt; 1 / 0 -&gt; 0</b>
	1 -> 1 / 0 -> -
	1 -> - / 0 -> 0
	1 -> 0 / 0 -> 1
	1 -> 0 / 0 -> -
	1 -> - / 0 -> -

Extended logic operation	
Parameter	Setting
<b>Time 1</b>	
Time 1 output	<b>Controls relay directly</b>
	Controls relay inverted
	Object logic output
	Object logic output inverted
	No reaction
Common time base	100 ms
	<b>1 s</b>
	1 Min
Factor for delay of 1-telegram	0-65536, <b>30</b>
Factor for delay of 0-telegram	0-65536, <b>30</b>

Extended logic operation	
Parameter	Setting
Logic function 2 input 1	<b>Not used</b>
	Time 1 output
	Time 1 output inverted

Extended logic operation	
Parameter	Setting
Logic function 2 input 2	<b>Not used</b>
	Switch/staircase timer object
	Switch/staircase timer object inverted
	Logic object A
	Logic object A inverted

Extended logic operation	
Parameter	Setting
Logic object B and limit values	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point
	4-byte fixed point without sign
Logic object B with (visible only if logic object B > 1 bit)	<b>1 limit value</b>
	2 limit values
Hysteresis value object B (symmetrically around limit value 1) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object B (symmetrically around limit value 2) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison B after bus recovery and download	<b>0</b>
	1
Limit value comparison B = 1 if logic object (visible only if logic object B > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)

Limit value comparison B limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison B limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 2 input 3	<b>Not used</b>
	Logic object B/limit value comparison B
	Logic object B/limit value comparison B inverted

Extended logic operation	
Parameter	Setting
Logic object C, limit values and object limit value comparison	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point

Extended logic operation	
Parameter	Setting
Logic object C with (visible only if logic object C > 1 bit)	4-byte fixed point without sign
	<b>1 limit value</b>
	2 limit values
	Object limit value logic operation C
Hysteresis value object C (symmetrically around limit value 1) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object C (symmetrically around limit value 2) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison C after bus recovery and download	<b>0</b>
	1
Limit value comparison C = 1 if logic object (visible only if logic object C > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)

Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison C limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
<b>Object limit value logic operation C</b>	
Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Value of limit value logic C after bus recovery	As parameter setting
Value of limit value logic C after bus recovery	As before bus voltage failure
	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting

Extended logic operation	
Parameter	Setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
	<b>Not used</b>
Logic function 2 input 4	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted
	Gate function (0 = lock 1 = release)
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted

Extended logic operation	
Parameter	Setting
Logic function 2	<b>OR</b>
	AND
	XOR

Extended logic operation	
Parameter	Setting
Gate function 2 control input	<b>Not used (gate open)</b>
	Switch object A
	Switch object A inverted
	Limit value comparison B
	Limit value comparison B inverted
	Limit value comparison C
	Limit value comparison C inverted
Gate function 2	Gate function inverted (1 = lock 0 = release)
	Gate function (0 = lock 1 = release)

Extended logic operation	
Parameter	Setting
<b>Time 2</b>	
Time 2 output	<b>Controls relay directly</b>
	Controls relay inverted
	Object logic output
	Object logic output inverted
Common time base	100 ms
	<b>1 s</b>
	1 Min
Factor for delay of 1-telegram	0-65536, <b>30</b>
Factor for delay of 0-telegram	0-65536, <b>30</b>

Extended logic operation	
Parameter	Setting
Logic object B and limit values	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign

Extended logic operation	
Parameter	Setting
	2-byte floating decimal point
	4-byte fixed point without sign
Logic object C, limit values and object limit value comparison	<b>1 bit</b>
	1 byte
	2-byte fixed point without sign
	2-byte fixed point with sign
	2-byte floating decimal point
	4-byte fixed point without sign
Logic object B with (visible only if logic object B > 1 bit)	<b>1 limit value</b>
	2 limit values
Logic object C with (visible only if logic object C > 1 bit)	<b>1 limit value</b>
	2 limit values
	Object limit value logic operation C
Hysteresis value object B (symmetrically around limit value 1) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object B (symmetrically around limit value 2) (visible only if logic object B > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object C (symmetrically around limit value 1) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Hysteresis value object C (symmetrically around limit value 2) (visible only if logic object C > 1 bit)	0-30%, <b>10 %</b>
Value of limit value comparison B after bus recovery and download	<b>0</b>
	1
Value of limit value comparison C after bus recovery and download	<b>0</b>
	1
Limit value comparison B = 1 if logic object (visible only if logic object B > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)
Limit value comparison C = 1 if logic object (visible only if logic object C > 1 bit)	> Limit value 1 (limit value exceeded)
	< Limit value 1 (limit value shortfall)
	> Limit value 2 OR < limit value 1 (exceed/fall short of range)
	> Limit value 1 AND < limit value 2 (range maintained)
Limit value comparison B limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting

Extended logic operation	
Parameter	Setting
Limit value comparison B limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Limit value comparison C limit value 2	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
<b>Object limit value logic operation C</b>	
Limit value comparison C limit value 1	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Value of limit value logic operation C after bus reset	0-255 for the 1 byte setting
	0-65536 for the 2-byte fixed point without sign setting
	-32767 - 32767 for the 2-byte fixed point with sign setting
	-32767 - 32767 for the 2-byte floating decimal point setting
	0-4294967295 for the 4-byte fixed point without sign setting
Logic function 1 input 1	<b>Not used</b>
	Switch/staircase timer object
	Switch/staircase timer object inverted
	Logic object A
	Logic object A inverted
Logic function 1 input 2	<b>Not used</b>
	Logic object B/limit value comparison B
	Logic object B/limit value comparison B inverted
Logic function 1 input 3	<b>Not used</b>
	Logic object C/limit value comparison C

Extended logic operation	
Parameter	Setting
	Logic object C/limit value comparison C inverted
Logic function 1	<b>OR</b>
	AND
	XOR
<b>Gate 1 -&gt; Filter 1 -&gt; Time 1 Gate 1</b>	
Gate function 1 control input	<b>Not used (gate open)</b>
	Switch object A
	Switch object A inverted
	Limit value comparison B
	Limit value comparison B inverted
	Limit value comparison C
	Limit value comparison C inverted
Gate function 1	<b>Gate function inverted (1 = lock 0 = release)</b>
	Gate function (0 = lock 1 = release)
<b>Filter 1</b>	
Filter 1	<b>1 -&gt; 1 / 0 -&gt; 0</b>
	1 -> 1 / 0 -> -
	1 -> - / 0 -> 0
	1 -> 0 / 0 -> 1
	1 -> 0 / 0 -> -
	1 -> - / 0 -> -
	1 -> - / 0 -> -
<b>Time 1</b>	
Common time base	100 ms
	<b>1 s</b>
	1 Min
Factor for delay of 1-telegram	0-65536, <b>30</b>
Factor for delay of 0-telegram	0-65536, <b>30</b>

Time 1 output	<b>Controls relay directly</b>
	Controls relay inverted
	Object logic output
	Object logic output inverted
	No reaction
Logic function 2 input 1	<b>Not used</b>
	Switch/staircase timer object
	Switch/staircase timer object inverted
	Logic object A
	Logic object A inverted
Logic function 2 input 2	<b>Not used</b>
	Logic object B/limit value comparison B
	Logic object B/limit value comparison B inverted
Logic function 2 input 3	<b>Not used</b>
	Logic object C/limit value comparison C
	Logic object C/limit value comparison C inverted
Logic function 2 input 4	<b>Not used</b>
	Time 1 output

Extended logic operation	
Parameter	Setting
	Time 1 output inverted
Logic function 2	<b>OR</b>
	AND
	XOR
<b>Gate 2 -&gt; Filter 2 -&gt; Time 2 Gate 2</b>	
Gate function 2 control input	<b>Not used (gate open)</b>
	Switch object A
	Switch object A inverted
	Limit value comparison B
	Limit value comparison B inverted
	Limit value comparison C
	Limit value comparison C inverted
Gate function 2	<b>Gate function inverted (1 = lock 0 = release)</b>
	Gate function (0 = lock 1 = release)
<b>Filter 2</b>	
Filter 2	<b>1 -&gt; 1 / 0 -&gt; 0</b>
	1 -> 1 / 0 -> -
	1 -> - / 0 -> 0
	1 -> 0 / 0 -> 1
	1 -> 0 / 0 -> -
	1 -> - / 0 -> -
	1 -> - / 0 -> -
<b>Time 2</b>	
Common time base	100 ms
	<b>1 s</b>
	1 Min
Factor for delay of 1-telegram	0-65536, <b>30</b>
Factor for delay of 0-telegram	0-65536, <b>30</b>

Time 2 output	<b>Controls relay directly</b>
	Controls relay inverted
	Object logic output
	Object logic output inverted

## ● Parameter of heating

Channel configuration	
Parameter	Setting
Device selection	<b>2 channels</b>
	4 channels
	8 channels
	12 channels
Channel X operation mode	<b>Switching</b>
	Heating

General	
Parameter	Setting
Summer/winter mode	<b>Summer = 1, Winter = 0</b>
	Summer = 0, Winter = 1

General	
Parameter	Setting
Time delay before all valves closed in minutes	30.. <b>60</b> ..255

General: Automatic reset of manual operation to the current object value	
Parameter	Setting
Time until reset 1 s x factor (0-65535, 0 = no reset)	<b>0</b> - 65535

Channel X heating	
Parameter	Setting
Reset manipulation (relay state as before manipulation)	<b>Disabled</b>
	After a time
	If object Reset manual operation = 1

Channel X heating	
Parameter	Setting
Control output for heating object	<b>1 byte</b>
	1 bit
Valve status 1 bit	<b>Disabled</b>
	Active status feedback object
	Passive status object
Valve status 1 byte	<b>Disabled</b>
	Active status feedback object
	Passive status object
Status/status response object	<b>At bus and manual operation change</b>
	At bus change
	At manual operation change
Status/status response object	<b>Pressed = 1; released= 0</b>
	Pressed = 0; released = 1
Valve type	<b>De-energized closed</b>
	De-energized opened
PWM cycle time (1 min x factor)	1 - 60, <b>15</b>
Valve protection (prio. 2)	
Activate valve protection	<b>Cyclic</b>
	Never
	With telegram

Channel X heating	
Parameter	Setting
	Cyclic and with telegram
Cycle time (1 day x factor)	1 - 255, <b>10</b>
Duration (1 min x factor)	1 - 255, <b>3</b>
Valve status	<b>No reaction</b>
	Active status response object
	Passive status response object
Failure behaviour	
Valve position in case of bus voltage failure	<b>No reaction</b>
	Opened
	Closed
Valve position upon bus voltage recovery	<b>Unchanged</b>
	Opened
	Closed
Locking behaviour	
Lock 1 (prio. 1)	<b>For object value "1"</b>
	For object value "0"
Lock 1 valve position for summer mode in %	0 - 100%, <b>30</b>
Lock 1 valve position for winter mode in %	0 - 100%, <b>50</b>
Lock 2 (prio. 3)	<b>For object value "1"</b>
	For object value "0"
Lock 2 valve position in summer mode in %	0 - 100%, <b>30</b>
Lock 2 valve position in winter mode in %	0 - 100, <b>50</b>
Cyclical monitoring of the control output	
Send alarm	<b>Disabled</b>
	For missing cyclical telegrams
Surveillance time 1 min x factor (0 = off)	1-255, <b>30</b>
Send emergency mode object cyclically 1 min x factor (0= no cyclical sending)	<b>0-255</b>
Valve position at exceeding in cycle time in the summer mode	0 - 100%, <b>50</b>
Valve position at exceeding in cycle time in the winter mode	0 - 100%, <b>30</b>