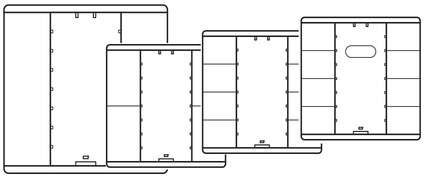
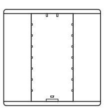
Push-button plus

System M

Operating instructions

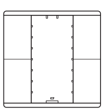


System M



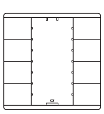
Push-button, 1-gang plus
Art. no. MTN6171.., MTN6275..

System M



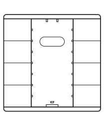
Push-button, 2-gang plus
Art. no. MTN6172.., MTN6276..

System M



Push-button, 4-gang plus
Art. no. MTN6174.., MTN6278..

System M



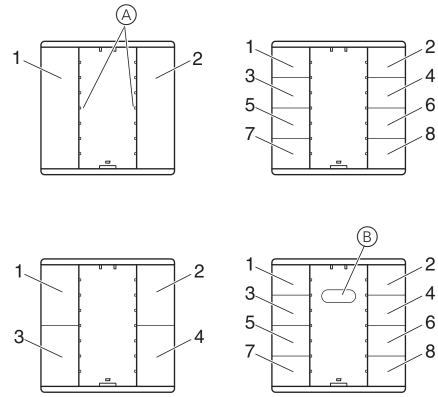
Push-button, 4-gang plus with IR receiver
Art. no. MTN6175.., MTN6279..

Getting to know the push-button

The KNX push-button provides you with two, four or eight operating surfaces, two in the case of 1-gang push-buttons, four in the case of 2-gang push-buttons, and eight in the case of 4-gang push-buttons.
The push-buttons can be set to perform various functions, allowing you, for example, to switch lighting on and off or dim it, control the blinds or retrieve stored scenes.

Push-buttons with IR receiver also allow you to operate each key on the push-button using a Schneider remote control or another IR remote control.

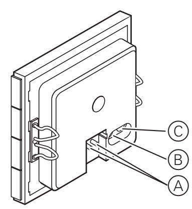
Connections, displays and operating elements



Ⓐ Status LEDs (next to the keys)

Ⓑ IR sensor

1-8 Operating surfaces
(depending on the push-button type; the sequence corresponds to the addressing in the application software)



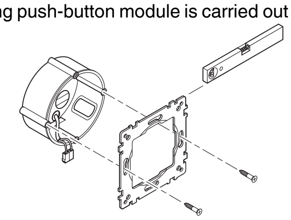
Ⓐ Bus connection

Ⓑ Programming LED

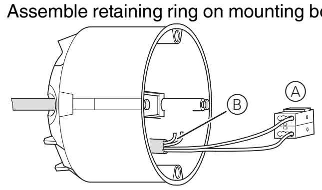
Ⓒ Programming button

How to install the push-button

You need a frame to mount the push-button.
The description which follows shows the installation of a 2-gang push-button. Installation of the 1-gang and 4-gang push-button module is carried out in the same way.



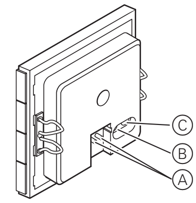
① Assemble retaining ring on mounting box.



② Connect the red bus wire to the red terminal (+) and the black one to the dark grey terminal Ⓐ (-).

The screen and the stability wire, as well as the white and yellow cores of the bus line Ⓑ, are not required.

③ Insulate the screen and stability wires and both cores and place them in the mounting box.



④ Insert the bus terminal into the connection of push-button Ⓐ.

⑤ Insert the push-button into the frame.

⑥ Attach the push-button with frame onto the retaining ring. Make sure that the push-button clicks into place.

How to operate the push-button

DANGER
Risk of fatal injury from electrical current.
All work carried out on the unit may only be performed by skilled electricians. Observe the regulations valid in the country of use, as well as the valid KNX guidelines.

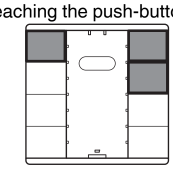
① Load the physical address into the push-button module from the ETS via the KNX.

② Set the desired configuration for the push-button module in the ETS, and transfer the configuration into the push-button module via the KNX.

Make a note of the assignment in the "Push-button assignment" table, last section.

How to operate the push-button by remote control

Push-buttons with IR receiver also allow you to operate each key on the push-button using an IR remote control. When using a remote control, the remote control keys are already assigned (see the operating instructions for the remote control).
If using another IR remote control, the push-button must be taught first.
Teaching the push-button:



① Press and hold the two uppermost keys on the right and the uppermost key on the left of the push-button, until all the status LEDs flash.

② Press the key to be taught on the push-button. The status LED of this key will light up continuously.

③ Repeatedly press the IR remote control key to be taught for approx. 1 second until all status LEDs on the push-button light up continuously.

④ Press any key on the push-button.
The status LEDs continue to flash. You can now teach further keys. If no key is operated for approx. 30 seconds after this, the teaching procedure will be terminated automatically.

The function of the taught-in push-button key can now also be activated using the taught-in IR remote control key.
If no key is operated for about 30 seconds after this, the teaching procedure will be terminated automatically. If the teaching procedure is interrupted, the IR LED only will light up for about 2 seconds. Any existing assignments of remote control keys to this key will then be deleted.
You can use most commercially available IR remote controls. In individual cases, however, an IR remote control key may not work with the push-button.

V6275-561-00 12/07

Technical data

Initialisation:	Due to the telegram rate limit, telegrams can only be created after a minimum of 17 sec has elapsed since initialisation.
Ambient temperature	
Operation:	-5 °C to +45 °C
Storage:	-25 °C to +55 °C
Transport:	-25 °C to +70 °C
Max. humidity:	93 % relative humidity, no moisture condensation
Type of protection:	IP 20

Schneider Electric Industries SAS

If you have technical questions, please contact the Customer Care Center in your country.

www.schneider-electric.com

This product must be installed, connected and used in compliance with prevailing standards and/or installation regulations. As standards, specifications and designs develop from time to time, always ask for confirmation of the information given in this publication.

V6275-561-00 12/07

Universal 1815/1.0

General information

You can use this application to program the push-buttons plus.

Up to two objects are available for each input.

Group addresses are managed dynamically. Maximum no. of group addresses and associations: 150.

Application functions

This application offers a wide range of setting options in order to execute numerous functions with a push-button and controlled KNX devices (e.g. dimming actuators, switch actuators etc).
Naturally, which function is possible in each individual case depends on the KNX devices being controlled. The functions of this application described here can therefore only apply to those specific KNX control functions. Here, only those tabs and parameters which are of relevance to these control functions are described.
You will find an overview of all the tabs, parameters and the related adjustable values in the last section "Parameters and settings".

> Adjustable times (staircase timer, ON delay, OFF delay etc.) are adjusted via the time base and time factor parameters. The actual time is calculated by multiplying both values; e.g. time base 1 second times time factor 3 gives 3 seconds.
> If only one of these parameters is shown, no time adjustment is possible for the parameter setting selected.

Basic settings

Device selection

> First you must adapt the application to the hardware used, as when the device selection is toggled, parameter settings and related group addresses are changed by ETS. Select the "2-gang" or "4-gang" setting.

Tab	**Parameter**
General	Push-button module

Operating LED

You can specify whether the operating LED is switched on or off.

Tab	**Parameter**
Key X	**Operating LED**

Push-button assignment in connection with the Merten remote control (only push-buttons with IR)

The push-button assignment is preset for Merten IR remote controls. However, you can reassign these via the "IR ranges of Merten remote control" parameter.

Tab	**Parameter**
Key X	**IR ranges of Merten remote control**

You may select the desired range for the remote control. The new push-button assignment is displayed below.

Tab	**Parameter**
Key X	**IR range a - b**

> The previous key assignments are overwritten as a result of the reassignment.

Transmit 1/8 bit toggle commands

You can address two actuator groups with 1 or 8 bits (1 byte) simultaneously.

With a 1-bit object type, the object value is first inverted with each push-button action, then sent on the bus, i. e. a "0" becomes a "1", and when the same key is pushed again, a "1" becomes a "0". The device is therefore switched on and off alternately. This switching behaviour is called "toggling". An update or change to the 1-bit/1-byte object value is possible via the bus when another sensor switches the actuator (e .g. via a two-way circuit or a central command). To prevent "incorrect" toggling, you must load the status of the actuator ("1" or "0") in the push-button. Verbinden Sie dazu die Gruppenadresse des zweiten Sensors mit dem Schalt-/Wertobjekt des Tasters.

For 1-byte object types, you can set two values, which are transmitted alternately after each push-button action.

Two objects can also be sent in any combination when the push-button is activated (1 bit / 1 byte).

Tab	**Parameter**
Key X	Functional selection
	Number of objects
	Triggering of status LED
	Object A/B
	Value 1/2

Communication objects

You can select the following communication objects:

Per input:

Function	Object name	Type	Prio	Flags	Behaviour
Input X	Schalt-/ Wertobjekt A/B	1 bit	Low	WCT	Transmit/ receive

Transmit 1/8-bit switching commands

You can address two actuator groups with 1 or 8 bits simultaneously.

Depending on the parameter settings, one of the following

– an ON or OFF telegram

– 1 byte values (0% - 100% in levels)

– 1 byte values (0 - 255) infinitely

– two objects, (1 bit / 1 byte) in any combination

will be transmitted via the switch/value object whenever a key is pressed.

Tab	Parameter
Key X	Functional selection
	Number of objects
	Triggering of status LED
	Object A/B
	Value

Communication objects

You can select the following communication objects:

Per input:

Function	Object name	Type	Prio	Flags	Behaviour
Key X	Switch/value object A/B	1 bit/1 byte	Low	WCT	Transmit/ receive

Dimming

You can use the dimming function for the following:

– dim brighter **and** darker via **one key** (single-surface dimming)

– either dim brighter **or** darker. You need a second key (second input) to dim in the other direction (dual-surface dimming).

You can use the corresponding key (input) to switch the light on or off (press key briefly) or dim it (press key for a longer period, the parameters for the exact period can be set). When switching takes place, an ON/OFF telegram is sent via the switch object. When dimming, dimming up or dimming down is carried out via the 4-bit dimming object; the parameters for the dimming steps can be set. In addition, you can also transmit the corresponding dimming step cyclically for a period of time which can be set as required.

Common parameters for single-surface and dual-surface dimming

Tab	Parameter
Key X	Functional selection
	Detection of long activation time 100 ms * Factor (4-250)
	Triggering of status LED
	Dimming direction
	Cyclical sending of the dimming levels
	only with cyclical transmission of the dimming steps: Base for cyclic interval
	only with cyclical transmission of the dimming steps: Factor for cyclic interval (3-255)

Additional parameters for single-surface dimming

You can dim brighter or darker and also switch on or off using a single key.

The current switching or dimming direction is always dependent on the previous action, i. e. if switched off, pressing the key briefly will switch the light on and vice versa, and if the light has been dimmed up, prolonged activation of the key will dim the light down again. On release after prolonged activation, a stop telegram will be sent via the 4-bit dimming object, thus terminating the dimming procedure in the dimming actuator.

An update or change to the switch/object value is possible via the bus when another sensor switches or dims the actuator (e.g. via a two-way circuit or a central command). To prevent the "wrong" switching/dimming activity, you must load the status of the actuator into the push-button. To do this, connect the group address of the second sensor to the switch/dimming object of the push-button.

A single command is sufficient to cycle through the dimming range. This dimming procedure can be used for most applications. The other possible dimming steps (1/2 - 1/64 brighter or darker) dim brighter or darker by the selected step. For example, to dim from minimum to maximum brightness, you would need to push the key for a prolonged period four times in succession if the level set is 1/4.

Tab	Parameter
Key X	Dimming direction
	Dimming steps (brighter)
	Dimming steps (darker)

"Dimming direction" parameter value for single-surface dimming:

– brighter and darker

Additional parameters for dual-surface dimming

These are used to dim either brighter or darker and to either switch on or off using a single key. Therefore, you must set the parameters for a second key (second input) for the opposite direction.

You can specify whether a stop telegram is to be transmitted when the key is released. When you have enabled the transmission of a stop telegram, a stop telegram will be sent via the 4-bit dimming object after prolonged activation of the key, thus terminating the dimming procedure in the dimming actuator.

A single command is sufficient to cycle through the dimming range. This dimming procedure can be used for most applications. The other possible dimming steps (1/2 - 1/64 brighter or darker) dim brighter or darker by the selected step. For example, to dim from min. to max. brightness, you would need to push the key for a prolonged period four times in succession if the level set is 1/4.

Tab	Parameter
Key X	Dimming direction
	only in the dimming direction "brighter": dimming steps (brighter)
	only in the dimming direction "darker": dimming steps (darker)
	Stop telegram after release

"Dimming direction" parameter value for dual-surface dimming:

– brighter
– darker

Communication objects

You can select the following communication objects:

Per input:

Function	Object name	Type	Prio	Flags	Behaviour
Key X	Switch object	1 bit	Low	WCT	Transmit/ receive
Key X	Dimming object	4 bit	Low	WCT	Transmit/ receive

Blind control

You can use the blind control function to do the following:

– move the blind using an individual key and adjust the slats (single-surface blind operation).
– With the blind control function, you can raise the blinds / adjust the slats using a single key and lower the blinds / adjust the slats using a second key (dual-surface blind operation).
– Move the blind to a pre-specified position.
– Move the blind between two previously specified positions.

Tab	Parameter
Key X	Functional selection

Blind control function up or down with one key in each case (dual-surface blind operation)

After the corresponding key is pressed for a short time, a stop/step telegram will be transmitted; after the key is activated for a prolonged period (the parameters for the exact period can be set), a movement telegram will be transmitted. With this function, you must set the parameters for a second key (second input) with the corresponding settings for the blind movement in the opposite direction. Both keys (inputs) must be given the same group addresses.

Tab	Parameter
Key X	Functional selection
	Detection of long activation time from 100 ms * Factor (4-250)
	Direction of movement

Blind control function up or down with a single key (single-surface blind operation)

The current direction of movement of the blind, or the direction of the slat adjustment, always depends on the previous action, i. e. when the blind has just been moved downwards, it will move upwards the next time the key is activated for a long period (parameters for the period can be set).

After a stop/step telegram has been transmitted to adjust the slats, a stop/step telegram for the same direction of movement can be created by pressing the key again, as long as this subsequent push-button action is carried out within a time period, the parameters for which can be set. If this time period has elapsed, the direction of rotation of the slats will change when the key is pressed briefly.

The push-button (input) can receive telegrams via the stop/step movement objects, and can create corresponding telegrams when the key is pressed, depending on the values received. An update or change to the switch/object value is possible via the bus when another sensor switches the actuator (e .g. via a two-way circuit or a central command). To prevent "incorrect" movement, you must load the status of the actuator into the push-button. To do this, connect the group address of the second sensor to the stop/step object and the movement object of the push-button.

Tab	Parameter
Key X	Detection of long activation time from 100 ms * Factor (4-250)
	Direction of movement
	Change in direction for slat adjustment from 100 ms * Factor (5-50)

Move the blind to a pre-specified position.

If the blind actuator is capable of approaching a specific position, you can define one or two positions using this function, which can be approached by the blind using 1-byte position values with a push-button action. The position values can be set in steps between 0% and 100%, or infinitely from 0-255.

When approaching a position, the set value for the blind position and the slat position is transmitted using a short (or long) push-button action.

To trigger two positions, enter the required blind position and slat position for both. Position value 1 is transmitted with a short push-button action, while position value 2 is transmitted with a long push-button action. No movement or stop/step objects exist with these set parameters.

Tab	Parameter
Key X	Direction of movement
	Selecting the positioning
	Position value 1 (short operation)
	Value for blind position
	Value for slat position
	only with "two positions": Position value 2 (press for long period)

Communication objects

You can select the following communication objects:

Per input:

Function	Object name	Type	Prio	Flags	Behaviour
Key X	Stop/step object	1 bit	Low	CT	Transmit/ receive
Key X	Movement object	1 bit	Low	CT	Transmit/ receive
Key X	Blind position	1 byte	Low	CT	Transmit/ receive
Key X	Slat position	1 byte	Low	CT	Transmit/ receive

Transmit 1-bit, 2-bit (priority control), 4 -bit or 1-byte pulse edge commands

You can use these pulse edge functions to parameterise different object actions. You can transmit one or two objects simultaneously, and select the size of the objects required (1 bit, 2 bit priority control, 4 bit or 1 byte in steps or infinitely) as needed. This enables you to parameterise a large number of application options.

You can specify which actions should be carried out when a key is pressed, and which should be carried out when a key is released. These actions could include:

- Transmit 1 or 0 (with 1 bit)
- Transmit value 1 or value 2 (with 2 bits, 4 bits or 1 byte):
  You can enter two values and set whether and how they are to be transmitted.
- Transmitting a value:
  The object transmits the value which it has currently been given. You can therefore transfer a value e. g. with the transmitting group address which was previously received by another group address.
- Toggling:
  The current object value is inverted and then transmitted. It is therefore switched on and off alternately (toggled). The value can be modified via the bus.
- no action

Tab	Parameter
Key X	Functional selection
	Pulse edge function
	only with extended pulse edge function: Detection of a long activation time from 100 ms * Factor (4-250)
	Number of objects

Normal pulse edge function

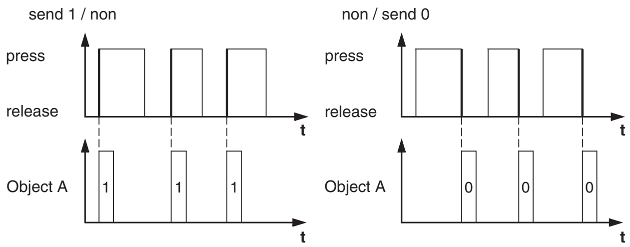
With the standard pulse edge function, you can transmit 1 bit, 2 bit, 4 bit priority control or 1 byte in steps or infinitely.

Tab	Parameter
Key X - (object A&B)	Object A/B
	Action when activated
	Action on release
	only with 2 bit and 4 bit (priority control): value 1 / value 2
	only with 1 byte in steps 0 - 100% value 1 / value 2
	only with 1 byte infinitely 0-255: value 1 / value 2

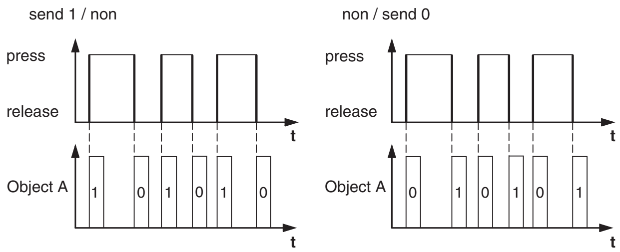
Principle of the pulse edge function

Using the following diagrams, you can see how the pulse edge function behaves when pulse edges rise or fall.

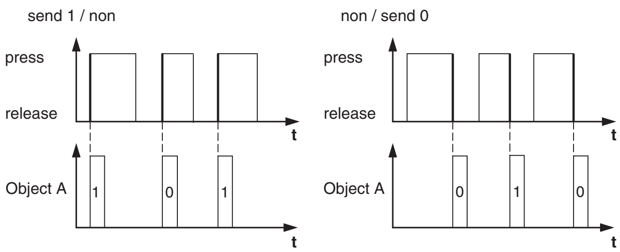
Object A = 1 Bit
action at pressed/released



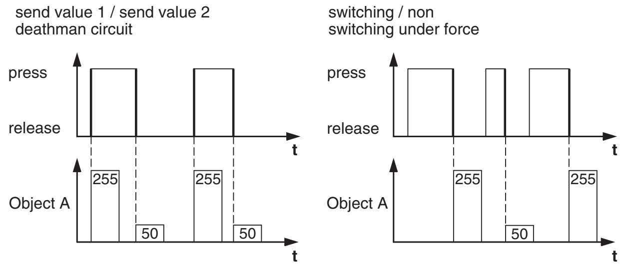
Object A = 1 Bit
action at pressed/released



Object A = 1 Bit
action at pressed/released

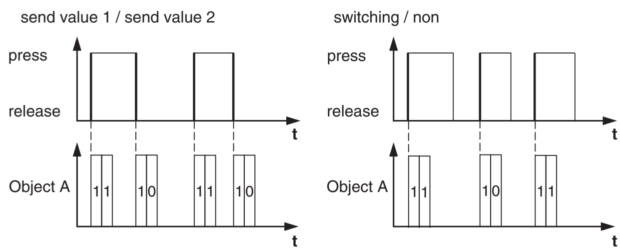


action at pressed/released



Example: Function "death man circuit" or "switching under force"

Object A = 2 Bit (guidance under force)
Wert 1 = 11 (switching on under force)
Wert 2 = 10 (switching off under force)
Action at pressed/released



Extended pulse edge function

With the extended pulse edge function, you have a wider range of functions available, e. g. you can set different actions with a shorter or longer activation time, which apply to both the actions which result when the key is pressed and when the key is released. You can also set a cycle time which can be parameterised for each object.

i When parameterising, bear in mind that you need to set all four types of key activation (brief/ long, pressing and releasing the key) in order to ensure that the push-button (input) functions as required.

i To read the object value it could be necessary to set the Read-Flag manually.

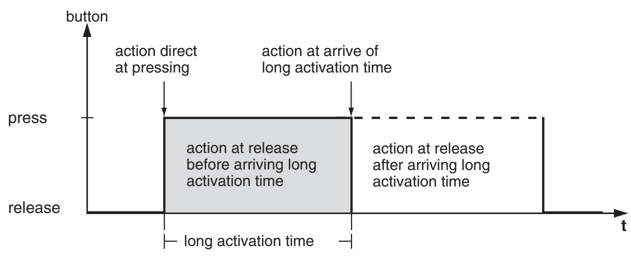
Tab	Parameter
Key X - (object A&B)	Object A/B

A description of the most important actions is given below:

– transmits [value]:
transmits the current value and stops a cyclical transmission.
– transmits [value] immediately and then cyclically:
If no cycle time is running, [value] is transmitted immediately and a new cycle time is started. If a cycle time is already running, this is interrupted, [value] is transmitted and a new cycle time is started.
– transmits [value] only cyclically:
If no cycle time is running, [value] is transmitted immediately and a new cycle time is started. If a cycle time is already running, this is **not** interrupted, [value] is transmitted after the current cycle time has elapsed, and a new cycle time is started.
– sets object value to [value] (readable only)
[value] is written into the object and is not transmitted. An active cycle time is terminated.
– toggles:
compares the current object value with [value]. If both are the same, value 1 or value 2 is transmitted. If both are different, [value] is transmitted.
– toggles, transmits immediately, then cyclically:
The value is toggled (see "toggles") if no cycle time is running, transmitted immediately and a new cycle time is started. If a cycle time is already running, this is interrupted, the toggled value is transmitted and a new cycle time is started. Subsequently, the value which has already been toggled is always transmitted cyclically.
– toggles, only transmits cyclically:
If no cycle time is running, the toggled value is transmitted immediately and a new cycle time is started. If a cycle time is already running, this is **not** interrupted, the toggled value is transmitted after the current cycle time has elapsed, and a new cycle time is started. Subsequently, the value which has already been toggled is always transmitted cyclically.
– toggles and is not transmitted:
The toggled value is written into the object and is not transmitted. An active cycle time is terminated.
– toggles cyclically, transmits immediately, then cyclically:
The value is toggled (see "toggles") if no cycle time is running, transmitted immediately and a new cycle time is started. If a cycle time is already running, this is interrupted, the toggled value is transmitted and a new cycle time is started. Subsequently, it is always toggled cyclically and the new value is transmitted.
– toggles cyclically, only transmits cyclically:
If no cycle time is running, the toggled value is transmitted immediately and a new cycle time is started. If a cycle time is already running, this is **not** interrupted, the toggled value is transmitted after the current cycle time has elapsed, and a new cycle time is started. Subsequently, it is always toggled cyclically and the new value is transmitted.
– toggles cyclically and is not transmitted:
The toggled value is written into the object and is **not** transmitted. Subsequently, it is always toggled cyclically and the new value is written into the object.
– transmits its value:
The current object value is transmitted. An active cycle time is terminated.
– sends its value immediately and then cyclically:
If no cycle time is running, the current object value is transmitted immediately and a new cycle time is started. If a cycle time is already running, this is interrupted, the current object value is transmitted and a new cycle time is started. Subsequently, the current object value is always transmitted cyclically.
– increase the current object value by [value] cyclically:
If no cycle time is running, [value] is added to the current object value, the object value is transmitted, and a new cycle time is started. If a cycle time is already running, this is **not** interrupted, the current object value with [value] added is transmitted and a new cycle time is started.
– reduce the current object value by [value] cyclically:
If no cycle time is running, [value] is subtracted from the current object value, the object value is transmitted, and a new cycle time is started. If a cycle time is already running, this is **not** interrupted, the current object value with [value] subtracted is transmitted and a new cycle time is started.
– transmits [valueA] and after a cycle time [valueB]:
[valueA] is transmitted immediately, and [valueB] is transmitted after **one** cycle time, regardless of whether a cycle time is already running or not (staircase timer function).
– none (stops cyclical transmission):
No action is carried out, and any active cycle time is stopped.
– no change:
The current action remains unchanged (e. g. "transmits value1 and after a cycle time, transmits value2").
– none (stop after current cycle time has elapsed):
No action is currently carried out, but any active cycle time is **not** stopped. It runs through until the end, and then transmits the corresponding value.

Examples of use for the pulse edge function

The following activation sequence diagram shows the phases into which the pulse edge function is divided:



For example: Staircase lighting function with cleaning light function

With a brief push-button action, the switch actuator switches on the light. A long push-button action extends the staircase lighting function (= cleaning light function) until a second, long push-button action switches off the actuator. The switch actuator requires a staircase lighting function and a disable function for this function.

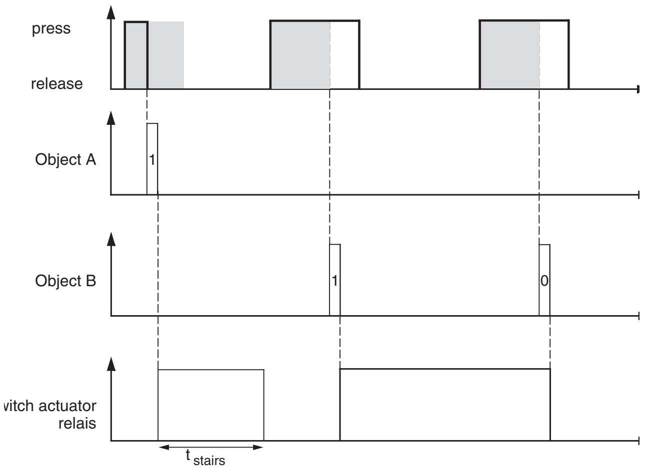
Number of objects = 2 (object A/B)

Object A/B = 1 bit

Object A: Action on release before the long activation time has elapsed = transmits 1

Object B: Action on completion of the long activation time = toggles

To do this, connect object A with the switch object and object B with the disable object of the switch actuator.



For example: short and long staircase time

You can use this function to produce a brief and a long staircase time with the push-button. The switch actuator requires no staircase lighting function for this request.

With a brief push-button action, the switch actuator switches on the light, and after a parameterised cycle time (e. g. 3 minutes), it switches it back off again. With a long push-button action, the same function is carried out, but with a longer cycle time (e. g. 6 minutes).

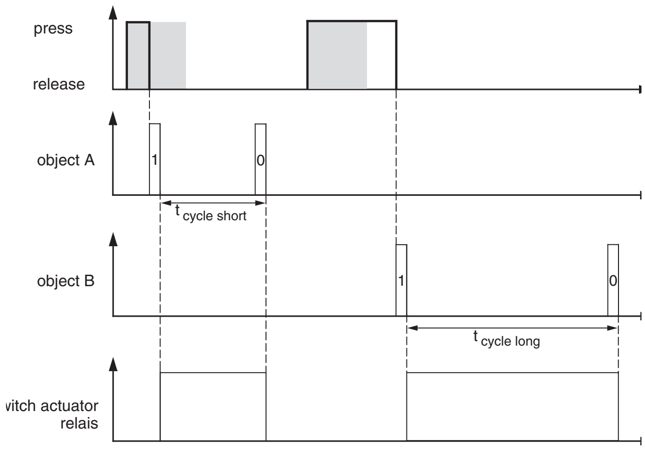
Number of objects = 2 (object A/B)

Object A/B = 1 bit

Object A: Action on release before the long activation time has elapsed = transmits 1. After a cycle time has elapsed (here 3 minutes) = transmits 0

Object B: Action on release when the long activation time is completed = transmits 1. After a cycle time has elapsed (here 6 minutes) = transmits 0

To do this, connect object A and object B with the switch object of the switch actuator.



For example: Switch the light on/off permanently, or switch off after a cycle time has elapsed.

With a brief push-button action, the switch actuator switches the light on or off permanently. With a long push-button action, the light switches on, and after a parameterised cycle time (e .g. 6 minutes), it switches back off again. Due to the cycle time in the push-button which can be parameterised, the switch actuator requires no staircase lighting function for this function.

Number of objects = 2 (object A/B)

Object A/B = 1 bit

Object A: Action on release before the long activation time has elapsed = toggles

Object B: Action when the long activation time is completed = transmits 1. After a cycle time has elapsed (here 6 minutes) = transmits 0. Action on release when the long activation time is completed = no change.

To do this, connect object A and object B with the switch object of the switch actuator.

For example: electronic protection against theft

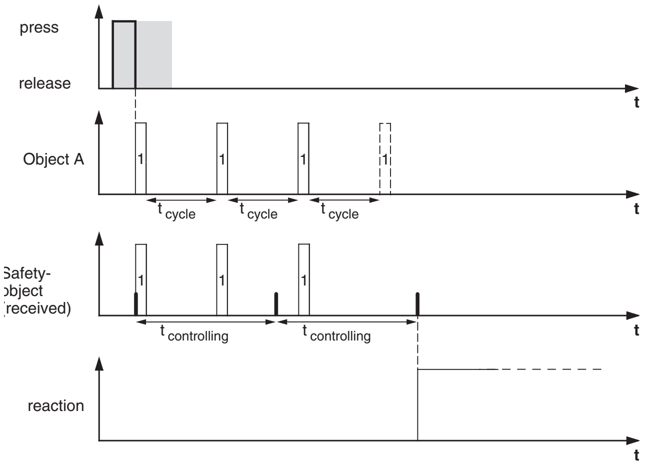
This example will show you how to program electronic protection against theft for the push-button. It is activated by a brief push-button action and then transmits cyclically. As soon as the push-button is separated, this can be shown on a display, or an alarm can be triggered.

Number of objects = 1 (object A)

Object A = 1 bit

Object A: Action on release before the long activation time has elapsed = transmits 1 immediately and then cyclically. Action when the long activation time is completed = no change. Action on release after the long activation time is completed = no change. Cycle time = e. g. 10 minutes

To do this, connect object A with an object which anticipates cyclical telegrams (e. g. a safety object). The monitoring time set on the safety object must be longer than the cycle time of the push-button. If the safety object receives no telegrams from the push-button during this time, a reaction which can be parameterised is activated (e. g. the channel is switched on).



For example: Effect lighting

This example shows you how to program effect lighting, for example for a display window. A long push-button action switches between two different lighting scenes. A short push-button action stops the toggling and transmits a scene (to retrieve the scene, use the scene module for the actuator which has been activated) which switches off everything.

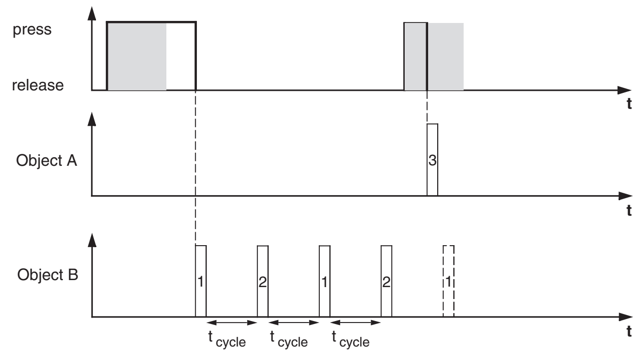
Number of objects = 2 (object A/B)

Object A/B = 1 byte infinite 0-255

Object A: Direct action when activated = none (stops cyclical transmission). Action on release before the long activation time has elapsed = transmits 1. Action when the long activation time is completed = none (stops cyclical transmission). Action on release after the long activation time is completed = none (stops cyclical transmission). Value 1 = 3.

Object B: Direct action when activated = none (stops cyclical transmission). Action on release before the long activation time has elapsed = none (stops cyclical transmission). Action when the long activation time is completed = none (stops cyclical transmission). Action on release after the long activation time has been completed = toggles cyclically, transmits immediately, then cyclically. Value 1 = 1, value 2 = 2nd cycle time = e. g. 1 minute.

To do this, connect object A and object B with the extension unit object of the scene function.



Communication objects

You can select the following communication objects:

Function	Object name	Type	Prio	Flags	Behaviour
Key X	Object A/B	1 bit	Low	WCT	Transmit/ receive
Key X	Object A/B	2 bit	Low	WCT	Transmit/ receive
Key X	Value object A/B	1 byte	Low	WCT	Transmit/ receive

Transmit 2-byte pulse edge commands via an object

You can also use these pulse edge functions to parameterise different object actions. However, in contrast to pulse edge functions, you can only transmit one object with 1 bit, 2 bits, 4 bits or 1 byte.

You can transmit normal or extended pulse edge commands. With normal pulse edge commands, you can specify which actions should be carried out when a key is pressed, and which should be carried out when a key is released. With extended pulse edge commands, you can also set the actions before and after the long activation time is completed.

You can transmit floating point numbers and whole numbers, without or without a sign.

Tab	Parameter
Key X	Functional selection
	Pulse edge function
	only with extended pulse edge function: Detection of a long activation time from 100 ms * Factor (4-250)
	Action when activated
	Action on release *only with extended pulse edge function:*
	Action on release before the long activation time has elapsed
	Action when the long activation time is completed
	Action on release after the long activation time has been completed
	Base for cyclic interval
	Factor for cyclic interval (3-255)

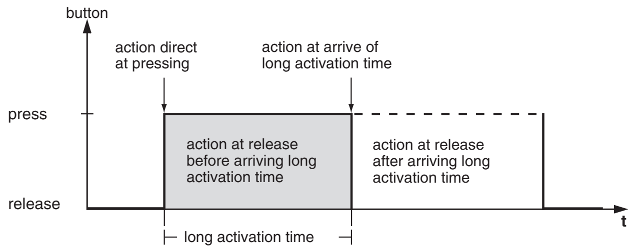
Explanation of actions: See 1-bit pulse edges.

Tab	Parameter
Key X - values	Object type value
	Only with floating point: value 1/2 Basic value, adjustable value range in brackets
	only with whole number with sign: value 1/2 (-32768 -32767)
	Only with floating point: value 1/2 (0 - 65535)

Communication objects

You can select the following communication objects:

Function	Object name	Type	Prio	Flags	Behaviour
Key X	Value object A	2 byte	Low	WCT	Transmit/ receive



Setting the parameters for the 8-bit linear regulator

You can use the 8-bit linear regulator function to program a key (input) as a linear regulator. You can parameterise all four actions when pressing/releasing with a shorter or longer operating time in each case. You can establish the function with or without limit values (start/end value).

Tab	Parameter
Key X	Functional selection
	Detection of long activation time from 100 ms * Factor (4-250)
Key X (2)	Linear regulator function
	Direct action when activated
	Action on release before the long activation time has elapsed
	Action when the long activation time is completed
	Action on release after the long activation time has been completed
	only with "start value and end value" Start value
	Step value
	only with "start value and end value" End value
	Base for cyclic interval
	Factor for cyclic interval (3-255)

A description of the actions is given below:

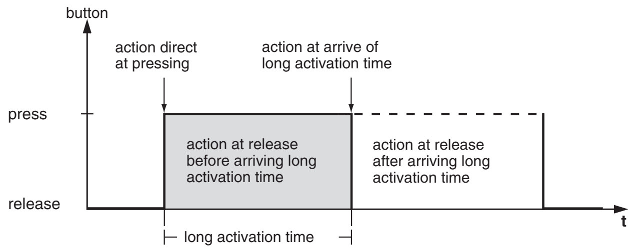
– Transmit the start value, then increase cyclically by the step value:
  If no cycle time is running, the start value is transmitted immediately and a new cycle time is started. If a cycle time is already running, this is interrupted, the start value is transmitted and a new cycle time is started.
– Transmit the end value, then reduce cyclically by the step value:
  If no cycle time is running, the end value is transmitted immediately and a new cycle time is started. If a cycle time is already running, this is interrupted, the end value is transmitted and a new cycle time is started.
– Increase the current object value cyclically:
  Increase the current object value cyclically by the parameterised step interval.
– Increase the current object value once:
  Increase the current object value once by the parameterised step interval. An active cycle time is terminated.
– Reduce the current object value cyclically:
  Reduce the current object value cyclically by the parameterised step interval.
– Reduce the current object value once:
  Reduce the current object value once by the parameterised step interval. An active cycle time is terminated.
– Reverse the slide direction and transmit cyclically:
  If no cycle time is running, the slide is pushed in the opposite direction (of this push-button) and a new cycle time is started. If a cycle time is already running, it is interrupted, the slide is immediately pushed in the opposite direction (of this push-button) and a new cycle time is started.
– Move step-by-step to the limit values, and then back again:
  The limit values are approached by one step interval at a time. When a limit is reached, the slide direction is reversed for the next action.
– none (stops cyclical transmission):
  No action is carried out, and any active cycle time is stopped.
– no change:
  No action is carried out, and any active cycle time is continued.

You can only maintain the limit values and the toggling to a new slide direction with local operation!

Communication objects

You can select the following communication objects:

Function	Object name	Type	Prio	Flags	Behaviour
Key X	Value object A	1 byte	Low	WCT	Transmit/ receive



Retrieving scenes

The scene retrieval function does not access the internal scene module, but only accesses the bus externally via communication objects.

There are two types of scene function:

– normal
– extended

With the standard scene function, a scene is retrieved by a brief push-button action while a long push-button action is used to save a scene. You merely have to set the time after which a push-button action is identified as being long, together with the status LED control and the scene address.

Tab	Parameter
Key X	Functional selection
	Detection of long activation time from 100 ms * Factor (4-250)
	Scene function
	Only with "extended" Number of objects
	Only with "normal scene function" Scene address (0-63)

With the extended scene function, a wider range of functions are available. You can set actions for a shorter or longer activation time for both pressing and releasing the keys. You can also program a cycle time.

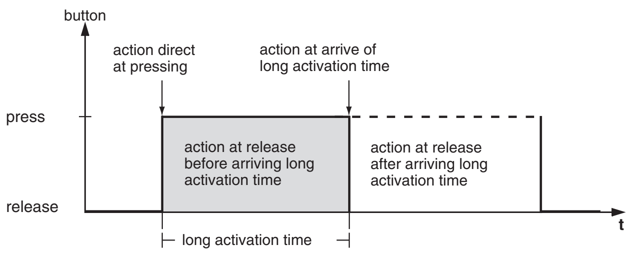
Depending on how many objects you have set in the scene function, you must make the settings in additional parameter windows ("Key X - object A") or ("Key X - object B"):

Tab	Parameter
Key X - (object A/B)	Direct action when activated
	Action on release before the long activation time has elapsed
	Action when the long activation time is completed
	Action on release after the long activation time has been completed
	Value 1 Scene address (0-63)
	Scene addresses are designed for the scene
	Value 2 Scene address (0-63)
	Scene addresses are designed for the scene
	Base for cyclic interval
	Factor for cyclic interval (3-255)

Communication objects

You can select the following communication objects:

Function	Object name	Type	Prio	Flags	Behaviour
Key X	Object A/B	1 byte	Low	WCT	Transmit/ receive



Activating the disable function for the inputs

You can block the inputs in different ways:

1. Separately for each input
2. All inputs function as a master input
3. Toggle between two local scenes

You can determine whether a disable object = 0 or = 1 should be blocked.

Tab	Parameter
Disable function	Disable function
	Block
	Type of blocking

Separately for each input

You can use this function to block each of the four or eight inputs individually. When an input is blocked, it fulfils no function. You can use an additional parameter to parameterise the behaviour of cyclical functions.

Tab	Parameter
Disable function/Key X: Disable	Key X
	Cyclical actions are

All keys function as a master input

You can use this action to specify one of the two or four inputs as the master input. When any key is pressed, the action which has been parameterised for the master key is carried out.

Tab	Parameter
Disable function	Disable function
	Block
	Type of blocking
	Master key

Toggle between two local scenes

This action enables you to toggle between two scenes which are parameterised in the actuator (local) or in the scene module.

Tab	Parameter
Disable function	Disable function
	Block
	Type of blocking
	Toggle between scene address
	and scene address

Communication objects

You can select the following communication objects:

Function	Object name	Type	Prio	Flags	Behaviour
Disable function	Disable object	1 bit	Low	WC	Receive

When a disable function is activated via the disable object, all current push-button functions are reset or interrupted.

● Parameterising scenes in the scene module

The push-button is fitted with its own scene module, which enables you to save up to eight scenes permanently. The saved scenes can be overwritten if you have parameterised a release for this purpose.

The entire scene function is controlled via the extension unit object (1 byte). You also have an object for programming release, as well as seven 1/8-bit objects and one object for values up to 16 bits, in order to transmit scene values to the bus.

You can set the time between the actuator read telegrams. This makes sense, for example, when the anticipated response can last a long time (line coupler, area coupler).

If a read request is lost, or is not responded to, the scene from the current object value is saved (either through a read request, or written via an output). To check the correct saving procedure, you should retrieve the scene last saved on the push-button. If this remains unchanged, the individual saving procedure has been completed free of errors. If there is a difference, an incorrect response has been made to a read request.

If the push-button works through a scene, and a further scene is retrieved, the current process is interrupted and the last retrieved scene is worked through.

Tab	Parameter
Scene function	Scene function
	only when scene function is activated: save scenes
	Interval between two actuator read telegrams 100 ms * Factor (2-255)

This tab and the subsequent scene tabs only appear when the scene function is switched on.

In this parameter window, you can specify the data type of the max. eight output objects. A particular feature is actuator group 7: with this group, you can also transmit values which are larger than 8 bits.

When the alarm function is active, the eighth actuator group is no longer valid, since it is reserved for the alarm function.

Tab	Parameter
Scene actuator groups	Actuator group 1-8
Scene X	Scene is retrieved with the following value (0-63)
	Interval between scene telegrams 100 ms * Factor (2-255)
	only with "Switch object" data type: value 1 to value 8
	only for data type "Value object (8 bit in steps)": value 1 to value 8
	only for data type "Value object (8 bit infinite)": value 1 to value 8
	only for data type "Priority control object": value 1 to value 8
	only with 16-bit value object: Value 7

Für jede Szene legen Sie fest, über welche Szenenadresse die Szene am Szenen-Eingangsobjekt aufgerufen werden soll. You also specify the time between the individual scene telegrams.

Make sure that you always enter clear scene addresses for this device, i. e. no scene address should be allocated more than once.

Finally, specify the actuator groups and their values for this scene. These only remain valid up to the first time the scene is saved.

The value range which can be set depends on the data type set for the "scene actuator groups".

Communication objects

You can select the following communication objects:

Function	Object name	Type	Prio	Flags	Behaviour
Scene function	Extension unit object	1 byte	Low	WC	Receive
Transmit value	Actuator group 1-8	1 byte	Low	WCT	Transmit/ receive
Transmit value	Actuator group 7	2 byte	Low	WCT	Transmit/ receive
Switching	Actuator group 1-8	1 bit	Low	WCT	Transmit/ receive
Priority control	Actuator group 1-8	2 bit	Low	WCT	Transmit/ receive
Save scenes	Enable object	1 bit	Low	WC	Receive

Behaviour when bus voltage is applied/ restored or fails

Behaviour on application/recovery of the bus voltage

When a bus voltage is applied or recovered, telegrams can be sent depending on the setting.

Behaviour on failure of the bus voltage

Any status LEDs which were lit will be switched off.

Parameters and settings

General	
Parameter	**Setting**
Push-button module	1-gang
	2-gang
	4-gang
	4-gang IR
Operating LED	switched on
	switched off
IR ranges of Merten remote control	yes
	no
IR range	a - b,

Key X	
Parameter	**Setting**
Functional selection	Toggle
	Switching
	Dimming
	Blind control
	Pulse edges 1 bit, 2 bit (priority), 1-byte values
	Pulse edges with 2-byte values
	8-bit linear regulator
	Scene

When "Functional selection" has been set to "Toggle" in the "Key X" tab:

Key X	
Parameter	**Setting**
Functional selection	Toggle
Number of objects	**one**
	two
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF

Key X	
Parameter	**Setting**
Object A/B	**1 bit**
	1 byte in steps 0% - 100%
	1 byte infinitely 0 - 255
Value	**100%**
	adjustable in steps of ten as well as 25% and 75%
Value	0 - 255, **255** default setting

When "Function selectional" has been set to "Switching" in the "Key X" tab:

Key X	
Parameter	**Setting**
Functional selection	Switching
Number of objects	**one**
	two
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF
Object A/B	**1 bit**
	1 byte in steps 0% - 100%
	1 byte infinitely 0 - 255
Value	**ON telegram**
	OFF telegram
Value	**100%**
	adjustable in steps of ten as well as 25% and 75%
Value	0 - 255, **255** default setting

When "Functional selection" has been set to "Dimming" in the "Key X" tab:

Key X	
Parameter	**Setting**
Functional selection	Dimming
Detection of long activation time 100 ms * Factor (4-250)	4 - 250, **6** default setting

Key X	
Parameter	**Setting**
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF
Dimming direction	brighter
	darker
	brighter and darker
Dimming steps (brighter)	**to max. brightness**
	1/2 brighter
	1/4 brighter
	1/8 brighter
	1/16 brighter
	1/32 brighter
	1/64 brighter
Dimming steps (darker)	**to min. brightness**
	1/2 darker
	1/4 darker
	1/8 darker
	1/16 darker
	1/32 darker
	1/64 darker
Cyclical transmission of the dimming steps	yes
	no
Base for cyclic interval	**0.1 second**
	1 second
	1 minute
	1 hour
	1 day
Factor for cyclic interval (3-255)	3 - 255, **8** default setting
Stop telegram after release	**enabled**
	disabled

When "Functional selection" has been set to "Blind control" in the "Key X" tab:

Key X	
Parameter	**Setting**
Functional selection	Blind control

Key X	
Parameter	**Setting**
Detection of long activation time from 100 ms * Factor (4-250)	4 - 250, **6** default setting
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF
Direction of movement	up
	down
	up and down
	with position values
Change in direction for slat adjustment from 100 ms * Factor (5-50)	5 - 50, **10** default setting
Selecting the positioning	one position (press briefly)
	two positions (differentiation between short/long operation)
Position value 1 (short operation)	**in steps of 0% - 100%**
	infinitely 0-255
Value for blind position	*with "steps":* 0% - 100 % in steps of 10, **100%** default setting
	with "infinitely": 0 - 255 in single steps, **255** default setting
Value for slat position	with "steps": 0 % - 100 % in steps of 10, **0%** default setting
	with "infinitely": 0 - 255 in single steps, **0** default setting
Position value 2 (long operation)	**in steps of 0% - 100%**
	infinitely 0-255

When "Functional selection" has been set to "Pulse edge 1 bit, 2 bit (priority), 1-byte values" in the "Key X" tab:

Key X	
Parameter	**Setting**
Functional selection	Pulse edges 1 bit, 2 bit (priority), 1-byte values
Pulse edge function	**normal (pressed, released)**
	extended (long and short activation)

Key X	
Parameter	**Setting**
Detection of long activation time from 100 ms * Factor (4-250)	4 - 250, **6** default setting
Number of objects	**one**
	two
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF
Object A/B	**1 bit**
	2 bit (priority control)
	1 byte in steps 0% - 100%
	1 byte infinitely 0 -255
Action when activated	*only with 1 bit:* **transmits 1**
	only with 1 bit: transmits 0
	only with 2 bit/1 byte: transmits value 1
	only with 2 bit/1 byte: transmits value 2
	toggles
	transmits its value
	none
Action on release	*only with 1 bit:* transmits 1
	only with 1 bit: **transmits 0**
	only with 2 bit/1 byte: transmits value 1
	only with 2 bit/1 byte: transmits value 2
	toggles
	transmits its value
	none
value 1 / value 2	switch on with priority control (11)
	switch off with priority control (10)
	switch off priority control (00)
value 1 / value 2	0 -100% in 10% steps **100%** default setting value 1, **0%** default setting value 2

Key X	
Parameter	**Setting**
value 1 / value 2	0- 255, **255** default setting value 1, **0** default setting value 2

Key X - (object A/B)	
Parameter	**Setting**
Object A/B	**1 bit**
	2 bit (priority control)
	1 byte in steps 0% - 100%
	1 byte infinitely 0 -255

Key X - (object A/B)	
Parameter	**Setting**
Direct action when activated Action on release before the long activation time has elapsed Action when the long activation time is completed Action on release after the long activation time has been completed	*only with 1 bit:*transmits 1
	only with 1 bit: transmits 1 immediately and then cyclically
	only with 1 bit: transmits 1 only cyclically
	only with 1 bit: sets object value to 1 (readable only)
	*only with 1 bit:*transmits 0
	only with 1 bit: transmits 0 immediately and then cyclically
	only with 1 bit: transmits 0 only cyclically
	only with 1 bit: sets object value to 0 (readable only)
	only with 2 bit/1 byte: transmits value 1
	only with 2 bit/1 byte: transmits value 1 immediately and then cyclically
	only with 2 bit/1 byte: transmits value 1 only cyclically
	only with 2 bit/1 byte: sets object value to value 1 (readable only)
	only with 2 bit/1 byte: transmits value 2
	only with 2 bit/1 byte: transmits value 2 immediately and then cyclically
	only with 2 bit/1 byte: transmits value 2 only cyclically
	only with 2 bit/1 byte: sets object value to value 2 (readable only)
	toggles
	toggles, transmits immediately, then cyclically
	toggles, only transmits cyclically
	toggles and is not transmitted
	toggles cyclically, transmits immediately, then cyclically
	toggles cyclically, only transmits cyclically
	toggles cyclically and is not transmitted
	only with 1 bit: transmits its value
	only with 1 bit: transmits its value immediately and then cyclically
	only with 1 bit: transmits 1 immediately and after a cycle time of 0
	only with 2 bit/1 byte transmits value 1 and after a cyclic interval value 2
	only with 1 byte increase the current object value by value 1 cyclically
	only with 1 byte reduce the current object value by value 2 cyclically
	none (stops cyclical transmission)
	no change
	none (stop after current cycle time has elapsed)

Key X - (object A/B)	
Parameter	**Setting**
value 1 / value 2	switch on with priority control (11)
	switch off with priority control (10)
	switch off priority control (00)
value 1 / value 2	0 -100% in 10% steps **100%** default setting value 1, **0%** default setting value 2
value 1 / value 2	0- 255, **255** default setting value 1, **0** default setting value 2
Base for cyclic interval	**0.1 second**
	1 second
	1 minute
	1 hour
	1 day
Factor for cyclic interval (3-255)	3-255, **10** default setting

When "Functional selection" has been set to "Pulse edges with 2-byte values" in the "Key X" tab:

Key X	
Parameter	**Setting**
Functional selection	Pulse edges with 2-byte values
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF
Pulse edge function	**normal (pressed, released)**
	extended (long and short activation)
Detection of long activation time from 100 ms * Factor (4-250)	4 - 250, **6** default setting

Key X	
Parameter	**Setting**
Action when activated	**transmits value 1**
	only with extended pulse edge function: transmits value 1 immediately and then cyclically
	only with extended pulse edge function: transmits value 1 only cyclically
	only with extended pulse edge function: sets object value to value 1 (readable only)
	transmits value 2
	only with extended pulse edge function: transmits value 2 immediately and then cyclically
	only with extended pulse edge function: transmits value 2 only cyclically
	only with extended pulse edge function: sets object value to value 2 (readable only)
	transmits its value
	only with extended pulse edge function: transmits value 1 and after a cyclic interval value 2
	only with extended pulse edge function: none (stops cyclical transmission)
	none
Action on release Action on release before the long activation time has elapsed Action when the long activation time is completed Action on release after the long activation time has been completed	transmits value 1
	only with extended pulse edge function: transmits value 1 immediately and then cyclically
	only with extended pulse edge function: transmits value 1 only cyclically
	only with extended pulse edge function: sets object value to value 1 (readable only)
	transmits value 2
	only with extended pulse edge function: transmits value 2 immediately and then cyclically
	only with extended pulse edge function: transmits value 2 only cyclically
	only with extended pulse edge function: sets object value to value 2 (readable only)
	transmits its value
	only with extended pulse edge function: transmits value 1 and after a cyclic interval value 2
	only with extended pulse edge function: **none (stops cyclical transmission)**
	none

Key X - values	
Parameter	**Setting**
Object type value	Floating point
	Whole number with sign (-32768 ... 32767)
	Whole number without sign (0 ... 65535)
Value 1/2 Basic value, adjustable value range in brackets	different values between 0.01 and 327.68 in different step intervals **0.01 (0 to 20.47)** default setting
value 1/2 (-32768 -32767)	-32768 - 32767
value 1/2 (0 - 65535)	0 - 65535

When "Functional selection" has been set to "8-bit linear regulator" in the "Key X" tab:

Key X (2)	
Parameter	**Setting**
Linear regulator function	**with start value and end value**
	without start value and end value
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF

Key X (2)	
Parameter	**Setting**
Direct action when activated Action on release before the long activation time has elapsed Action when the long activation time is completed Action on release after the long activation time has been completed	*only with "start value and end value"* Transmit the start value, then increase cyclically by the step interval
	only with "start value and end value" Transmit the end value, then reduce cyclically by the step interval
	Increase the current object value cyclically
	Increase the current object value once
	Reduce the current object value cyclically
	Reduce the current object value once
	Reverse the slide direction and transmit cyclically
	Move step-by-step to the limit values and then back again
	none (stops cyclical transmission)
	no change
Start value	0 - 255 in single steps
Step value	0 - 255 in single steps
End value	0 - 255 in single steps
Base for cyclic interval	**0.1 second**
	1 second
	1 minute
	1 hour
	1 day
Factor for cyclic interval (3-255)	3 - 255 in single steps, **5** default setting

When "Functional selection" has been set to "Scene" in the "Key X" tab:

Key X	
Parameter	**Setting**
Functional selection	Scene
Detection of long activation time from 100 ms * Factor (4-250)	4 - 250 in single steps, **30** default setting
Scene function	**normal (short = send / long = save)**
	extended
Number of objects	**one**
	two

Key X	
Parameter	**Setting**
Triggering of status LED	switched on
	switched off
	from switch/value object A
	from switch/value object B
	from status feedback object
	operation = ON / release = OFF
	prolonged operation = ON / release = OFF
	flashes
	flashes if switch/value object A not equal to 0
	flashes if switch/value object B not equal to 0
	flashes if switch/value object A equal to 0
	flashes if switch/value object B equal to 0
	flashes if status feedback object equal to 1
	flashes if status feedback object equal to 0
	operation = flash / release = OFF
	prolonged operation = flash / release = OFF
Scene address (0-63)	0 - 63 in single steps

Key X - (object A/B)	
Parameter	**Setting**
Action when activated	transmits value 1
	transmits value 2
	toggles
	toggles cyclically, transmits immediately, then cyclically
	transmits value 1 and after a cyclic interval value 2
	none (stops cyclical transmission)
	no change
Action on release before the long activation time has elapsed	**transmits value 1**
	transmits value 2
	toggles
	toggles cyclically, transmits immediately, then cyclically
	transmits value 1 and after a cyclic interval value 2
	none (stops cyclical transmission)
	no change
Action when the long activation time is completed	transmits value 1
	transmits value 2
	toggles
	toggles cyclically, transmits immediately, then cyclically
	transmits value 1 and after a cyclic interval value 2
	none (stops cyclical transmission)
	no change

Key X - (object A/B)	
Parameter	**Setting**
Action on release after the long activation time has been completed	transmits value 1
	transmits value 2
	toggles
	toggles cyclically, transmits immediately, then cyclically
	transmits value 1 and after a cyclic interval value 2
	none (stops cyclical transmission)
	no change
Scene address 1 (0-63) Scene address 2 (0-63)	0 - 63 in single steps
Scene address 1 is designed to Scene address 2 is designed to	retrieve the scene (set by default to value 1)
	store the scene (set by default to value 2)
Base for cyclic interval	0.1 second
	1 second
	1 minute
	1 hour
	1 day
Factor for cyclic interval (3-255)	3 - 255 in single steps, **10** default setting

Disable function	
Parameter	**Setting**
Disable function	**disabled**
	enabled
Block	For object value "0"
	For object value "1"
Behaviour at start of blocking	separately for each input
	all inputs function as a master input
Master input	Input **1**... 4

Taste 1-X: Disable	
Parameter	**Setting**
Key X	**lock**
	do not disable

Scene function	
Parameter	**Setting**
Scene function	switched on
	switched off
only when scene function is activated: Save scenes	**enabled**
	if enable object is equal to "1"
	disabled
Interval between two actuator read telegrams 100 ms * Factor (2-255)	2 - 255 in single steps, **10** default setting

Scene actuator groups	
Parameter	**Setting**
Actuator group X	Switch object
	Value object (8 bit in steps)
	Value object (8 bit infinitely)
	Priority control object
	only with actuator group 7: Value object (16-bit whole number without sign)
	only with actuator group 7: Value object (16-bit whole number with sign)
	only with actuator group 7: Value object (16-bit floating point value)

Scene X	
Parameter	**Setting**
Scene is retrieved with the following value (0-63)	0 - 63 in single steps
Interval between scene telegrams 100 ms * Factor (2-255)	2 - 255 in single steps

Scene X values	
Parameter	**Setting**
only with "Switch object" data type: Value 1 to value 8	ON telegram
	OFF telegram
	do not send a telegram
only for data type "Value object (8 bit in steps)": value 1 to value 8	**do not send a telegram**
	0 -100% in 10% steps
only for data type "Value object (8 bit infinite)": value 1 to value 8	**do not send a telegram**
	0 - 254 in single steps
only for data type "Priority control object": value 1 to value 8	switch on with priority control (11)
	switch off with priority control (10)
	remove priority control (00)
	do not send a telegram
only with 16-bit value object: Value 7	send telegram
	do not send a telegram