

## Product documentation

Signal panel  
Art. No. MBT 2424



**ALBRECHT JUNG GMBH & CO. KG**  
Volmestraße 1  
58579 Schalksmühle  
GERMANY

Telefon: +49 2355 806-0  
Telefax: +49 2355 806-204  
kundencenter@jung.de  
www.jung.de

Issue: 09.10.2020  
MBT 2424

## Table of Contents

<b>1</b>	<b>Product definition</b>	<b>4</b>
1.1	Product catalogue	4
1.2	Function	4
1.3	Accessories	5
<b>2</b>	<b>Installation, electrical connection and operation</b>	<b>6</b>
2.1	Safety instructions	6
2.2	Device components	7
2.3	Fitting and electrical connection	9
2.4	Commissioning	11
2.5	Operation	13
<b>3</b>	<b>Technical data</b>	<b>15</b>
<b>4</b>	<b>Software description</b>	<b>16</b>
4.1	Software "MBT 2424 501311"	16
4.1.1	Software specification	16
4.1.2	Scope of functions	17
4.1.3	Object table	18
4.1.3.1	General	18
4.1.3.2	Button objects	19
4.1.3.3	Disabling objects	28
4.1.3.4	Alarm message objects	32
4.1.3.5	Objects of the logic gate	33
4.1.3.6	Objects of the timers	34
4.1.3.7	Scene objects	35
4.1.4	Functional description	36
4.1.4.1	Push button function	36
4.1.4.1.1	"Switching/pushing" button function	36
4.1.4.1.2	Button function "Dimming"	37
4.1.4.1.3	Button function "Venetian blind"	38
4.1.4.1.4	Button function "Scene extension"	41
4.1.4.1.5	Button function "value transmitter"	42
4.1.4.1.6	2-channel operation function	45
4.1.4.2	LED functions	47
4.1.4.2.1	Button-dependent LED functions	47
4.1.4.2.2	Button-independent LED functions	49
4.1.4.3	Disabling function	53
4.1.4.4	Light scene function	56
4.1.4.5	Piezo signal device function	59
4.1.4.6	Dismantling alarm	60
4.1.4.7	Software functions	62
4.1.4.7.1	Logic gate	62
4.1.4.7.2	Timers	63
4.1.5	Parameters	64
4.1.5.1	General parameters	64
4.1.5.2	Push-button parameters	66
4.1.5.3	Parameter of the disabling functions	81

4.1.5.4	Parameters of the Piezo signal device .....	87
4.1.5.5	Parameters of the dismantling alarm .....	88
4.1.5.6	Parameters of the logic gate .....	89
4.1.5.7	Parameters of the timers .....	91
4.1.5.8	Parameters of the scene functions .....	93
<b>5</b>	<b>Appendix .....</b>	<b>96</b>
5.1	Index .....	96

## 1 Product definition

### 1.1 Product catalogue

Product name: MBT 2424

Use: Controller

Design: UP (concealed)

Art. No. MBT 2424

### 1.2 Function

This device combines the functions of a KNX bus coupling unit, a push-button sensor, a device for acoustic signalling and optical display of operating states of the bus subscribers. The unit is supplied via the bus and an external 24VAC/DC power supply (SELV according to EN60742). The push-button sensor is operated by touching the capacitive sensor fields. The optical display takes place by means of three colour LEDs, the acoustic signalling by means of a Piezo signal device variable in pitch.

Push button functionality:

When a button is pressed, the device transmits telegrams to the KNX, depending on the ETS parameter settings. These can be, for instance, telegrams for switching or push button control, for dimming or for controlling blinds. It is also possible to program value transmitter functions, such as dimming value transmitters, light scene extensions, temperature value transmitters or brightness value transmitters. The operation concept of one operating area is predefined as a push-button function. A button-click can indicate acoustically the pressing of a button.

Display functionality:

Three-colour LEDs on the right hand side of the sensor buttons can indicate a variety of display information completely independent of the push-button function. The LEDs can, for example, indicate operating states of fault sensors as well as room temperature controllers, results of logic value comparisons or signal fault messages, flash or be permanently switched on or off. Each colour of the LED can be controlled by three separate objects or by a mutual object so that traffic light functions can also be implemented, for example, depending on a limiting value by means of an LED.

General:

A bus coupling unit is already permanently integrated in the device, allowing the device to be connected directly to the bus cable during commissioning. The switch-over to the programming mode takes place by simultaneously pressing the buttons 1 and 17 (figure 5). If the device is in programming mode, the programming LED lights up red.

## 1.3 Accessories

Power supply AC 24 V ~

Power supply 320 mA

Power supply 640 mA

Surface-mounted housing

Two-gang appliance box, e.g. Kaiser, Art. no. 9062-02

JUNG Labelling tool – [www.jung-label.de](http://www.jung-label.de)

Art. No. WSSV10

Art. No. 2005 REG

Art. No. 2002 REG

Art. No. EBG 2424

## **2 Installation, electrical connection and operation**

### **2.1 Safety instructions**

**Electrical equipment may only be installed and fitted by electrically skilled persons. The applicable accident prevention regulations must be observed.**

**Before working on the device or exchanging the connected loads, disconnect it from the power supply (switch off the miniature circuit breaker), otherwise there is the risk of an electric shock.**

**Do not use sharp objects for cleaning. Do not use sharp cleaning agents, acids or organic solvents.**

**Failure to observe the instructions may cause damage to the device and result in fire and other hazards.**

**Make sure during the installation that there is always sufficient insulation between the mains voltage and the bus. A minimum distance of at least 4 mm must be maintained between bus conductors and mains voltage cores.**

## 2.2 Device components

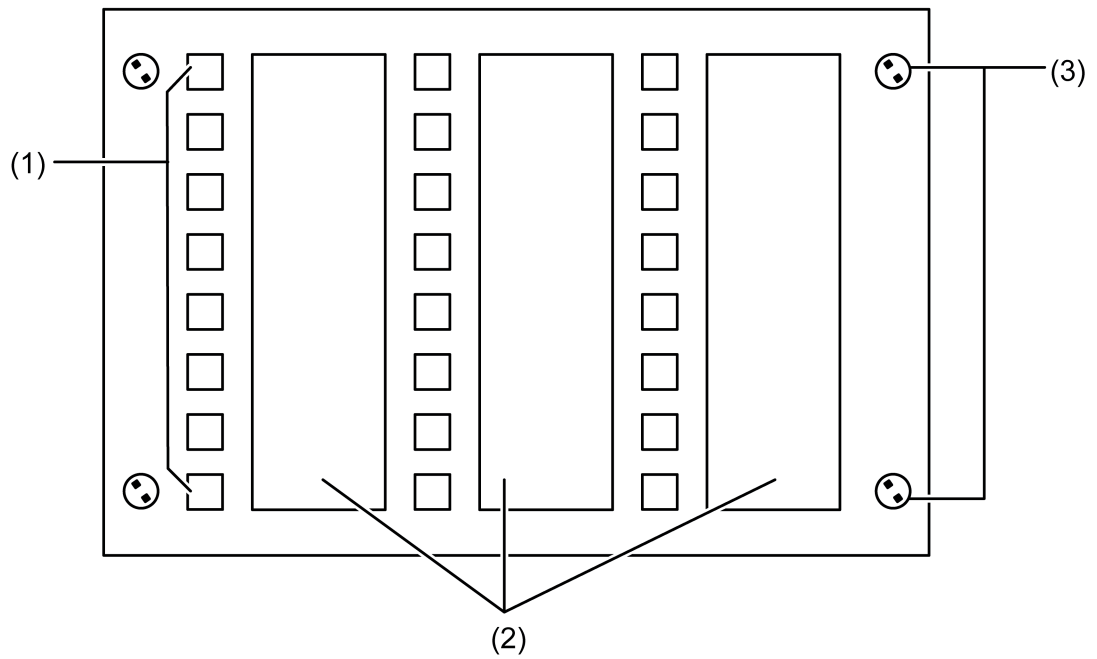


Figure 1: Device components, front side

- (1) Sensor areas with status LED
- (2) Labelling with push-in labels
- (3) Fastening screws, front plate

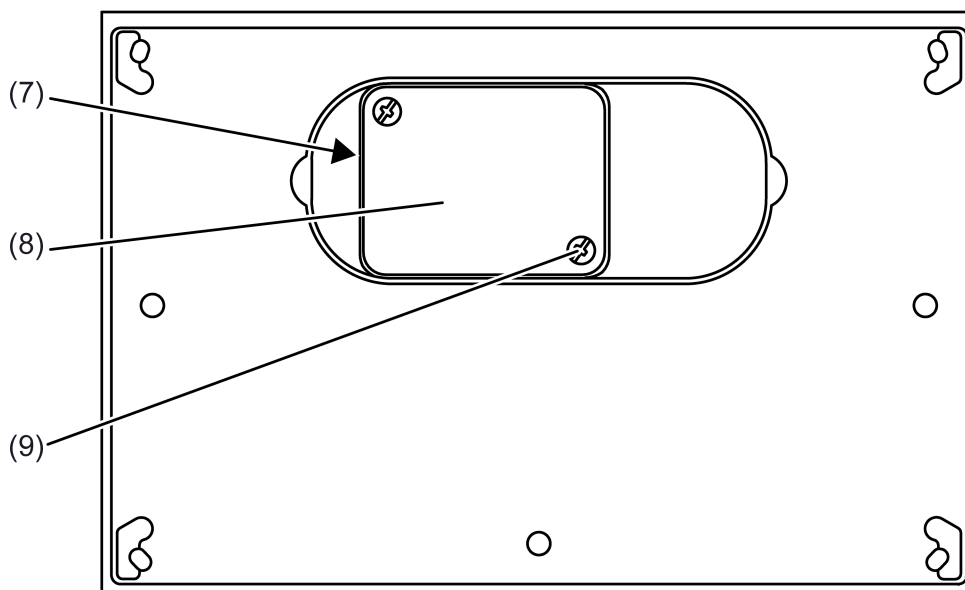


Figure 2: Device components, rear side

- (7) Connection box
- (8) Lid of the connection box
- (9) Cover screw

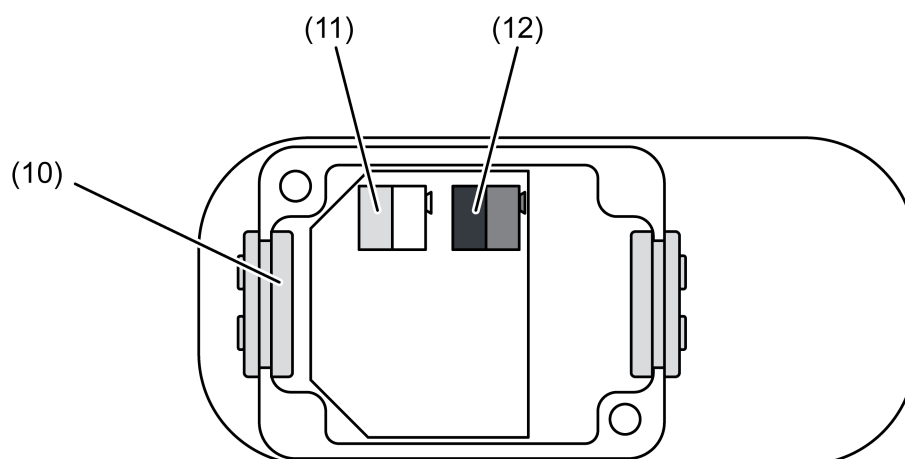


Figure 3: Connection box opened

- (10) Cable penetration
- (11) Connection of the external AC/DC 24 V power supply
- (12) Connection for KNX bus cable



## 2.3 Fitting and electrical connection



### DANGER!

Electrical shock on contact with live parts in the installation environment.

Electrical shocks can be fatal.

Before working on the device, disconnect the power supply and cover up live parts in the working environment.

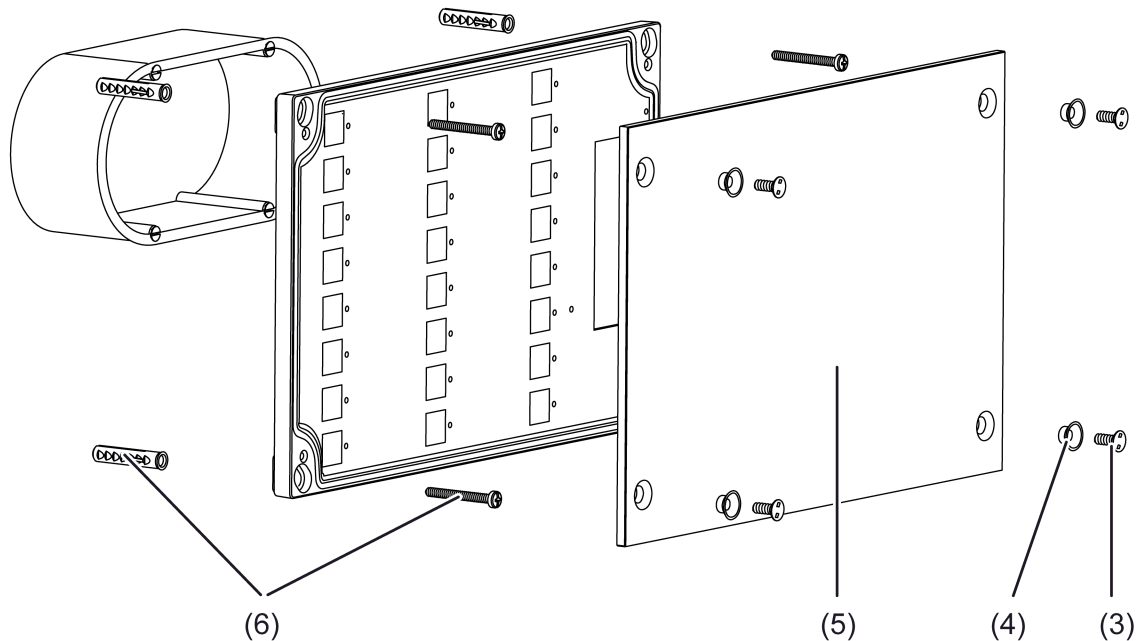


Figure 4: Fitting

- (3) Fastening screw front plate
- (4) Screw sleeve
- (5) Front plate
- (6) Screws/anchor set wall mounting
- (8) Cover connection box

Recommended installation height: 1.50 m.

Installation only using deep concealed appliance box, double.

Only for horizontal installation.

Only install on flat walls.

**i** Soiling in the slots on the front plate, e.g. due to unsuitable pins or paper strips, cannot be removed. The labelling of the sensor areas must be able to resist wipes and moisture and must be made on suitable media, e.g. transparent film.

- Mark the installation location using the enclosed template. Install the concealed appliance box, drill anchor holes for screw fastening on the substrate.
- Slacken the fastening screws (3) with the enclosed tool and remove the front plate (5).
- Slacken the lid screws (9) on the connection box (7) and remove the lid (8).
- Run the connecting cables through the openings (10) of the connection box.
- Connect the bus wires with bus terminal to the bus connection (12).
- Connect the power supply with the yellow and white terminal to the connection (11).

- i** The second wire pair of the bus cable can be used to provide the power supply.
  - Close the lid (8).
- i** Note the physical address on the rear side of the device and behind the front plate. If possible, load the physical address into the device before the final installation.
  - Place the device in the concealed box and screw it to the substrate. Use the enclosed screw/anchor set (6).
  - Push the written push-in labels into the top of the intended slots on the rear side of the front plate.
  - Attach the front plate again. Insert the screw sleeves (4) and fasten them to the housing using the fastening screws (3).

## 2.4 Commissioning

After the device has been connected to the bus and mounted on the wall, it can be put into operation. The commissioning is essentially confined to the programming of the actuator via the ETS.

- i** After connecting the power supply, the programming LED (13) lights up. During this time (approx. 10 seconds), the device has no function.

### Assignment of the physical address



#### **DANGER!**

**Electrical shock on contact with live parts in the installation environment.  
Electrical shocks can be fatal.**

**Before working on the device, disconnect the power supply and cover up live parts in the working environment.**

The device has an integrated bus coupling unit. It does not have a separate programming button, but has a programming LED. The programming mode is activated by pressing the sensor buttons 1 and 17 simultaneously and signalled by the programming LED. The physical address is programmed as described below.

The device must have been connected and ready for use.

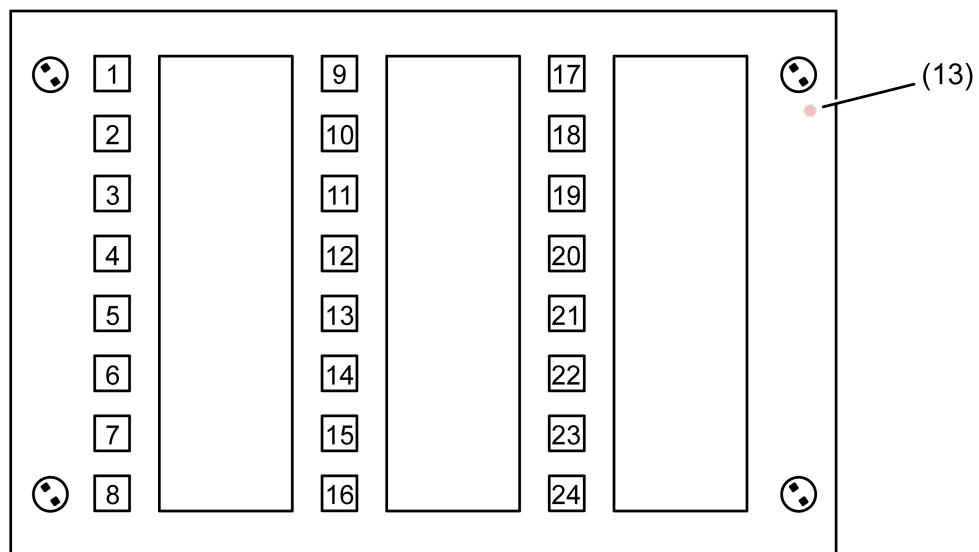


Figure 5: Numbering of the sensor buttons

#### (13) Programming LED

- Activate Programming mode. To do this, press the sensor buttons 1 and 17 simultaneously. Programming mode is activated. The programming LED (13) lights up red.
- Program the physical address with the help of the ETS.  
The programming LED goes out.

- i** If Programming mode is to be activated or deactivated in a device which is already programmed with a valid application, there is the possibility that telegrams will be transmitted to the bus at the time the button is pressed. The telegram transmitted depends on the button function programmed.

### **Programming the application**

The application must then be programmed into the device with the help of the ETS. The ETS3.0 from version "d" onwards or the ETS4 detects automatically whether a valid application has already been programmed into the device before. To reduce the programming time, the ETS downloads the whole application only if the device was programmed beforehand with another application or with no application at all. In all other cases, the ETS makes a time-optimised partial download in which only the modified data is loaded into the device. For commissioning, the ETS3.0 from Version d Patch A onwards or the ETS4 is required.

If no application has been loaded, the programming LED flashes green with a frequency of about 2 Hz to indicate an error. The device then does not work.

An LED test is started by simultaneously pressing the buttons 1 and 24 (figure 5). In this case, all the LEDs light up in sequence for 2 seconds each in red, green and blue.

### **Delivery state**

In the state as delivered, no telegrams are transmitted to the bus, there is no response to the bus. The programming LED flashes green.

## 2.5 Operation

The front end of the device consists of 24 control surfaces (1) functionally separated from each other, 24 three-colour LEDs (14) and a programming LED (13). The capacitive control surfaces are behind the screwed on design glass plate and enable operation of the device pushbutton functions (figure 6).

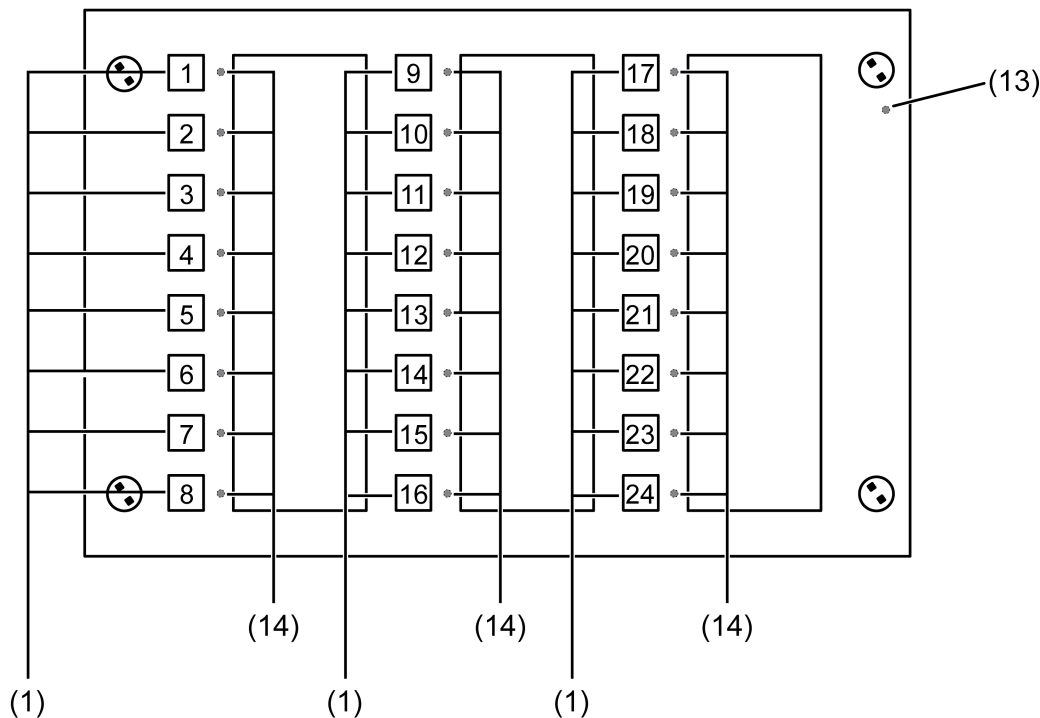


Figure 6: Arrangement of the control surfaces and the LED

- (1) Capacitive operating elements
- (13) Programming LED
- (14) Three-colour LED (status LED)

The control surfaces are allocated to the pushbutton sensor function. The function of these buttons can also be configured in the ETS to any desired pushbutton sensor function (switching, dimming, shutter ...). Alternatively, the display function of the status LED can also be set.

The push button sensor function is an independent function section of the device with its own parameter blocks in the ETS. For a more detailed description of the operating functions, please see Chapter 4 of this documentation.

The operation concept of an operating area is predefined as a push-button function.

On the right hand side next to the control surfaces are the three-colour Status-LEDs. These status LEDs can be internally connected to the operating function according to the function of the rocker or pushbuttons, thus indicating the operating status directly. They may, however, also be used for indicating completely independent display information in red, green and blue, or can also flash or be permanently on or off. In addition, the LEDs can always be switched on or off via the parameterization.

In the case of the internal connection to the push-button function, the status LEDs can be used as a button-press display of the corresponding button depending on the button or push-button function and as status displays of the corresponding push-button function for the switching and

dimming functions of the switching object. In 2-channel operation, as telegram acknowledgment as well.

Irrespective of the push-button function, each colour (red, green, blue) of the status LEDs can be controlled via a separate object.

A kind of 'traffic light' (red, blue, green) can be implemented for the limiting value function, which signals three states depending on an internally configured and externally received limiting value. The current state of a comparator "less than reference value", "equal to reference value" or "greater than reference value" is assigned to one colour each and displayed via the status LED. Furthermore, they can indicate faults or the status of external room temperature controllers.

## 3 Technical data

### General

Protection class	III
Degree of protection	IP 54
Mark of approval	KNX/EIB
Ambient temperature	-20 ... +70 °C
Storage/transport temperature	-20 ... +75 °C
Relative humidity	15 ... 95 % (No moisture condensation)

### External supply

Rated voltage	AC/DC 24 V SELV
Rated frequency	50 / 60 Hz
Power consumption	max. 2.2 W
Connection of power supply single stranded	Connection terminal 0.6 ... 0.8 mm <sup>2</sup>

### KNX supply

KNX medium	TP 256
Commissioning mode	S-mode
Rated voltage KNX	DC 21 ... 32 V SELV
Current consumption KNX	max. 5 mA
Connection, Bus	device connection terminal

### Dimensions front plate

Dimensions W×H×D	approx. 236x156x14 mm
Installation depth	approx. 39 mm

## 4 Software description

### 4.1 Software "MBT 2424 501311"

#### 4.1.1 Software specification

ETS search paths:	Display / Signal panel / MBT 2424
BAU used:	TPUART + $\mu$ C
KNX type class:	3b device with cert. Physical layer + stack
Configuration:	S-mode standard
PEI type:	"00" <sub>Hex</sub> / "0" <sub>Dec</sub>
PEI connector:	No connector

#### Application program:

No.	Short description	Name	Version	from mask version
1	Multifunctional indication and operating panel Up to 24 sensor areas for the pushbutton functions, up to 24 three-colour LEDs for visualization and a Piezo.	MBT 2424 501311	1.1 for ETS3.0d onwards and ETS 4.0	705



## 4.1.2 Scope of functions

### Scope of functions

- The panel is used to control building functions and to display building statuses of various features.
  
- 24 freely programmable three-colour LEDs can be used for the visualization of various building states, fault messages and a dismantling alarm. Each colour of the LED has a separate KNX communication object. The LED can be addressed or assigned to this irrespective of the sensor button.
  
- The following functions can be assigned to each sensor area: Switching, dimming, venetian blind, value, scene or acknowledgment. These function areas can be linked directly to the function of the LED or can also trigger independent functions.
  
- Each LED can
  - always be switched ON or OFF,
  - display the operation of the corresponding button
  - be controlled separately via three objects (one for each LED colour),
  - indicate the state of the switching object of the push-button function,
  - display the controller status or operating mode of an external controller,
  - display the state of comparators,
  - visualize limiting values,
  - display faults.
  
- Up to 8 light scenes with a total of 8 actuator groups can be recalled or stored.
  
- Up to 5 logic operations with up to 8 inputs each.
  
- A maximum of 5 timers with switch-on and switch-off delay and filter function are available.
  
- The built-in Piezo buzzer can be used to generate an acoustic 'button click' for acoustically indicating acknowledgement of a sensor activation, fault messages and dismantling alarm. Up to 4 different sound patterns consisting of the pitch, sound interval with adjustable pulse and pause relation and volume can be set for the signals.
  
- The configuration takes place via a fully integrated ETS plug-in. The commissioning takes place easily via the programming function of the ETS.


## 4.1.3 Object table

Number of communication objects:	236
Number of addresses (max):	254
Number of assignments (max):	255
Dynamic table management	No


### 4.1.3.1 General

#### General objects


---

Function:	Controller status indication KNX compliant				
Object	Function	Name	Type	DPT	Flag
 <sup>235</sup>	Input KNX controller status	B.Controller status	2 byte	22.101	C, W, T, (R) <sup>1</sup>
Description	2-byte objec via which the panel receives the status of the external controller.				

---

Function:	Controller status indication general				
Object	Function	Name	Type	DPT	Flag
 <sup>235</sup>	Input controller status	B.Controller status	1 byte	--- <sup>2</sup>	C, W, T, (R) <sup>1</sup>
Description	1-byte object via which the panel receives the status of the external controller.				

---

Function:	Controller status indication general				
Object	Function	Name	Type	DPT	Flag
 <sup>236</sup>	Input status signal addition	B.Controller status	1 byte	--- <sup>3</sup>	C, W, T, (R) <sup>1</sup>
Description	1-byte object via which the panel receives further statuses of the external controller in addition to the object 235.				

1: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.


2: Non-standardised DP type (in accordance with KNX AN 097/07 rev 3).

3: Non-standardised DP type.


## 4.1.3.2 Button objects

### Objects for button function


---

Function:	Switching				
Object	Function	Name	Type	DPT	Flag
 0... 23	Switching	B.button 1 <sup>1</sup>	1-bit	1.xxx	C, W, T, (R) <sup>2</sup>
Description	1-bit object for transmission of switching telegrams (ON, OFF).				


---

Function:	Dimming				
Object	Function	Name	Type	DPT	Flag
 0... 23	Switching	B.button 1 <sup>1</sup>	1-bit	1.xxx	C, W, T, (R) <sup>2</sup>
Description	1-bit object for transmission of switching telegrams (ON, OFF).				


---

Function:	Dimming				
Object	Function	Name	Type	DPT	Flag
 30... 53	Dimming	B.button 1 <sup>1</sup>	4-bit	3.007	C, W, T, (R) <sup>2</sup>
Description	4-bit object for the transmission of relative dimming telegrams.				

---

Function:	Venetian blind				
Object	Function	Name	Type	DPT	Flag
 0... 23	Short time operation	B.button 1 <sup>1</sup>	1-bit	1.007	C, -, T, (R) <sub>2</sub>
Description	1-bit object for the transmission of telegrams with which a Venetian blind or shutter drive motor can be stopped or with which the blind slats can be adjusted by short time operation.				

---

Function:	Venetian blind				
Object	Function	Name	Type	DPT	Flag
 30... 53	Long-time operation	B.button 1 <sup>1</sup>	1-bit	1.008	C, W, T, (R) <sup>2</sup>
Description	1-bit object for the transmission of telegrams with which a Venetian blind or shutter drive motor can be moved upwards or downwards.				

1: The objects have been described for sensor button 1 as an example. The objects for the other sensor buttons are defined in the same way by shifting the object number and changing the object name.

2: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

Function: 1-byte value transmitter

Object	Function	Name	Type	DPT	Flag
<input type="checkbox"/> ← 0... 23	Value	B.button 1 <sup>1</sup>	1 byte	5.001, 5.005, 5.010	C, W, T, (R) <sup>2</sup>

Description 1-byte object for the transmission of values from 0 to 255 (corresponding to values from 0 % to 100 %). If the adjustment of the value is enabled, the object can transmit telegrams cyclically after long actuation with which the value can be reduced or increased by a presettable amount.

Function: 2-byte value transmitter

Object	Function	Name	Type	DPT	Flag
<input type="checkbox"/> ← 0... 23	Value	B.button 1 <sup>1</sup>	2 byte	7.001	C, W, T, (R) <sup>2</sup>

Description 2-byte object for the transmission of values from 0 to 65535. If the adjustment of the value is enabled, the object can transmit cyclical telegrams after a long press with which the value can be reduced or increased by an adjustable amount.

Function: 2-byte value transmitter

Object	Function	Name	Type	DPT	Flag
<input type="checkbox"/> ← 0... 23	Temperature value	B.button 1 <sup>1</sup>	2 byte	9.001	C, W, T, (R) <sup>2</sup>

Description 2 -byte object for the transmission of a temperature value from 0 °C to 40 °C. If the adjustment of the value is enabled, the object can transmit telegrams cyclically after a long press with which the value can be reduced or increased by 1 K.

Function: 2-byte value transmitter

Object	Function	Name	Type	DPT	Flag
<input type="checkbox"/> ← 0... 23	Brightness value	B.button 1 <sup>1</sup>	2 byte	9.004	C, W, T, (R) <sup>2</sup>

Description 2-byte object for the transmission of a brightness level value from 0 to 1500 lux. If the adjustment of the value is enabled, the object can transmit cyclical telegrams after a long press with which the value can be reduced or increased by 50 lux.

Function: Scene extension


Object	Function	Name	Type	DPT	Flag
<input type="checkbox"/> ← 0... 23	Scene extension	B.button 1 <sup>1</sup>	1 byte	18.001	C, -, T, (R) <sub>2</sub>

Description 1-byte object for recalling or for storing one of 64 scenes max. from a scene push button sensor.

1: The objects have been described for sensor button 1 as an example. The objects for the other sensor buttons are defined in the same way by shifting the object number and changing the object name.


2: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 0... 23	Channel 1 switching	B.button 1 <sup>1</sup>	1-bit	1.xxx	C, W, T, (R) <sup>2</sup>


Description 1-bit object for the transmission of switching telegrams, if 2-channel operation is activated.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 0... 23	Channel 1 value	B.button 1 <sup>1</sup>	1 byte	5.001, 5.005, 5.010	C, -, T, (R) <sup>2</sup>


Description 1-byte object for the transmission of value telegrams, if 2-channel operation is activated.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 0... 23	Channel 1 temperature value	B.button 1 <sup>1</sup>	2 byte	9.001	C, -, T, (R) <sup>2</sup>


Description 2-byte object for the transmission of temperature value telegrams if 2-channel operation is activated.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 30... 53	Channel 2 switching	B.button 1 <sup>1</sup>	1-bit	1.xxx	C, W, T, (R) <sup>2</sup>

Description 1-bit object for the transmission of switching telegrams, if 2-channel operation is activated.

Function: 2-channel operation


Object	Function	Name	Type	DPT	Flag
 30... 53	Channel 2 value	B.button 1 <sup>1</sup>	1 byte	5.001, 5.005, 5.010	C, -, T, (R) <sup>2</sup>

Description 1-byte object for the transmission of value telegrams, if 2-channel operation is activated.

1: The objects have been described for sensor button 1 as an example. The objects for the other sensor buttons are defined in the same way by shifting the object number and changing the object name.

2: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.


Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 30... 53	Channel 2 temperature value	B.button 1 <sup>1</sup>	2 byte	9.001	C, -, T, (R) <sub>2</sub>

Description 2-byte object for the transmission of temperature value telegrams if 2-channel operation is activated.


## Objects for status LED

Function: Separate control of status-LED red

Object	Function	Name	Type	DPT	Flag
 60, 63, 66... 129	Status-LED red	B.LED 1 <sup>3</sup>	1-bit	1.001	C, W, -, (R) <sub>4</sub>


Description 1-bit object for activation of the red status LED.

Function: Separate control of status-LED green

Object	Function	Name	Type	DPT	Flag
 61, 64, 67... 130	Status LED green	B.LED 1 <sup>3</sup>	1-bit	1.001	C, W, -, (R) <sub>4</sub>

Description 1-bit object for activation of the green status LED.

Function: Separate control of status-LED blue

Object	Function	Name	Type	DPT	Flag
 62, 65, 68... 131	Status LED blue	B.LED 1 <sup>3</sup>	1-bit	1.001	C, W, -, (R) <sub>4</sub>


Description 1-bit object for activation of the blue status LED.


1: The objects have been described for sensor button 1 as an example. The objects for the other sensor buttons are defined in the same way by shifting the object number and changing the object name.


2: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.


3: The objects for the LED are described on the left next to the sensor button 1 as an example. The objects next to the other sensor buttons are defined in the same way by shifting the object number and changing the object name.


4: For reading, the R-flag must be set. The last value written to the object via the bus will be read.

Function: Status LED for operating mode display (KNX controller)					
Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Operating mode red	B.LED 1 <sup>1</sup>	1 byte	20.102	C, W, -, (R) <sub>2</sub>
Description      Object via which the panel receives the controller operating mode (automatic, comfort, standby, night, frost/heat protection mode) that should be displayed by the red status LED.					

Function: Status LED for operating mode display (KNX controller)					
Object	Function	Name	Type	DPT	Flag
 61, 64, 67 ... 130	Operating mode green	B.LED 1 <sup>1</sup>	1 byte	20.102	C, W, -, (R) <sub>2</sub>
Description      Object via which the panel receives the controller operating mode (automatic, comfort, standby, night, frost/heat protection mode) that should be displayed by the green status LED.					

Function: Status LED for operating mode display (KNX controller)					
Object	Function	Name	Type	DPT	Flag
 62, 65, 68 ... 131	Operating mode blue	B.LED 1 <sup>1</sup>	1 byte	20.102	C, W, -, (R) <sub>2</sub>
Description      Object via which the panel receives the controller operating mode (automatic, comfort, standby, night, frost/heat protection mode) that should be displayed by the blue status LED.					


Function: 1 byte comparator without sign					
Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Comparator red	B.LED 1 <sup>1</sup>	1 byte	5.xxx	C, W, -, (R) <sub>2</sub>
Description      1-byte object via which the panel receives the reference value between 0 and 255, which is compared with a configurable value. The result is displayed by the red status LED.					

Function: 1 byte comparator without sign					
Object	Function	Name	Type	DPT	Flag
 61, 64, 67 ... 130	Comparator green	B.LED 1 <sup>1</sup>	1 byte	5.xxx	C, W, -, (R) <sub>2</sub>
Description      1 byte object via which the panel receives the reference value between 0 and 255, which is compared with a configurable value. The result is displayed by the green status LED.					

1: The objects for the LED are described on the left next to the sensor button 1 as an example. The objects next to the other sensor buttons are defined in the same way by shifting the object number and changing the object name.


2: For reading, the R-flag must be set. The last value written to the object via the bus will be read.

Function: 1 byte comparator without sign

Object	Function	Name	Type	DPT	Flag
 62, 65, 68 ... 131	Comparator blue	B.LED 1 <sup>1</sup>	1 byte	5.xxx	C, W, -, (R) <sub>2</sub>


Description 1 byte object via which the panel receives the reference value between 0 and 255, which is compared with a configurable value. The result is displayed by the blue status LED.

Function: 2 byte comparator without sign

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Comparator red	B.LED 1 <sup>1</sup>	2 byte	7.001	C, W, -, (R) <sub>2</sub>


Description 2 byte object via which the panel receives the reference value between 0 and 65535, which is compared with a configurable value. The result is displayed by the red status LED.

Function: 2 byte comparator without sign

Object	Function	Name	Type	DPT	Flag
 61, 64, 67 ... 130	Comparator green	B.LED 1 <sup>1</sup>	2 byte	7.001	C, W, -, (R) <sub>2</sub>


Description 2-byte object via which the panel receives the reference value between 0 and 65535, which is compared with a configurable value. The result is displayed by the green status LED.

Function: 2 byte comparator without sign

Object	Function	Name	Type	DPT	Flag
 62, 65, 68 ... 131	Comparator blue	B.LED 1 <sup>1</sup>	2 byte	7.001	C, W, -, (R) <sub>2</sub>

Description 2-byte object via which the panel receives the reference value between 0 and 65535, which is compared with a configurable value. The result is displayed by the blue status LED.

Function: 1 byte comparator with sign

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Comparator red	B.LED 1 <sup>1</sup>	1 byte	6.xxx	C, W, -, (R) <sub>2</sub>


Description 1-byte object via which the panel receives the reference value between -128 and 127, which is compared with a configurable value. The result is displayed by the red status LED.

1: The objects for the LED are described on the left next to the sensor button 1 as an example. The objects next to the other sensor buttons are defined in the same way by shifting the object number and changing the object name.

2: For reading, the R-flag must be set. The last value written to the object via the bus will be read.




Function: 1 byte comparator with sign

Object	Function	Name	Type	DPT	Flag
 61, 64, 67 ... 130	Comparator green	B.LED 1 <sup>1</sup>	1 byte	6.xxx	C, W, -, (R) <sub>2</sub>


Description 1-byte object via which the panel receives the reference value between -128 and 127, which is compared with a configurable value. The result is displayed by the green status LED.

Function: 1 byte comparator with sign

Object	Function	Name	Type	DPT	Flag
 62, 65, 68 ... 131	Comparator blue	B.LED 1 <sup>1</sup>	1 byte	6.xxx	C, W, -, (R) <sub>2</sub>


Description 1-byte object via which the panel receives the reference value between -128 and 127, which is compared with a configurable value. The result is displayed by the blue status LED.

Function: 2 byte comparator with sign

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Comparator red	B.LED 1 <sup>1</sup>	2 byte	8.001	C, W, -, (R) <sub>2</sub>


Description 2-byte object via which the panel receives the reference value between -32768 and 32767, which is compared with a configurable value. The result is displayed by the red status LED.

Function: 2 byte comparator with sign

Object	Function	Name	Type	DPT	Flag
 61, 64, 67 ... 130	Comparator green	B.LED 1 <sup>1</sup>	2 byte	8.001	C, W, -, (R) <sub>2</sub>

Description 2-byte object via which the panel receives the reference value between -32768 and 32767, which is compared with a configurable value. The result is displayed by the green status LED.

Function: 2 byte comparator with sign


Object	Function	Name	Type	DPT	Flag
 62, 65, 68 ... 131	Comparator blue	B.LED 1 <sup>1</sup>	2 byte	8.001	C, W, -, (R) <sub>2</sub>

Description 2-byte object via which the panel receives the reference value between -32768 and 32767, which is compared with a configurable value. The result is displayed by the blue status LED.

1: The objects for the LED are described on the left next to the sensor button 1 as an example. The objects next to the other sensor buttons are defined in the same way by shifting the object number and changing the object name.


2: For reading, the R-flag must be set. The last value written to the object via the bus will be read.

Function: Limit value display value range 0 to 255

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Limit value display	B.LED 1 <sup>1</sup>	1 byte	5.010	C, W, -, (R) <sub>2</sub>


Description 1-byte object via which the panel receives the reference value to be monitored within the value range from 0 to 255, which is compared with two configurable limiting values. The result is displayed by the three colours red, green or blue of the status LED.

Function: Limit value display value range 0% to 100%

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Limit value display	B.LED 1 <sup>1</sup>	1 byte	5.001	C, W, -, (R) <sub>2</sub>


Description 1-byte object via which the panel receives the reference value to be monitored within the value range from 0% to 100%, which is compared with two configurable limiting values. The result is displayed by the three colours red, green or blue of the status LED.

Function: Limit value display temperature

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Limit value display	B.LED 1 <sup>1</sup>	2 byte	9.001	C, W, -, (R) <sub>2</sub>

Description 2-byte object via which the panel receives the reference value to be monitored within the value range from -273°C to 670760°C, which are compared with two configurable limiting values. The result is displayed by the three colours red, green or blue of the status LED.

Function: Limit value display brightness


Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Limit value display	B.LED 1 <sup>1</sup>	2 byte	9.004	C, W, -, (R) <sub>2</sub>

Description 2-byte object via which the panel receives the brightness to be monitored within the value range from 0 Lux to 670760 Lux, which is compared with two configurable limiting values. The result is displayed by the three colours red, green or blue of the status LED.

1: The objects for the LED are described on the left next to the sensor button 1 as an example. The objects next to the other sensor buttons are defined in the same way by shifting the object number and changing the object name.


2: For reading, the R-flag must be set. The last value written to the object via the bus will be read.

Function: Limit value display wind speed

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Limit value display	B.LED 1 <sup>1</sup>	2 byte	9.005	C, W, -, (R) <sub>2</sub>


Description: 2-byte object via which the panel receives the brightness to be monitored within the value range from 0 m/s to 670760 m/s, which is compared with two configurable limiting values. The result is displayed by the three colours red, green or blue of the status LED.

Function: Limit value display value range 0 to 65535

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Limit value display	B.LED 1 <sup>1</sup>	2 byte	7.001	C, W, -, (R) <sub>2</sub>


Description: 2-byte object via which the panel receives the reference value to be monitored within the value range from 0 to 665535, which is compared with two configurable limiting values. The result is displayed by the three colours red, green or blue of the status LED.

Function: Input fault message

Object	Function	Name	Type	DPT	Flag
 60, 63, 66 ... 129	Fault message input	B.LED 1 <sup>1</sup>	1-bit	1.001	C, W, -, (R) <sub>2</sub>

Description: 1-bit object, via which the panel receives the fault message. The active fault message is displayed by the status LED flashing in red.

Function: Acknowledgement fault message

Object	Function	Name	Type	DPT	Flag
 151 ... 174	Fault message acknowledgement	B.LED 1 <sup>1</sup>	1-bit	1.001	C, W, -, (R) <sub>2</sub>

Description: 1-bit object via which the panel sends the acknowledgement or receives the external acknowledgement of the fault message. The acknowledgement is displayed continuously in red by the status LED.


1: The objects for the LED are described on the left next to the sensor button 1 as an example. The objects next to the other sensor buttons are defined in the same way by shifting the object number and changing the object name.

2: For reading, the R-flag must be set. The last value written to the object via the bus will be read.


## 4.1.3.3 Disabling objects

### Objects for disabling functions


---

Function:	Switching				
Object	Function	Name	Type	DPT	Flag
 24 ... 29	Switching	B.Disabling function 1 to 6	1-bit	1.xxx	C, W, T, (R) <sup>1</sup>
Description	1-bit object for transmission of switching telegrams (ON, OFF).				


---

Function:	Dimming				
Object	Function	Name	Type	DPT	Flag
 24 ... 29	Switching	B.Disabling function 1 to 6	1-bit	1.xxx	C, W, T, (R) <sup>1</sup>
Description	1-bit object for transmission of switching telegrams (ON, OFF).				


---

Function:	Dimming				
Object	Function	Name	Type	DPT	Flag
 54 ... 59	Dimming	B.Disabling function 1 to 6	4-bit	1.007	C, W, T, (R) <sup>1</sup>
Description	4-bit object for the transmission of relative dimming telegrams.				

---


Function:	Venetian blind				
Object	Function	Name	Type	DPT	Flag
 24 ... 29	Short time operation	B.Disabling function 1 to 6	1-bit	1.007	C, -, T, (R) 1
Description	1-bit object for the transmission of telegrams with which a Venetian blind or shutter drive motor can be stopped or with which the blind slats can be adjusted by short time operation.				

---

Function:	Venetian blind				
Object	Function	Name	Type	DPT	Flag
 54 ... 59	Long-time operation	B.Disabling function 1 to 6	1-bit	1.008	C, W, T, (R) <sup>1</sup>
Description	1-bit object for the transmission of telegrams with which a Venetian blind or shutter drive motor can be moved upwards or downwards.				


1: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

Function: 1-byte value transmitter

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Value	B.Disabling function 1 to 6	1 byte	5.001,- 5.005,- 5.010	C, W, T, (R) <sup>1</sup>


Description 1-byte object for the transmission of values from 0 to 255 (corresponding to values from 0 % to 100 %). If the adjustment of the value is enabled, the object can transmit telegrams cyclically after long actuation with which the value can be reduced or increased by a presettable amount.

Function: 2-byte value transmitter

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Value	B.Disabling function 1 to 6	2 byte	7.001	C, W, T, (R) <sup>1</sup>


Description 2-byte object for the transmission of values from 0 to 65535. If the adjustment of the value is enabled, the object can transmit cyclical telegrams after a long press with which the value can be reduced or increased by an adjustable amount.

Function: 2-byte value transmitter

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Temperature value	B.Disabling function 1 to 6	2 byte	9.001	C, W, T, (R) <sup>1</sup>


Description 2 -byte object for the transmission of a temperature value from 0 °C to 40 °C. If the adjustment of the value is enabled, the object can transmit telegrams cyclically after a long press with which the value can be reduced or increased by 1 K.

Function: 2-byte value transmitter

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Brightness value	B.Disabling function 1 to 6	2 byte	9.004	C, W, T, (R) <sup>1</sup>

Description 2-byte object for the transmission of a brightness level value from 0 to 1500 lux. If the adjustment of the value is enabled, the object can transmit cyclical telegrams after a long press with which the value can be reduced or increased by 50 lux.


Function: Scene extension

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Scene extension	B.Disabling function 1 to 6	1 byte	18.001	C, -, T, (R) 1

Description 1-byte object for recalling or for storing one of 64 scenes max. from a scene push button sensor.


1: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Channel 1 switching	B.Disabling function 1 to 6	1-bit	1.xxx	C, W, T, (R) <sup>1</sup>


Description 1-bit object for the transmission of switching telegrams, if 2-channel operation is activated.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Channel 1 value	B.Disabling function 1 to 6	1 byte	5.001,- 5.005,- 5.010	C, -, T, (R) <sup>1</sup>


Description 1-byte object for the transmission of value telegrams within a range from 0 to 255 if 2-channel operation is activated.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 24 ... 29	Channel 1 temperature value transmitter	B.Disabling function 1 to 6	2 byte	9.001	C, -, T, (R) <sup>1</sup>


Description 2-byte object for the transmission of temperature telegrams within a range from 0°C to 40°C if 2-channel operation is activated.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 54 ... 59	Channel 2 switching	B.Disabling function 1 to 6	1-bit	1.xxx	C, W, T, (R) <sup>1</sup>


Description 1-bit object for the transmission of switching telegrams, if 2-channel operation is activated.

Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 54 ... 59	Channel 2 value	B.Disabling function 1 to 6	1 byte	5.001,- 5.005,- 5.010	C, -, T, (R) <sup>1</sup>

Description 1-byte object for the transmission of value telegrams within a range from 0 to 255 if 2-channel operation is activated.


Function: 2-channel operation

Object	Function	Name	Type	DPT	Flag
 54 ... 59	Channel 2 temperature value transmitter	B.Disabling function 1 to 6	2 byte	9.001	C, -, T, (R) <sup>1</sup>

Description 2-byte object for the transmission of temperature telegrams within a range from 0°C to 40°C if 2-channel operation is activated.


1: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

Function: Disabling function of entire panel

Object	Function	Name	Type	DPT	Flag
 132	Disabling	B.Disable buttons	1-bit	1.001	C, W, -, (R) 1


Description 1-bit object via which all buttons of the panel can be disabled and enabled again (polarity configurable).

Function: Disabling function left column

Object	Function	Name	Type	DPT	Flag
 132	Disabling	B.Disable column 1	1-bit	1.001	C, W, -, (R) 1


Description 1-bit object via which the buttons of the left column of the panel can be disabled and enabled again (polarity configurable).

Function: Disabling function middle column

Object	Function	Name	Type	DPT	Flag
 133	Disabling	B.Disable column 2	1-bit	1.001	C, W, -, (R) 1

Description 1-bit object via which the buttons of the middle column of the panel can be disabled and enabled again (polarity configurable).

Function: Disabling function right column

Object	Function	Name	Type	DPT	Flag
 134	Disabling	B.Disable column 1	1-bit	1.001	C, W, -, (R) 1


Description 1-bit object via which the buttons of the right column of the panel can be disabled and enabled again (polarity configurable).

1: For reading, the R-flag must be set. The last value written to the object via the bus will be read.


## 4.1.3.4 Alarm message objects

### Objects of the dismantling alarm

---

Function:	Dismantling alarm				
Object	Function	Name	Type	DPT	Flag
 <sup>150</sup>	Switching	B.Dismantling alarm	1-bit	1.001	C, -, T, (R) 1
Description	1-bit object for transmitting the one alarm signalling (polarity configurable) after the end of the transmit delay when dismantling the glass front.				

---


Function:	Dismantling alarm				
Object	Function	Name	Type	DPT	Flag
 <sup>150</sup>	Value	B.Dismantling alarm	1 byte	5.010	C, -, T, (R) 1
Description	1-byte object for transmitting the one alarm signalling (value configurable) after the end of the transmit delay during dismantling of the glass front.				


1: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.



## 4.1.3.5 Objects of the logic gate

### Objects of the logic gate

Function: Input logic gate					
Object	Function	Name	Type	DPT	Flag
 175 ... 182, 184 ... 191, 193 ... 200, 202 ... 209, 211 ... 218	Input 1 to max. 8	L.Logic gate 'n'	1-bit	1.001	C, W, -, (R) 1
Description	1-bit objects for externally controlling up to eight logic inputs of the logic gate $n$ ( $n$ = number of activated logic gates 0 to 5). Depending on the parameterisation, the inputs can be linked normally inverted 'OR', 'AND', 'exclusive OR' as well as 'AND with return'.				


Function: Output logic gate					
Object	Function	Name	Type	DPT	Flag
 183, 192, 201, 210, 219	Output	L.Logicgate 'n'	1-bit	1.001	C, W, T, (R) <sup>2</sup>
Description	1-bit output object of the logic gate $n$ ( $n$ = number of activated logic gates 0 to 5). The result of the logic operation can be output normally or inverted. A transmission after each input event or after each status change of the output as well as the transmission of only ON or only OFF telegrams can be configured.				

1: For reading, the R-flag must be set. The last value written to the object via the bus will be read.


2: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

## 4.1.3.6 Objects of the timers


---

Function:	Timer				
Object	Function	Name	Type	DPT	Flag
 220, 223, 226, 229, 232	Input	L.Timer	1-bit	1.001	C, W, -, (R) <sup>1</sup>
Description	1 bit object, the value of which is forwarded to the output object of the timer 'n' ('n' = number of activated timers 0 to 5), depending on the value of the appropriate disabling object, the set filter function and the defined delays.				

---

Function:	Timer				
Object	Function	Name	Type	DPT	Flag
 221, 224, 227, 230, 233	Output	L.Timer	1-bit	1.001	C, T,- (R) <sup>2</sup>
Description	1 bit object which forwards the logical switching state of the timer output 'n' ('n' = number of activated timers 0 to 5).				

---

Function:	Timer				
Object	Function	Name	Type	DPT	Flag
 222, 225, 228, 231, 234	Disabling	L.Timer	1-bit	1.001	C, W, -, (R) <sup>1</sup>
Description	1 bit object which specifies whether the value of the corresponding input object is forwarded to the output object. The behaviour of the disabling object can be set.				


1: For reading, the R-flag must be set. The last value written to the object via the bus will be read.

2: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

## 4.1.3.7 Scene objects


### Object for light scene function

Function: Scene function switching

Object	Function	Name	Type	DPT	Flag
 141 ... 148	Switching	B.Scene output 1-8	1-bit	1.001	C, W, T, (R), A <sup>1</sup>


Description 1-bit objects for controlling up to eight actuator groups (ON, OFF).

Function: Scene function value

Object	Function	Name	Type	DPT	Flag
 141 ... 148	Value	B.Scene output 1-8	1 byte	5.010	C, W, T, (R), A <sup>2</sup>


Description 1-byte objects for controlling up to eight actuator groups (0...255).

Function: Scene function value/shutter position

Object	Function	Name	Type	DPT	Flag
 141 ... 148	Value	B.Scene output 1-8	1 byte	5.001	C, W, T, (R), A <sup>2</sup>

Description 1-byte objects for controlling up to eight actuator groups (0%...100%).

Function: Light scene function

Object	Function	Name	Type	DPT	Flag
 149	Extension unit input	B.Scene	1 byte	18.001	C, W, -, (R) 1

Description 1-byte object with which one of the eight internally stored scenes can be recalled or stored again.

1: For reading, the R-flag must be set. The last value written to the object via the bus will be read.

2: For reading, the R-flag must be set. The last value written to the object via the bus or by the device will be read.

## 4.1.4 Functional description

### 4.1.4.1 Push button function

#### 4.1.4.1.1 "Switching/pushing" button function

For each button with the function set to "Switching", the ETS indicates a 1-bit communication object. The parameters of the button permit fixing the value this object is to adopt on pressing and / or on releasing (No reaction, ON, OFF, TOGGLE – toggling of the object value). Thus, it is possible to set whether it is a switch-on or switch-off telegram. There is another parameterisation option, the Toggle (Buttons) option in order to be able to switch ON and OFF. With this option, the assigned group is switched on if it was switched off during the operation and vice versa.

No distinction is made between a brief or long press.

- i** After about 30 seconds an internal calibration is performed. This cancels the detection of a longer pressing.

The status LEDs can be configured independently . (see chapter 4.1.4.2. LED functions)

## 4.1.4.1.2 Button function "Dimming"

For each button with the function set to "Dimming", the ETS indicates a 1-bit object and a 4-bit object. Generally, the panel transmits a switching telegram after a brief press (see chapter 4.1.4.1.1. "Switching/pushing" button function) and a dimming telegram after a long press. In the standard parameterisation it transmits a telegram for stopping the dimming action after a long press. The time needed by the panel to detect an actuation as a long actuation can be set in the parameters.

Whether with the button-press the lighting is switched On, Off or toggled or whether it is dimmed brighter or darker is equally dependent on the setting, just as the time that must elapse until the dimming telegram is triggered instead of a switching telegram. Furthermore, the settings for this function can also determine the area that should be dimmed, whether a telegram should be transmitted for stopping the dimming process when releasing the button or whether the dimming telegram should be transmitted repeatedly.

The status LEDs can be configured independently (see chapter 4.1.4.2. LED functions).

### Advanced parameters

For the dimming function, the panel can be programmed with advanced parameters which are hidden in the standard view for greater clarity. If necessary, these advanced parameters can be activated and thus be made visible.

The advanced parameters can be used to determine whether the panel is to cover the full adjusting range of the actuator with one dimming telegram continuously ("Increase brightness by 100 %", "Reduce brightness by 100 %") or whether the dimming range is to be divided into several small levels (50 %, 25 %, 12.5 %, 6 %, 3 %, 1.5 %).

In the continuous dimming mode (100%), the push button sensor transmits a telegram only at the beginning of the long press to start the dimming process and generally a stop telegram after the end of the press. For dimming in small levels it may be useful if the push button sensor repeats the dimming telegram in case of a sustained press for a pre-settable time (parameter "Telegram repetition"). The stop telegram after the end of the press is then not needed.

When the parameters are hidden ("Advanced parameters = deactivated"), the dimming range is set to 100 %, the stop telegram is activated and the telegram repetition is deactivated.

- i In order to function properly when operating with a button (TOGGLE) the dimming actuator must transmit its status back to the switching object of the corresponding button. This is also relevant for the correct status indication via the status LED. For operation with two buttons the switching and dimming objects of the related buttons must be assigned with the same group addresses.

### 4.1.4.1.3 Button function "Venetian blind"

For each button with the function set to "Venetian blind", the ETS indicates the two 1-bit objects "Short-time operation" and "Long-time operation".

The status LEDs can be configured independently (see chapter 4.1.4.2. LED functions).

#### Operation concept for the Venetian blind function

For the control of Venetian blind, roller shutter, awning or similar drives, the push button sensor supports four operation concepts in which the telegrams are transmitted in different time sequences. The push button can therefore be used to operate a wide variety of drive configurations.

The different operation concepts are described in detail in the following chapter.

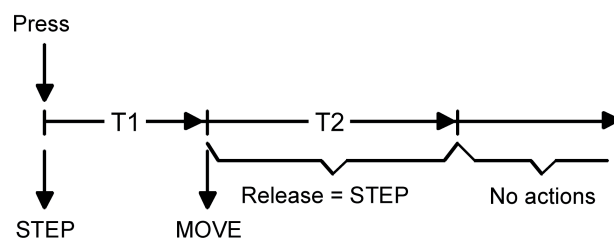


Figure 7: Operation concept "short – long – short"

#### Operation concept "short – long – short":

In the operation concept "short – long – short", the push button sensor shows the following behaviour:

- Immediately on pressing the button, the push button sensor transmits a short time telegram. Pressing the button stops a running drive and starts time T1 ("time between short time and long time command"). No other telegram will be transmitted, if the key is released within T1. This short time serves the purpose of stopping a continuous movement. The "time between short time and long time command" in the push button sensor should be selected shorter than the short time operation of the actuator to prevent a jerky movement of the blind.
- If the button is kept depressed longer than T1, the push button sensor transmits a long time telegram after the end of T1 for starting up the drive and time T2 ("slat adjusting time") is started.
- If the button is released within the slat adjusting time, the push button sensor sends another short time telegram. This function is used for adjusting the slats of a blind. The function permits stopping the slats in any position during their rotation. The "slat adjusting time" should be chosen as required by the drive for a complete rotation of the slats. If the "slat adjusting time" is selected longer than the complete running time of the drive, a push button function is possible as well. This means that the drive is active only when the button is kept depressed.
- If the button is kept depressed longer than T2, the push button sensor transmits no further telegram. The drive remains on until the end position is reached.

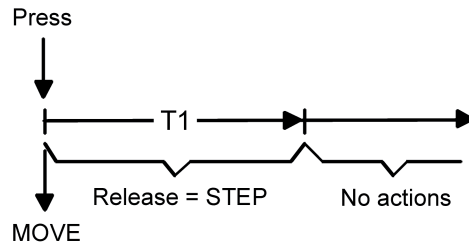


Figure 8: Operation concept "long – short"

**Operation concept "long – short":**

If the operation concept "long – short" is selected, the push button sensor shows the following behaviour:

- Immediately on pressing the button, the push button sensor transmits a long time telegram. The drive begins to move and time T1 ("slat adjusting time") is started.
- If the button is released within the slat adjusting time, the push button sensor transmits a short time telegram. This function is used for adjusting the slats of a blind. The function permits stopping the slats in any position during their rotation. The "slat adjusting time" should be chosen as required by the drive for a complete rotation of the slats. If the "slat adjusting time" is selected longer than the complete running time of the drive, a push button function is possible as well. This means that the drive is active only when the button is kept depressed.
- If the button is kept depressed longer than T1, the push button sensor transmits no further telegram. The drive remains on until the end position is reached.

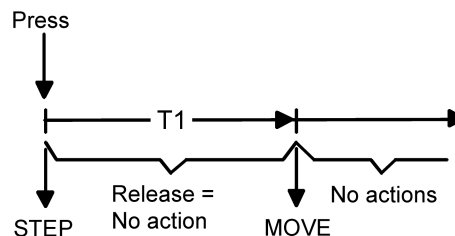


Figure 9: Operation concept "short – long"

**Operation concept "short – long":**

In the operation concept "short – long", the push button sensor shows the following behaviour:

- Immediately on pressing the button, the push button sensor transmits a short time telegram. Pressing the button stops a running drive and starts time T1 ("time between short time and long time command"). No other telegram will be transmitted, if the key is released within T1. This short time serves the purpose of stopping a continuous movement. The "time between short time and long time command" in the push button sensor should be selected shorter than the short time operation of the actuator to prevent a jerky movement of the blind.
- If the button is kept depressed longer than T1, the push button sensor transmits a long time telegram after the end of T1 for starting the drive.
- No further telegram is transmitted when the button is released. The drive remains on until the end position is reached.

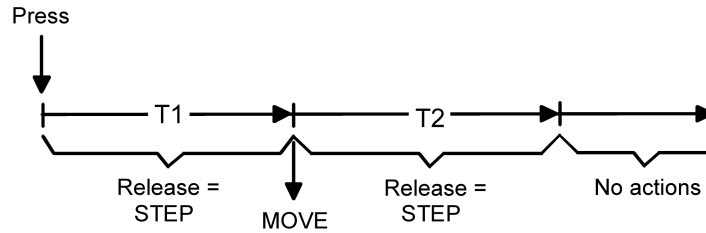


Figure 10: Operation concept "long – short or short"

### Operation concept "long – short or short":

In the operation concept "long – short or short", the push button sensor shows the following behaviour:

- Immediately on pressing the button, the push button sensor starts time T1 ("time between short time and long time command") and waits. If the button is released again before T1 has elapsed, the push button sensor transmits a short time telegram. This telegram can be used to stop a running drive. A stationary drive rotates the slats by one level.
- If the button is kept depressed after T1 has elapsed, the push button sensor transmits a long time telegram and starts time T2 ("slat adjusting time").
- If the button is released within T2, the push button sensor sends another short time telegram. This function is used for adjusting the slats of a blind. The function permits stopping the slats in any position during their rotation.  
The "slat adjusting time" should be chosen as required by the drive for a complete rotation of the slats. If the "slat adjusting time" is selected longer than the complete running time of the drive, a push button function is possible as well. This means that the drive is active only when the button is kept depressed.
- If the button is kept depressed longer than T2, the push button sensor transmits no further telegram. The drive remains on until the end position is reached.



#### 4.1.4.1.4 Button function "Scene extension"

For each button with the function set to "scene extension unit" the ETS indicates the "Function" parameter which distinguishes between the following settings...

- "Scene extension without storage function",
- "Scene extension with storage function",
- "Recall of internal scene without storage function",
- "Recall of internal scene extension with storage function".

In the scene extension function, the push button sensor transmits a preset scene number (1...64) via a separate communication object to the bus after a button-press. This feature permits recalling scenes stored in other devices and also storing them, if the storage function is used.

The recall of an internal scene does not result in a telegram being transmitted to the bus. For this reason, the corresponding communication object is missing. This function can rather be used to recall – and with the storage function also to store – the up to 8 scenes stored internally in the device.

In the setting "... without storage function", a button-press triggers the simple recall of a scene. A long button-press has no further or additional effect.

In the setting "... with storage function", the push button sensor monitors the length of the actuation. A button-press of less than two seconds results in a simple recall of the scene as mentioned above.

After a button-press of more than five seconds, the pushbutton sensor generates a storage instruction. In the scene extension function, a storage telegram is in this case transmitted to the bus. If configured for the recall of an internal scene, the sensor will store the internal scene. An operation lasting between two and five seconds will be discarded as invalid.

The parameter "Scene number" specifies which of the maximum of 8 internal or 64 external scenes is to be used after a button-press.

The status LEDs can be configured independently (see chapter 4.1.4.2. LED functions).

## 4.1.4.1.5 Button function "value transmitter"

For each button with the function set to "1-byte value transmitter" or "2-byte value transmitter" the ETS indicates a corresponding object. On the press of a button, the configured value or the value last stored internally by a value change (see below) will be transmitted to the bus.

The status LEDs can be configured independently (see chapter 4.1.4.2. LED functions).

### Value ranges

The "Function" parameter determines the value range used by the push button.

As a 1-byte value encoder, the pushbutton sensor can optionally transmit integers from 0 ... 255 or relative values within a range of 0 ... 100 % (e.g. as dimming value transmitter).

As a 2-byte value encoder, the pushbutton sensor can optionally transmit integers from 0 ... 65535, temperature values within a range of 0 ... 40 °C or brightness values from 0 ... 1500 lux.

For each of these ranges, the value that can be transmitted to the bus for each actuation of a button is configurable.

### Adjustment by means of long button-press

If the value adjustment feature has been enabled in the ETS, the button must be kept depressed for more than 5 seconds in order to vary the current value of the value transmitter. The value adjustment function continues to be active until the button is released again. In a value adjustment, the panel distinguishes between the following options...

- The "Starting value in case of value adjustment" parameter defines the original starting value for the adjustment. Adjustment can begin from the value configured in the ETS, from the final value of the last adjustment cycle or from the current value of the communication object, with the last option not being available for the temperature and brightness value transmitter.
- The parameter "Direction of value adjustment" defines whether the values will always be increased ("upwards"), always reduced ("downwards") or alternately increased and reduced ("toggling").
- For the value transmitters 0 ... 255, 0 ... 100 % and 0 ... 65535, the "level size" by which the current value is to be changed during the value adjustment can be specified. In case of the temperature and the brightness value transmitter, the level size specifications (1 °C and 50 lux) are fixed.
- The parameter "Time between two telegrams" can be used in connection with the step size to define the time required to cycle through the full respective value range. This value defines the time span between two value transmissions.
- If, during the value adjustment, the push button sensor detects that the preset level size would result in the limits being exceeded with the next telegram, it adapts the level size once in such a way that the respective limit value is transmitted together with last telegram. Depending on the setting of the parameter "Value adjustment with overflow", the push button sensor stops the adjustment at this instance or inserts a pause consisting of two levels and then continues the adjustment beginning with the other limit value.

Type	Function	Lower numerical limit	Upper numerical limit
1-byte value transmitter	0...255	0	255
1-byte value transmitter	0...100 %	0 % (value = 0)	100 % (value = 255)
2-byte value transmitter	0...65535	0	65535
2-byte value transmitter	Temperature value	0 °C	40 °C
2-byte value transmitter	Brightness value	0 lux	1.500 lux

Table 1: Value range limits for the different value transmitters

- i** During a value adjustment, the newly adjusted values are only in the volatile RAM memory of the push button sensor. Therefore, the stored values are replaced by the preset values programmed in the ETS when a reset of the push button sensor occurs (bus voltage failure or ETS programming).
- i** With the 1-byte value encoder in the "Value transmitter 0...100 %" function, the level size of the adjustment will also be indicated in "%". If the starting value of the communication object is used, it may happen in this case during value adjustment that the value last received via the object must be rounded and adapted before a new value can be calculated on the basis of the level size and transmitted. Due to the computation procedure used, the new calculation of the value may be slightly inaccurate.

### Value adjustment examples

#### Configuration example:

- Value transmitter 1-byte (all other value transmitters identical)
- Function = value transmitter 0...255
- Value configured in the ETS (0...255) = 227
- Level size (1...10) = 5
- Start on value adjustment = same as configured value
- Direction of value adjustment = toggling (alternating)
- Time between two telegrams = 0.5 s

#### Example 1: Value adjustment with overflow? = No

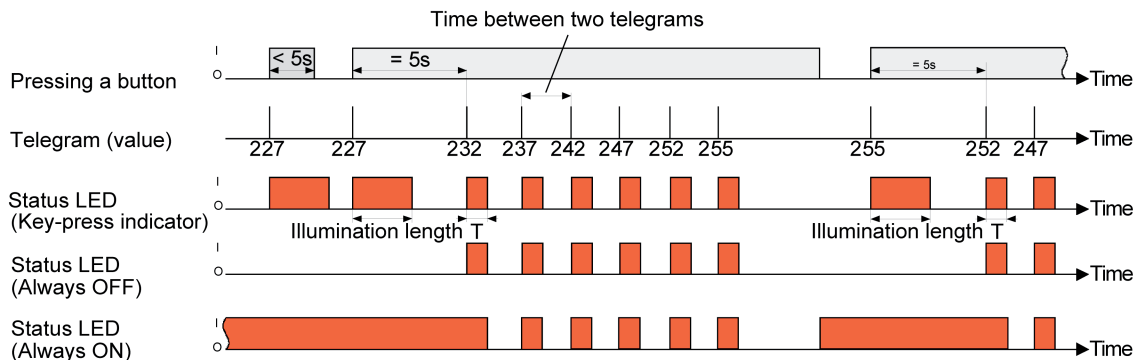


Figure 11: Example of value adjustment without value range overflow

#### Example 2: Value adjustment with overflow? = Yes

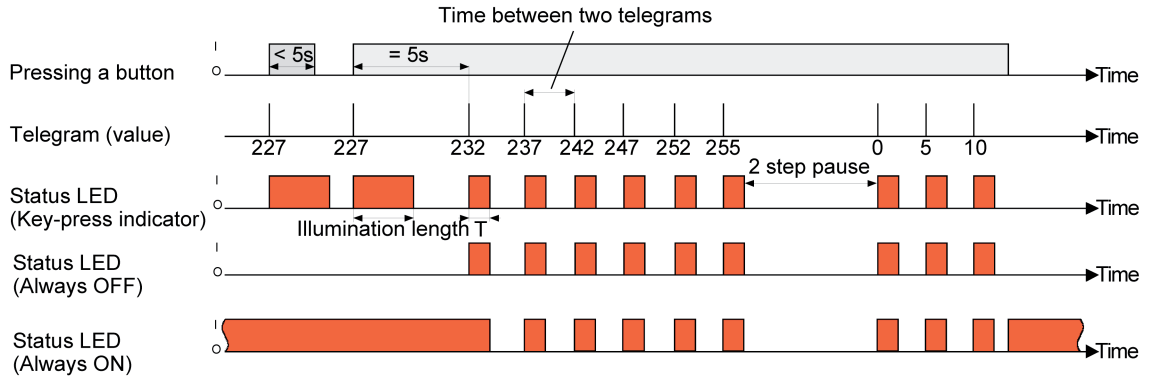


Figure 12: Example of value adjustment with value range overflow

## 4.1.4.1.6 2-channel operation function

In some situations it is desirable to control two different functions with a single button-press and to transmit different telegrams, i.e. to operate two function channels at a time. This is possible with the "2-channel operation" function.

For both channels, the parameters "Function channel 1" and "Function channel 2" can be used to determine the communication object types to be used. The following types are available for selection...

- Switching (1 bit)
- Value transmitter 0 ... 255 (1-byte)
- Value transmitter 0 ... 100 % (1-byte)
- Temperature value transmitter (2 bytes)

The object value the push button sensor is to transmit on a button-press can be selected depending on the selected object type. The "Switching (1 bit)" type permits selecting whether an ON or an OFF telegram is to be transmitted or whether the object value is to be switched over (TOGGLE) and transmitted on the press of a button.

The configuration as "Value transmitter 0 ... 255 (1 byte)" or as "Value transmitter 0 ... 100 % (1 byte)" permits entering the object value freely within a range from 0 to 255 or from 0% to 100%. The "Temperature value transmitter (2 bytes)" permits selecting a temperature value between 0°C and 40°C.

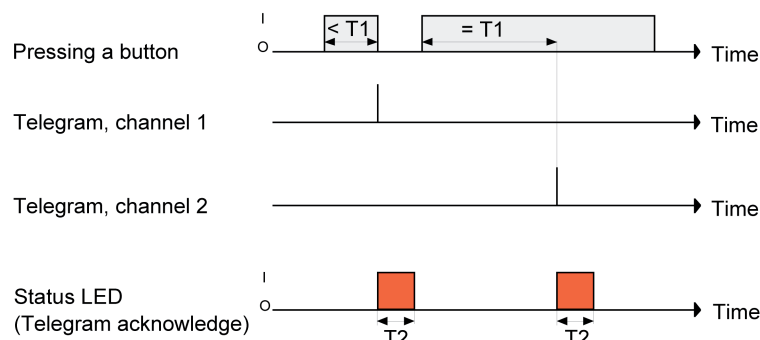
In this case, the adjustment of the object value on a long button-press is not possible as the determination of the actuation length is needed for the adjustable operation concepts.

Unlike in the other button functions, the application software assigns the "Telegram acknowledge" function instead of the "Button-press display" function to the status LED (see chapter 4.1.4.2. LED functions). In this mode, the status LED lights up for approx. 250 ms with each telegram transmitted. Alternatively, the status LEDs can be configured independently (figure 13).

### Operation concept channel 1 or channel 2

In this operation concept, exactly one telegram will be transmitted on each press of a button.

- On a brief press the push button sensor transmits the telegram for channel 1.
- On a long press the push button sensor transmits the telegram for channel 2.



$T1$  = Time between channel 1 und channel 2  
 $T2$  = Illumination length for telegram acknowledge (approx. 250 ms)

Figure 13: Example of operation concept "Channel 1 or Channel 2"

The time required for distinguishing between a short and a long operation is defined by the parameter "Time between channel 1 and channel 2". If the button is pressed for less than the

configured time, only the telegram to channel 1 is transmitted. If the length of the button-press exceeds the time between channel 1 and channel 2, only the telegram to channel 2 will be transmitted. This concept provides the transmission of only one channel. To indicate that a telegram has been transmitted, the status LED lights up for approx. 250 ms in the "Telegram acknowledge" setting.

### Operation concept channel 1 and channel 2

With this operation concept, one or alternatively two telegrams can be transmitted on each button-press.

- On a brief press the push button sensor transmits the telegram for channel 1.
- A long press causes the push button sensor to transmit first the telegram for channel 1 and then the telegram for channel 2.

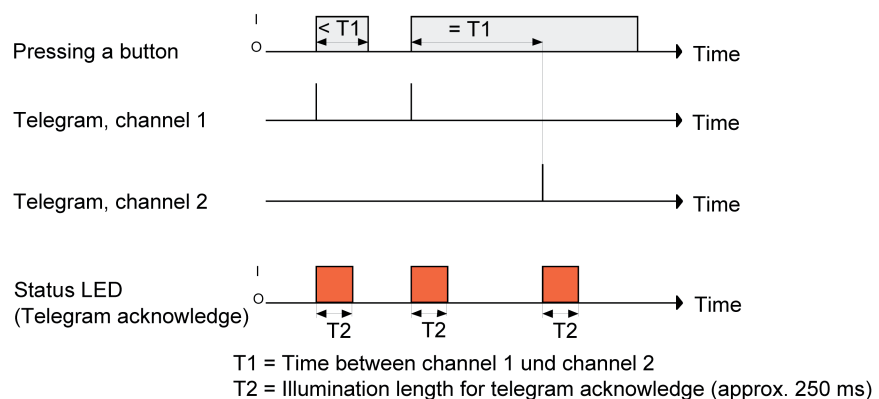


Figure 14: Example for operation concept "Channel 1 and channel 2"

The time required for distinguishing between a short and a long operation is defined by the parameter "Time between channel 1 and channel 2". In this operation concept, a button-press sends this telegram is immediately to channel 1. If the button is held depressed for the configured time, the telegram for the second channel is transmitted as well. If the button is released before the time has elapsed, no further telegram will be transmitted. This operation concept, too, offers the configurable possibility of having the transmission of a telegram signalled by the status LED (setting "Telegram acknowledge").

## 4.1.4.2 LED functions

Each sensor area of the device has a status LED. Depending on the configuration of the buttons, the possible LED functions available differ slightly.

The configuration within the individual push-button functions makes it possible to decide whether the LED should be assigned to the button or functions, which are independent of the button, should be displayed.

If the status LED is connected to a button, the parameter settings "Button-press display", "Telegram acknowledgment" and "Status indication (switching object)" are available. This is described in detail in the chapter "Pushbutton-dependent functions" (see chapter 4.1.4.2.1. Button-dependent LED functions).


The status LED can visualize statuses completely separate from the push-buttons via three separate communication objects, one each for red, green and blue. Various LED functions are available for this purpose, which are explained in the chapter "Pushbutton independent functions" (see chapter 4.1.4.2.2. Button-independent LED functions). Depending on the selected LED function, the object type of the communication object changes to the respective status LED.

### 4.1.4.2.1 Button-dependent LED functions

Each sensor area of the device has a status LED. Depending on the configuration of the buttons, the possible LED functions available differ.

One of the following push-button functions can be assigned to each of the push-buttons of the device:

- Switching/pushing
- Dimming
- Venetian blind
- Scene extension
- 1-byte value transmitter
- 2-byte value transmitter
- 2-channel operation
- No function

 The pushbutton-independent functions of the status LED are summarized under " button-independent functions " (see chapter 4.1.4.2.2. Button-independent LED functions).

#### Switching/pushing and dimming

The status LED assigned to a button can indicate the switching object status as well as the actuation of the button, depending on its configuration. Afterwards, the colour that the LED should light up in, must be selected.

#### Blind and scene extension

With these functions the LED can only be used as a button press display. Since these functions have no status object, this cannot be visualised by the status LED either.

#### 1 value transmitter and 2-byte value transmitter

The associated status LED of the button indicates the value adjustment by one step and the transmission of the telegram with a flash of approx. 250 ms in the colour described below.

Colour or the status LED in case of value adjustment:

- green means that the transmitted value is equal to the starting value
- blue means that the transmitted value is less than the starting value
- red means that the transmitted value is greater than the starting value

## 2-channel operation

With the 2-channel operation the status LED can be used to indicate that a telegram has been transmitted (Telegram acknowledge) . In this case, the status LED of the push-button flashes for approx 250ms, if it was configured to telegram acknowledge.

## No function

If "No function" is selected, the status LED can be used to indicate the pressing of a button by flashing red three times.

- i** If a button-dependent function is assigned to the status LED and an LED colour is selected, the two other colours of the LED are no longer available for further functions.
- i** Irrespective of the push-button function, the status LED can always be switched on (orientation light), or if the glow of the LED is regarded as distracting, it can also be set so that the LED always remains off.



## 4.1.4.2.2 Button-independent LED functions

All of the 24 status LEDs, irrespective of each configured push-button function, can be assigned to functions for the visualization of statuses with the aid of separate communication objects.

The button-independent statuses visualizable by the status LED are:

- Activation via separate LED object
- Operating mode display (KNX controller)
- Controller status general (General/... activate).
- Controller status KNX (General/... activate).
- Comparator without sign
- Comparator with sign
- Limit value display
- Fault message display

### Activation via separate LED object

Each colour of a status LED can indicate the state of a separate LED communication object. In this case, not only is an object generated for the status LED, but a 1-bit object for each colour of the 3-colour LED, too. Here, the LED can be switched on or off statically per colour via the object value received, or flash depending on the object value. An inverted display of the object value is also configurable.

The following parameter pairs can be configured:

- 1=LED ON / 0=LED OFF
- 1=LED OFF / 0=LED ON
- 1=LED flashes / 0=LED OFF
- 1=LED OFF / 0=LED flashes

- i** The colour in which the LED is currently lit depends on which of the three objects of the LED concerned last receives a value. If two or all of the three objects are associated with a group address, no receiving sequence is recognisable. In this case, it is stipulated that the sequence: green -> blue -> red then applies.

### Operating mode display (KNX controller)

An external KNX room temperature controller has a 1-byte communication object whose value returns the current operating mode.

These operating modes are:

- Automatic mode
- Comfort mode
- Standby mode
- Night mode
- Frost / heat protection

Additionally, the combination

- Comfort/Standby/Night mode can be configured

An operating mode for which the LED should light up must be selected here.

The status LED always lights up in configured and active operating mode. The colour is selectable.

When selecting the combination "Comfort/Standby/Night mode", a colour is selectable for each operating mode. In this configuration, the LED indicates the respective operating mode by its colour. Wrong configurations are possible in terms of red/red/red, for example.

### Controller status general

An external KNX room temperature controller has two communication objects whose values

return the current controller status (advanced operating mode).

These statuses are:

- Comfort mode
- Standby mode
- Night mode
- Frost / heat protection
- Controller disabled
- Heating / cooling (Heating = 1 / Cooling = 0)
- Controller inactive (deadband operation)
- Frost alarm
- Normal/Forced operating mode (Forced = 1 / Normal = 0)
- Comfort mode extension
- Open window
- Additional level active
- Dew point alarm
- Comfort/Standby/Night mode can be configured

The status for which the LED should light up must be selected here.

When selecting the combination "Comfort/Standby/Night mode" or the operating mode "Heating/Cooling", a colour is selectable for each operating mode. In this configuration, the LED indicates the respective operating mode by its colour. Wrong configurations are possible in terms of red/red/red, for example.

- i** In order for this indication to function properly, the "Controller status indication" parameter must be set to "Controller status general" in the "General" parameter group.

#### Controller status KNX compliant

An external KNX room temperature controller has a 2-byte communication object whose value returns the current controller status in accordance with the KNX specification. These statuses are:

- Heating / cooling (Heating = 1 / Cooling = 0)
- Dew point alarm
- Controller fault
- Frost protection temperature undershot
- Heat protection temperature exceeded)

The status for which the LED should light up must be selected here.

When selecting the operating mode "Heating/Cooling", a colour is selectable for each operating mode. In this configuration, the LED indicates the respective operating mode by its colour. Wrong configurations are possible in terms of red/red, for example.

- i** In order for this indication to function properly, the "Controller status indication" parameter must be set to "Controller status compliant" in the "General" parameter group.

#### Comparator without sign

The integrated comparators compare the value of the object with an internal configurable value. They distinguish the two different number formats 1-byte und 2-byte.

In the 1-byte parameter setting, unsigned values ranging from 0 to 255 are used for comparison. The configured value is compared with the object value last received. Depending on the parameter setting, the status LED indicates by lighting up whether the received value is greater, less than or equal to the configured value.

In the 2-byte parameter setting, unsigned values for 2-byte values (e. g. lux, temperature...) ranging from 0 to 65535 are used for comparison. The configured value is compared with the object value last received. Depending on the parameter setting, the status LED indicates by switching on whether the received value is greater, less than or equal to the configured value.

The possible comparison operations are:

- Reference value greater than received value.

- Reference value less than received value.
  - Reference value equal to received value.
- The status LED lights up only if the comparison is "true".

### Comparator with sign

The integrated comparators compare the value of the object with an internal configurable value. They distinguish the two different number formats 1-byte und 2-byte.

In the 1-byte parameter setting, signed values ranging from –128 to 127 are used for comparison. The configured value is compared with the object value last received. Depending on the parameter setting, the status LED indicates by lighting up whether the received value is greater, less than or equal to the configured value.

In the 2-byte parameter setting, signed values for 2-byte values (e. g. lux, temperature...) ranging from –32768 to 32767 are used for comparison. The configured value is compared with the object value last received. Depending on the parameter setting, the status LED indicates by switching on whether the received value is greater, less than or equal to the configured value.

The possible comparison operations are:

- Reference value greater than received value.
- Reference value less than received value.
- Reference value equal to received value.

The status LED lights up only if the comparison is "true".

### Limit value display

In this display a function resembling a traffic light can light up the LED in a specific colour. In this case, the LED has a mutual communication object for all three colours. Three ranges can be defined within the corresponding value range with the aid of two limiting values.

The LED colour and the behaviour can be configured for each range separately. It can be set to "off", "red", "green", "blue", "red flashing", "green flashing" and blue flashing".

### Fault message display

The device has a possible fault function for each RGB LED to be able to display fault messages. The fault message function has a communication object "Fault message input LED x" and an object "Fault message acknowledgement LED x". If the fault message is activated, the default function of the adjoining button is inactive and the button acts as an acknowledgement button. The button does not act in the originally assigned function until the fault message has been acknowledged (internally via the button or externally via the bus). After reset, neither a fault nor acknowledgement is displayed.

Procedure for a fault message:

1. Fault:

- LED flashes red (for fault message coming)
- Piezo signal device reproduces the configured alarm signal.

2. Acknowledgement via button or object (transmitting & receiving):

- LED lights up red continuously
- Any continuous signal of the Piezo-signal device that may have been set becomes silent.

### 3. Fault going:

- unacknowledged, LED lights up blue continuously.
- acknowledged, LED lights up green continuously.

If a fault message reoccurs, it starts again at 1.

Acoustic signalling can be set in addition to visual signalling. Under "General/..." an appropriate sound pattern must be selected for this purpose, which is defined under "Piezo". Under "General/...". the playback duration of the sound pattern is configurable. If a 0 is set as a parameter value, the signalling is continuous, it ends with the acknowledgement or when the fault ends.

The selection of the sound pattern applies to all fault messages.

The duration of the sound output is adjustable in a range from 1..255 with the basis of 0.5s. 0 means a continuous sound output. If the sound output should be active as a result of a fault message, the sound pattern will not start playing again with a further fault coming. The duration of the sound output will, however, be restarted with each fault coming.

## 4.1.4.3 Disabling function

### Configuring the disabling function

The operating area of the panel can be completely or partially disabled column by column by means of the 1-bit communication objects "B.Disable buttons" (all buttons), "B.Disable button column 1" (left column), "B.Disable button column 2" (middle column) and "B.Disable button column 3" (right column). During a disable, the buttons can also temporarily execute other functions.

An active disable applies only to the functions of the buttons. The functions of the status LED and scene function are not affected by the disabling function.

The disabling function and the pertaining parameters and communication objects are enabled if the parameter "Disabling function ?" in the parameter node "Disable" is set to "Yes".

You can parameterize the polarity of the disabling object. In case of polarity inversion (disabled = 0 / enabled = 1), the disabling function is not activated immediately after a device reset (object value = "0"). There must first be an object update "0" until the disabling function will be activated.

- i** All status LEDs track their status irrespective of the disabling function. They also update their status in the disabled state of the buttons.
- i** Telegram updates from "0" to "0" or from "1" to "1" on the "button disabling" object remain without effect.

### Configuring the reaction at the beginning and end of a disable

Operating buttons of the device can be disabled. A disable can influence all buttons of the device, only individual button columns or individual buttons. When the disabling is active, the buttons affected cannot show any reaction at all to a button-press, or alternatively can perform a defined behaviour (behaviour like one of the other configured buttons or like one of the two "virtual" disabling functions). The scene functionality is not affected by a disabling function because it is a stored function. Likewise, all status LEDs track their status irrespective of the disabling function. They also update their status in disabled state of the buttons.

Instead of executing another push-button function during a disabling, a disabling function specially projected for this purpose can be executed, too. The disabling function can be projected separately and is only recalled by the corresponding buttons during disabling. All of the push-button functions described previously can be parameterised as a disabling function. If the entire MBT is disabled, individual columns or buttons do not show any reaction when pressed. A status LED projected on an actuation indicator will not light up when a button is pressed in the disabled state. The disabling object can also be treated inversely, i. e. a received "zero" initiates the previously described disabling procedure and a "one" cancels it.

The parameters "Telegram when disabling" and "Telegram when enabling" define how the MBT should behave at the beginning and at the end of the disabling function. Hence, it is possible to configure that the device should trigger a telegram at the beginning of the disabling, how it would otherwise be triggered when pressing a button or when releasing a target button. This is based on the action configured for the respective target button. If the target button is configured in such a way that it has no function or does not transmit a telegram on pressing or releasing of the button, then there is also no reaction to disabling or to enabling. The following table shows the possible reactions of the individual push-button functions when "Pressing" and "Releasing", which can be expected when disabling or enabling.

- i** After about 30 seconds an internal calibration is performed. This cancels the detection of a longer pressing.

Function of >>target button<<	Reaction "as >>target button<< on pressing"	Reaction "as >>target button<< on releasing"
Switching / toggling	Switching telegram	Switching telegram
Dimming	Switching telegram	No telegram
Venetian blind	Move telegram	No telegram
Scene extension	Scene recall telegram	No telegram
1-byte value transmitter	Value telegram	No telegram
2-byte value transmitter	Value telegram	No telegram
Temperature value transmitter	Temperature value telegram	No telegram
Brightness value transmitter	Brightness value telegram	No telegram
2-channel operation Channel 1: 1-bit object type	Switching telegram	No telegram
2-channel operation Channel 1: 1-byte object type	Value telegram	No telegram
2-channel operation Channel 1: 2-byte object type	Temperature value telegram	No telegram
No function	No telegram	No telegram

Table 2: Telegram reactions of the panel with respect to the target push-button function

### Procedure for a configuration (example)

The disabling function must have been enabled in advance.

- Set parameter "Telegram when disabling" and "Telegram when enabling" to "no reaction".  
 The individual buttons, a column or all buttons of the MBT show no reaction at the beginning or at the end of the disabling function. The sensor only adopts the state as provided for by the "Behaviour during active disabling".
- Set parameter "Telegram when disabling" and "Telegram when enabling" to "Internal scene recall scene 1 ...8".  
 The push-button sensor recalls one of the up to 8 internal scenes. Scene storage is not possible.
- Set parameter "Telegram when disabling" and "Telegram when enabling" to "Reaction as button >> X << / >> Y << when pressed / released".  
 Individual buttons, a column or all buttons of the MBT execute the function assigned to any "target button" in non-disabled state. Target buttons are operating buttons of the panel which may be configured for rocker or for button operation. The target buttons are configured separately for the beginning (X) or for the end (Y) of disabling.  
 The action configured for the respective target button is executed. If the target button is configured in such a way that it has no function or does not transmit a telegram on pressing or releasing of the button, then there is also no reaction to disabling or to re-enabling. The telegrams are transmitted to the bus via the required communication object of the target button.
- Set parameter "Telegram when disabling" and "Telegram when enabling" to "Reaction as disabling function 1 ... 6 when pressed / released".

The button, column or all buttons execute the function, which was configured for the disabling function. The disabling functions are button functions with independent communication objects and independent parameters. Except for the status LED, the setting possibilities available for disabling function 1 to 6 are the same as those for the buttons. The respective configuration of the predefined disabling function will be executed. If "no function" or "no telegram" is configuration in the disabling function on pressing or releasing of a button, then there is also no reaction to disabling or to re-enabling. For this setting, Table 2 shows all possible telegram reactions of the push-button sensor depending on the project design of the disabling function. The telegrams are transmitted to the bus via the required communication object of the disabling function.

## 4.1.4.4 Light scene function

### Scene control

The panel can be used in two different ways as part of a scene control system...

- Each button can work as a scene extension. This feature makes it possible to recall or to store scenes which may be stored in other devices (see chapter 4.1.4.1.4. Button function "Scene extension").
- The panel can independently store up to eight scenes with eight actuator groups. These internal scenes can be recalled or stored by the buttons (internal scene recall 1 to 8) and also by the communication object "scene extensions".  
In the following subsections the internal scene function will be dealt with in greater detail.

### Scene definition and scene recall

If the internal scenes are to be used, the parameter "Scene function ?" in the parameter node "Scenes" must be set to "Yes".

The matching data types for the eight scene outputs must then be selected and adapted to the actuator groups used. The types "Switching", "Value (0 ... 255)" or "Value / blind position (0 ... 100 %)" can be selected. As a rule, Venetian blinds are controlled via two scene outputs. One output controls the blind height and the other one adjusts the slat position.

There is a separate parameter node available in the ETS for each scene output. The data types can be selected in this node using the parameters of the same name. The ETS sets the corresponding communication objects and the additional parameters of the scene commands.

The scene parameters can be set in the parameter node of a scene output for each individual scene ("scene 1 ... 8"). The setting options are the same for all 8 scenes.

It is possible to overwrite the values later on for the individual scenes preset by the parameters with the storage function when the system is in operation. If the application program is then loaded again with the ETS, these locally adapted values will normally be overwritten by the parameters. <Due to the fact that it may take considerable efforts to readjust the values for all scenes in the system, the parameter "Overwrite scene values during ETS download ?" offers the possibility of retaining the scene values stored in operation without overwriting them.

These internal scenes can be recalled directly via the buttons (function "recall internal scene") and also by another bus device via the "Scene extension" communication object. This 1 byte communication object supports the evaluation of up to 64 scene numbers. For this reason it must be specified which of the external scene numbers (1 ... 64) is to recall the internal scene (1 ... 8). This specification is made using the parameters "Recall scene 1...8 via extension object with scene number" in the "Scenes" parameter node. If the same scene number is listed for several internal scenes at this point, it is always only the first of these scenes that will be activated (scene with the lowest scene number).

In some situations there may be the requirement that a group of actuators is not controlled by all, but only by certain scenes. A classroom, for instance, may require open blinds for the "Welcome" and "Break" scenes, closed blinds in the "PC presentation" scene and no change in the "Discussion" scene. In this example, the parameter "Permit transmission ?" in the parameter node of a scene output can be set to "No" for the "Discussion" scene. The scene output is then deactivated during the corresponding scene.



The parameter "Transmit delay" permits an individual waiting time for each scene output. This transmit delay can be used in different situations...

- When the actuators participating in a scene transmit status messages automatically or when several scene buttons are used to increase the number of channels within the scenes, the recall of a scene may result for a short time in high bus loading. The transmit delay helps to reduce the bus load at the time of scene recall.
- Sometimes, it is desirable that an action is started only after another action has ended. This can be for instance the illumination which is to shut off only after the blinds/shutters have been raised.

The transmit delay can be set separately for each scene output in the parameter group of a scene. The transmit delay defines the time delay between the individual telegrams during a scene recall. The setting specifies how much time must pass after the first scene telegram before the second is transmitted. After transmission of the second scene telegram, the configured time must again pass before the third is transmitted and so forth. The transmit delay for the scene telegram of the first output is started immediately after recalling the scene. The transmit delay between telegrams can also be deactivated (setting "0"). The telegrams are then transmitted at the shortest possible time interval. In this case, however, the order of the telegrams transmitted can deviate from the numbering of the scene outputs.

- i** When a new scene recall (also with the same scene number) occurs during a current scene recall - even in consideration of the pertaining transmit delays - the scene processing started first will be aborted and the newly received scene number will be processed. A running scene is also aborted when a scene is being stored!
- i** During a scene recall - even if delayed - the operating areas of the panel are operational.

## Storing scenes

For each output of a scene, the user can define a corresponding scene value in the ETS which is then transmitted to the bus during a scene recall. During the ongoing operation of the system it may be necessary to adapt these preset values and to save the adapted values in the panel. This can be ensured by the storage function of the scene control.

The value storage function for the corresponding scene number is enabled with the parameter "Permit storing ?" ("Yes") or disabled ("No"). When the storage function is disabled, the object value of the corresponding output is not sampled during storage.

A scene storage process can be initiated in two different ways...

- by press on a button of a control surface configured as "Scene extension",
- by a storage telegram to the extension object.

During a storage process, the panel reads the current object values of the connected actuators. This is carried out by means of eight read telegrams (ValueRead) addressed to the devices in the scene which return their own value (ValueResponse) as a reaction to the request. The returned values are received by the panel and taken over permanently into the scene memory. Per scene output, the panel waits one second for a response. If no answer is received during this time, the value for this scene output remains unchanged and the panel scans the next output.

In order to enable the panel to read the object value of the actuator addressed when a scene is stored, the read flag of the corresponding actuator object must be set. This should be done only for one actuator out of an actuator group so that the value response is unequivocal.

The stored values overwrite those programmed into the panel with the ETS.

- i The storage process will always be executed completely by the panel and cannot not be aborted before it has ended.
- i Recalling scenes in the course of a storage process is not possible, the buttons of the panel remain nevertheless operational.

## 4.1.4.5 Piezo signal device function

The Piezo signal device can

- be used for the acoustic acknowledgement of the actuation of a button and
- acoustic fault messages can be output.

### Acoustic acknowledgement of the actuation of a button

In order to give direct acknowledgement to the user that a sensor operation has been 'understood' by the device, a button-click can be generated by the Piezo signal device as acoustic acknowledgement for the operation of the buttons. The sound of the button-click is permanently stored in the MBT and cannot be altered by the project designer. The volume is adjustable in levels one, two and three. Level 1 is quiet and level 3 is loud. Pressing, releasing and both events can be signalled. The sound, when releasing a button, is lower than when pressing.

- i** The set parameters apply to all buttons with the exception of the programming button. This normally does not generate any button-click.
- i** You perform this setting in the 'General' parameter window.
- i** If a button is set to "no function", its actuation is indicated by a longer 'error beep

### Acoustic messages for dismantling alarm and fault message

4 different sound patterns can be assigned to the dismantling alarm messages and fault messages.

Each sound pattern includes

- the setting of the pitch (pitch sound  $n$ ) in three different pitches (low, medium, high),
  - the setting of the volume (volume sound  $n$ ) in three different levels (level 1...3),
  - the setting of the sound- /pause time, for generating pulsating sounds (on-time sound  $n$  and off-time sound  $n$ ).
- i** You perform this setting in the 'Piezo' parameter window.

### Priority of the sound output

The sound output of the device can be triggered by different events. Since the MBT only has one Piezo buzzer, it is necessary to assign different priorities to the events. The following table shows the predefined priorities:

Priority	Event
High	Dismantling alarm
Medium	Fault message
Low	Button-click

Table 3: Priorities of the sound output

Sounds of a low priority are interrupted by sounds with a high priority. Sounds of high priority, however, are not influenced by sounds with low priority.

Example: A sound pattern that is playing during an active fault message is interrupted by the triggering of the dismantling alarm with a configured sound output. The sound pattern of the dismantling alarm is output. The reset of the alarm ends its sound output, since the fault message is still active, the sound output of the fault message is resumed from this point.

## 4.1.4.6 Dismantling alarm

For protection against theft the panel has the function of the dismantling alarm. When removing the glass plate, a configurable value is transmitted by the controller. A choice of two object types is available for the dismantling alarm, a switching object or an 8-bit value transmitter object. The object values are freely selectable.

In order to rule out the possibility of a false alarm, a preset time elapses before the message telegram is transmitted. The panel first transmits the dismantling alarm to the bus when the time has elapsed the contact is still open.

- i** In addition to generating the dismantling alarm, the operation of the panel is completely disabled when dismantling the glass plate.

The behaviour of the device after cancelling the dismantling alarm can be defined via the parameter "Automatic reset of the dismantling alarm object" as follows.

Type of dismantling alarms...

- "Dismantling alarm" = "Switching telegram" and "Automatic reset of the dismantling alarm" = "Yes"
- "Dismantling alarm" = "Switching telegram" and "Automatic reset of the dismantling alarm" = "No"
- "Dismantling alarm" = "Value telegram" and "Automatic reset of the dismantling alarm" = "Yes"
- "Dismantling alarm" = "Value telegram" and "Automatic reset of the dismantling alarm" = "No"

"Dismantling alarm" = "Switching telegram" and "Automatic reset of the dismantling alarm" = "Yes"

The dismantling alarm is activated after initially mounting the glass front onto the electronics after reprogramming. Until the control unit has been mounted properly after bus connection, the configured alarm value "1" or "0" is entered into the dismantling alarm object so that when querying the object value it can be detected whether the panel has been mounted properly and is correspondingly ready for operation - in terms of the dismantling alarm.

After the initial proper mounting of the glass front, the object value of the dismantling alarm object is set to the inverted alarm value and the dismantling alarm is switched to active.

In the event of an alarm or dismantling of the glass front, a dismantling telegram (1-bit) is transmitted after the transmit-delay has elapsed. Depending on the configuration "1" or "0".

- i** After a bus voltage failure and subsequent bus voltage return, a previously transmitted dismantling message is not transmitted repeatedly. The object of the dismantling alarm object matches the configured alarm value after bus voltage return.

After mounting the glass front again properly, an inverted dismantling alarm telegram (inverted alarm value) is transmitted and the panel is enabled (the device is functional).

"Dismantling alarm" = "Switching telegram" and "Automatic reset of the dismantling alarm" = "No"

As described above, the panel is still disabled after mounting the glass front again properly until an enabling telegram with an inverted alarm value is received. After receipt of the inverted alarm value (enabling telegram), the panel must be mounted completely. If this is not the case, the enabling telegram is ignored and the entire panel remains disabled.

"Dismantling alarm" = "Value telegram" and "Automatic reset of the dismantling alarm" = "Yes"

The dismantling alarm is activated after the initial mounting of the glass front onto the electronics after reprogramming. The configured alarm value (1 ...255) is entered into the dismantling alarm object until the glass front has been mounted properly after bus connection in

order to detect whether the panel has been installed properly and is correspondingly ready for operation when querying the object value - in terms of the dismantling alarm.

After the initial proper mounting of the glass front, the object value of the dismantling alarm object is loaded with the alarm value "0" and the dismantling alarm is switched to active.

In the event of an alarm or dismantling of the glass front, a dismantling telegram (1-byte) is transmitted after the transmit-delay has elapsed. Depending on the configuration "1" to "255".

- i** After a bus voltage failure and subsequent bus voltage return, a previously transmitted dismantling message is not transmitted repeatedly. The object of the dismantling alarm object matches the configured alarm value after bus voltage return.

After mounting the glass front again properly, a dismantling alarm telegram with the value "0" (inverted alarm value) is transmitted and the panel is enabled (the device is functional).

"Dismantling alarm" = "Value telegram" and  
"Automatic reset of the dismantling alarm" = "No"

As described above, the panel is still disabled after mounting the glass front again properly until an enabling telegram with an inverted alarm value "0" is received. After receipt of the inverted alarm value (enabling telegram), the panel must be mounted completely. If this is not the case, the enabling telegram is ignored and the entire panel remains disabled.

- i** An active dismantling alarm can be signalled acoustically. For this purpose, a sound pattern defined under "Piezo" must be selected. The signalling is continuous, it ends with the reset of the dismantling alarm.

## 4.1.4.7 Software functions

In order to be able to establish logical and time-related relationships between objects or states, the panel provides five logic gates and five timers each.

### 4.1.4.7.1 Logic gate

In order to be able to implement logical dependencies, the device software has five logic gates. Each gate can have from one to a maximum of eight inputs. For each logic gate, the type of logic operation 'AND', 'OR', 'exclusive OR', and 'AND with feedback' can be set. In addition, each input and the output can be operated normally or inverted.

For more complex functions it is possible to combine a number of logic gates. Feedback, i.e. connecting an output with an input of the same gate (poss. also via other logic gates or disabling modules) is not prevented by the configuration software. This does not interfere with the other device functions.

- i** Since feedback can lead to a very large number of telegrams, reasonable switch-on or switch-off delays should be set with the aid of the timers (see chapter 4.1.4.7.2. Timers) in such cases.
- i** With an 'AND with feedback' the value of the output is fed back internally to input 1. The result of this is that the output can only have the value '1' again if input 1 is set to '1' after the value '1' is already present on all of the other inputs. As soon as one of the other inputs is given the value '0', the output and thus input 1 is set to '0' because of the feedback.  
Example: A luminaire that should be switched on manually and switched off again automatically upon reaching a limiting value.  
 For this purpose, the switching object of a push-button is connected to the input 1 and the limiting value object is connected to input 2 of the gate 'AND with feedback'. The pushbutton on input 1 can first be used to switch on the light after the limiting value has set input 2 to '1'. If the light is not switched off again manually, the feedback ensures that input 1 is also reset internally to '0' when the value falls below the limiting value. Without this feedback the light would be switched on again automatically the next time the limiting value is reached if the switching object remains unchanged.

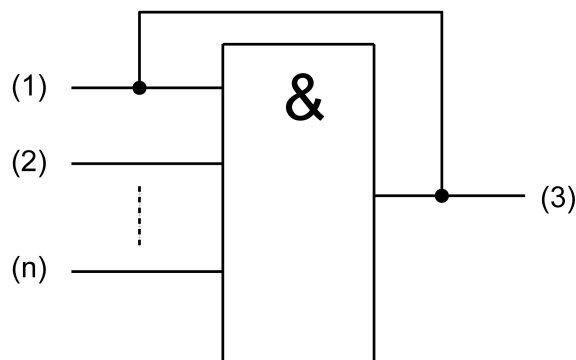


Figure 15: AND with feedback

- (1) Input 1
- (2) Input 2
- (3) Output
- (n) Input 8

## 4.1.4.7.2 Timers

The device has the ability to use up to 5 timers. A timer allows the filtering of incoming 1 bit telegrams of a communication object, the delaying of switching statuses according to the switching edge and then forwarding via an output communication object.

A timer consists of an input object, an output object and an optional disabling object. In the device software, the timer acts like a sluice with time delay and filter function. Depending on the value of the disabling object and the parameters, the value of the input object is either forwarded to the output object or is disabled.

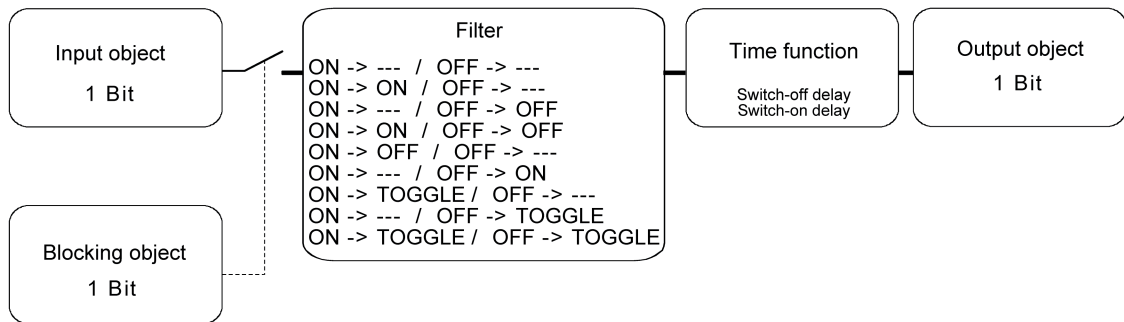


Figure 16: Elements of a timer

The disabling object is a 1 bit communication object, whose behaviour (disable on 0, disable on 1) can be set in the parameter group of a timer. If, during a block, the input value is changed, then the output can automatically send a telegram, as soon as the block is lifted, or it waits until the next input telegram.

## 4.1.5 Parameters

### 4.1.5.1 General parameters

Description	Values	Comment
□ General		
Transmit delay after reset or bus voltage return	Yes	After a device reset, the device can automatically transmit read telegrams to update the object statuses. If other bus devices are still installed in the bus transmitting telegrams immediately after a reset, it may be useful at this point to activate the transmit delay of the automatically generated read request for the controller status in order to reduce the bus load.  When transmit delay is activated (setting: "Yes"), the device computes the delay time from its device ID in the physical address. There is a maximum delay of 30 seconds before the telegrams are transmitted.  This parameter is only available if the controller status display - is enabled and the value request activated.
	No	
Light period of status LED for button-press indicator	1 sec	This parameter defines the switch-on time the status LED is lit up to indicate actuation. The setting concerns all status LEDs whose function is set to "Button-press display".
	2 sec	
	<b>3 sec</b>	
	4 sec	
	5 sec	
Acoustical signal after a button-press	No	A button-click can be generated by the Piezo signal device as acoustic acknowledgement for the operation of the buttons in order to give direct acknowledgement to the user that a sensor operation has been 'understood' by the device. An unchangeable 'clicking' sound is available for this purpose. If a button is set to "no function", its actuation is indicated by a longer 'error beep'
	<b>When pressed</b>	
	When released	
	When pressed and released	
Volume button operation	Level 1	This setting defines the volume of the button operation.
	<b>Level 2</b>	
	Level 3	
Sound for fault message	<b>Off</b>	This setting defines which of the sounds configured under the "Piezo" parameter group is played back in the event of a fault message.
	Sound 1	
	Sound 2	
	Sound 3	
	Sound 4	



Controller status indication	<b>No</b> KNX compliant Controller status general	A KNX controller transmits its status among other things in a 2-byte object (KNO compliant) or in two 1-byte objects (controller status general). If the controller status parameter is enabled, then one or two receiving objects are created for receiving this controller status byte. The information 'Automatic', 'Comfort mode', 'Standby mode', 'Night mode', 'Frost/heat protection mode', 'Controller disabled', 'Heating/cooling', 'Controller inactive', 'Frost alarm', 'Dew point alarm', 'Controller fault', 'Frost protection temperature undershot' or 'Heat protection temperature exceeded' are displayed with the aid of the status LED in this parameter setting.
Value request for controller status display	<b>Yes</b> <b>No</b>	Requests an external controller to transmit its status.  This parameter is only available if the controller status display - is enabled.

## 4.1.5.2 Push-button parameters

Description	Values	Comment
□↵ Buttons -> Button 1		
Function	No function <b>Switching</b> Dimming Venetian blind 1-byte value transmitter 2-byte value transmitter Scene extension 2-channel operation	This parameter defines the basic function of the button. Depending on this setting, the ETS displays different communication objects and parameters for this button.

The following parameters are only valid for the push button function "Switching"...

Command on pressing the button	No reaction ON OFF <b>TOGGLE</b>	This parameter specify the reaction when the button is pressed or released.
Command on releasing the button	<b>No reaction</b> ON OFF TOGGLE	

The following parameters are only valid for the push button function "Dimming"...

Command on pressing the button	No reaction Brighter (ON) Darker (OFF) <b>Brighter / darker (TOGGLE)</b> Brighter (TOGGLE) Darker (TOGGLE)	This parameter defines the reaction when the button is pressed. If the panel is to toggle on a brief press, the corresponding switching objects of other sensors with the same function must be linked with one another. In the "Brighter/darker (TOGGLE)" setting, the dimming objects must be interlinked as well so that the button can send the correct telegram on the next button-press.
Time between switching and dimming (100 ... 50000 x 1 ms)	100 ... <b>400</b> ... 50000	This parameter defines how long the button must be pressed for the panel to transmit a dimming telegram.
Increase brightness by	1.5 % 3 % 6 % 12.5 % 25 % 50 % <b>100 %</b>	This parameter sets the relative dimming level when the brightness is increased. On each button-press, the brightness is changed at maximum by the configured level. Especially with smaller dimming levels it is advisable for the panel to repeat the dimming telegrams automatically (see "telegram repetition").

Reduce brightness by	1.5 % 3 % 6 % 12.5 % 25 % 50 % <b>100 %</b>	This parameter sets the relative dimming level when the brightness is reduced. On each button-press, the brightness is changed at maximum by the configured level. It is recommendable, particularly with a smaller dimming level, when the panel repeats the dimming telegrams automatically (see "telegram repetition").
Send stop telegram ?	<b>Yes</b> No	On "Yes" the panel transmits a telegram for stopping the dimming process when the button is released. When the panel transmits telegrams for dimming in smaller levels, the stop telegram is generally not needed.
Telegram repeat?	Yes <b>No</b>	This parameter can be used to activate telegram repetition for dimming. While the button is pressed, the panel will then transmit the relative dimming telegrams (in the programmed level size) until the button is released.
Time between two telegrams	<b>200 ms</b> 300 ms 400 ms 500 ms 750 ms 1 sec 2 sec	This parameter defines the interval at which the dimming telegrams are automatically repeated in the telegram repetition mode. Visible only if "Telegram repetition = Yes"!
The following parameters are only valid for the push button function "Venetian blind"...		
Command on pressing the button	DOWN UP <b>TOGGLE</b>	This parameter defines the running direction of a drive after a button-press. If the setting is "TOGGLE", the direction is changed after each long time command. If several push buttons are to control the same drive, the long time objects of the push buttons must be interlinked for a correct change of the running direction.
Operation concept	<b>short – long – short</b>  long – short  short – long	For Venetian blind control, four different operation concepts can be selected. For these concepts, the ETS shows further parameters.

long – short or short

Time between short-time and long-time command (1 ... 3000 x 100 ms)	1 ... <b>4</b> ... 3000	This parameter sets the time after which the long-time operation will be executed on pressing the button for UP. This parameter is not visible with "Operation concept = long – short"!
Slat adjusting time (0 ... 3000 x 100 ms)	0 ... <b>5</b> ... 3000	Time during which a transmitted MOVE telegram can be terminated by releasing the key for UP (STEP). This function serves to adjust the slats of a blind. This parameter is not visible with "Operation concept = long – short"!

The following parameters are only valid for the push-button function "value transmitter 1 byte"...

Function	<b>Value transmitter 0 ... 255</b> Value transmitter 0 ... 100 %	A button configured as "Value transmitter 1 byte" permits selecting whether the values to be transmitted are interpreted as integers from 0 to 255 or as a percentage from 0 % to 100 %. The following parameters and their settings depend on this distinction.
Value (0 ... 255)	<b>0...255</b>	This parameter defines the object value when the button is pressed. Visible only if "Function = ... 0...255"!
Value (0 ... 100 %)	<b>0...100</b>	This parameter defines the object value when the button is pressed. Visible only if "Function = ... 0...100 %"!
Value adjustment by long button-press	enabled  <b>disabled</b>	If value adjustment by long button-press is enabled, the ETS shows further parameters. Value adjustment begins, when the button is held down for more than 5 s. In this case, the respective status LED flashes as a sign that a new telegram has been transmitted.
Starting value in case of value adjustment	Same as configured value  Same as value after last adjustment  <b>Same as value from communication object</b>	Value adjustment can begin with different starting values. In the setting "Same as parameterised value", after each long press the panel always starts with the value programmed in the ETS. In the setting "Same as value after last adjustment", after a long press the panel starts with the value transmitted by itself or by another device with this group

		<p>address as the last value.          In the setting "Same as value from communication object", after a long press the panel starts with the value transmitted by itself or by another device with this group address as the last value.          This parameter is only visible if "Value adjustment by long button-press = enabled"!</p>
Direction of value adjustment	<p>Upwards</p> <p>Downwards</p> <p><b>Toggling (alternating)</b></p>	<p>With a long press, the panel can either vary the values always in the same direction or it stores the direction of the last adjustment and reverses it on the next button-press. Visible only if "Value adjustment by long button-press = enabled"!</p>
Level size (1 ... 15)	<p>1...<b>15</b></p>	<p>In a value adjustment, the panel determines the new telegram value from the previous value and the preset level size. If the value falls below the lower limit of the adjustment range (0 or 0 %) or if it exceeds the upper limit (255 or 100%), the sensor adapts the level size of the last level automatically. Visible only if "Value adjustment by long button-press = enabled"!</p>
Time between two telegrams	<p><b>0.5 sec</b></p> <p>1 sec</p> <p>2 sec</p> <p>3 sec</p>	<p>This parameter defines the interval at which the panel transmits new telegrams during a value adjustment. Visible only if "Value adjustment by long button-press = enabled"!</p>
Value adjustment with overflow	<p>Yes</p> <p><b>No</b></p>	<p>If value adjustment is to be effected without overflow (setting "No") and if the panel reaches the lower limit of the adjustment range (0 or 0 %) or the upper limit (255 or 100 %) during value adjustment, the adjustment will be stopped automatically by the sensor. If the value adjustment with overflow is programmed (setting "Yes") and if the panel reaches the lower or the upper limit, it will transmit the value of this range limit and then add a pause the duration of which corresponds to two levels. Thereafter, the panel transmits a telegram with the value of the other range limit and continues the value adjustment in the same direction.</p>

The following parameters are only valid for the push button function "value transmitter 2 byte"...

Function	<b>Temperature value transmitter</b>  Brightness value transmitter  Value transmitter (0 ... 65535)	A button configured as "Value transmitter 2 byte" permits selecting whether the values to be transmitted are to be interpreted as temperature values (0 °C to 40 °C), as brightness values (0 lux to 1500 lux) or as integers (0 to 65535). The following parameters and their settings depend on this selection.
Temperature value (0 ... 40 °C)	0... <b>20</b> ...40	This parameter defines the object value when the button is pressed. Visible only if "Function = Temperature value transmitter"!
Brightness value	0, 50, ... <b>300</b> ... 1450, 1500 lux	This parameter defines the object value when the button is pressed. Visible only if "Function = Brightness value transmitter"!
Value (0 ... 65535)	<b>0</b> ... 65535	This parameter defines the object value when the button is pressed. Visible only with "Function = Value transmitter (0 ... 65535)"!
Value adjustment by long button-press	enabled  <b>disabled</b>	If value adjustment by long button-press is enabled, the ETS shows further parameters. Value adjustment begins, when the button is held down for more than 5 s. In this case, the respective status LED flashes as a sign that a new telegram has been transmitted.
Starting value in case of value adjustment	Same as configured value  Same as value after last adjustment  <b>Same as value from communication object</b>	Value adjustment can begin with different starting values. This parameter is only visible if "Value adjustment by long button-press = enabled"! In the setting "Same as parameterised value", after each long press the panel always starts with the value programmed in the ETS. In the setting "Same as value after last adjustment", after a long press the panel starts with the value transmitted by itself or by another device with this group address as the last value. In the setting "Same as value from communication object", after a long press the panel starts with the value transmitted by itself or by another device with this group address as the last

		value. This setting selectable only if "Functionality = Value transmitter (0...65535)!"
Direction of value adjustment	<p>Upwards</p> <p>Downwards</p> <p><b>Toggling (alternating)</b></p>	With a long press, the panel can either vary the values always in the same direction or it stores the direction of the last adjustment and reverses it on the next button-press. Visible only if "Value adjustment by long button-press = enabled"!
Level size	<b>1 K</b>	For temperature values the level size of the adjustment is permanently set to 1 K. Visible only if "Function = Temperature value transmitter" and "Value adjustment by long button-press = enabled"!
Level size	<b>50 lux</b>	For brightness values, the level size of the adjustment is fixed to 50 lux. Visible only if "Function = Brightness value transmitter" and "Value adjustment by long button-press = enabled"!
Level size	<p>1</p> <p>2</p> <p>5</p> <p>10</p> <p>20</p> <p>50</p> <p>75</p> <p>100</p> <p>200</p> <p>500</p> <p>750</p> <p><b>1000</b></p>	This parameter sets the level size of the value adjustment for the 2-byte value transmitter. Only visible if "Function = Value transmitter (0 ... 65535)" and "Value adjustment by long button-press = enabled"!
Time between two telegrams	<p>0.5 sec</p> <p><b>1 sec</b></p> <p>2 sec</p> <p>3 sec</p>	This parameter defines the interval at which the panel transmits new telegrams during a value adjustment. Visible only if "Value adjustment by long button-press = enabled"!
Value adjustment with overflow	<p>Yes</p> <p><b>No</b></p>	If value adjustment is to be effected without overflow (setting "No") and if the panel reaches the lower limit of the adjustment range (0 °C, 0 or Lux, 0) or the upper limit (+ 40 °C, 1500 Lux, 65535) during value adjustment, the adjustment will be stopped automatically by the sensor. If the value adjustment with overflow is

programmed (setting "Yes") and if the panel reaches the lower or the upper limit, it will transmit the value of this range limit and then add a pause the duration of which corresponds to two levels. Thereafter, the panel transmits a telegram with the value of the other range limit and continues the value adjustment in the same direction.

The following parameters are only valid for the push button function "scene extension"...

Function	<b>Scene extension without storage function</b>	This parameter defines the functionality of the extension. If a button is used as a scene extension, the scenes can either be stored in one or in several other KNX devices (e.g. light scene push button sensor). During a scene recall or in a storage function, the panel transmits a telegram with the respective scene number via the extension object of the rocker. During the recall of an internal scene, a scene stored internally in the panel is recalled or stored again. In this case, the sensor transmits no telegram to the bus via a scene extension object. For this setting, the internal scene function must be enabled.
	Scene extension with storage function	
	Recall of internal scene extension without storage function	
	Recall of internal scene with storage function	
Scene number (1 ... 64)	1...64	In accordance with the KNX standard, objects with data type 18.001 "Scene Control" can recall or store up to 64 scenes by their numbers. The parameter defines the scene number to be transmitted when the button is pressed.
Scene number (1 ... 8)	1...8	This parameter defines the number of the internal scene which is recalled or stored when a button is pressed.

The following parameters are only valid for the push button function "2-channel operation"...

Operation concept	<b>Channel 1 or channel 2</b>	This parameter defines the 2-channel operation concept. If the setting "channel 1 or channel 2" is selected, the panel decides dependent on the button-press duration which of the channels will be used.
	Channel 1 and channel 2	If the setting "channel 1 and channel 2" is selected, the panel transmits only the telegram of channel 1 on a short button-press and both telegrams on a sustained button-press.



Function channel 1 (2)	No function <b>Switching (1 bit)</b> Value transmitter 0 ... 255 (1-byte) Value transmitter 0 ... 100 % (1-byte) Temperature value transmitter (2-bytes)	This parameter defines the channel function and specifies which other parameters and which communication object are to be displayed for channel 1 or channel 2.
Command of button for channel 1 (2)	ON OFF <b>TOGGLE</b>	This parameter defines the object value transmitted to the bus when the button is pressed. Only visible if "Function channel 1 (2) = Switching (1 bit)"!
Value of the button for Channel 1 (2) (0 ... 255)	<b>0...255</b>	This parameter defines the object value transmitted to the bus when the button is pressed. Visible only if "Function channel 1 (2) = value transmitter 0...255 (1 byte)"!
Value of the button for Channel 1 (2) (0 ... 100 %)	<b>0...100</b>	This parameter defines the object value transmitted to the bus when the button is pressed. Visible only if "Function channel 1 (2) = value transmitter 0...100 % (1 byte)"!
Temperature value of the button for channel 1 (2) (0 ... 40 °C)	0 ... <b>20</b> ... 40	This parameter defines the temperature value transmitted to the bus when the button is pressed. Visible only if "Function channel 1 (2) = Temperature value transmitter (2 bytes)"!
Time between channel 1 and channel 2 (1 ... 255 x 100 ms)	0... <b>30</b> ...255	Depending on the selected operation concept, this parameter defines the interval at which the panel transmits the telegram for channel 1 and the telegram for channel 2 when the button is pressed.

☐ Buttons -> Button 2 ... n see Button 1

☐ Buttons -> Button 1...24 -> Function of status LED

Function of status LED	always OFF	Irrespective of the pushbutton function, the status LED is switched off permanently.
------------------------	------------	--

---

Always ON	Irrespective of the pushbutton function, the status LED is switched on permanently. This setting causes the additional parameters "colour of the LED" to be shown.
Button-press display	The status LED indicates a button actuation. The ON time is set on the parameter page "General" in common for all status LEDs that are configured as actuation displays. This setting <u>cannot</u> be configured for the pushbutton function "2-channel operation". This setting causes the additional parameter "colour of the LED" to be shown.
Telegram acknowledgment	The status LED indicates the transmission of a telegram in 2-channel operation. This setting can only be configured for the pushbutton function "2-channel operation". This setting causes the additional parameter "colour of the LED" to be shown.
Status indication (switching object)	In the "Switching" and "Dimming" push-button functions, the status LED signals the status of the "Switching" object. The object value is evaluated as follows: "ON" -> "LED illuminated / OFF" -> LED goes out. This setting causes the additional parameter "colour of the LED" to be shown.
Inverted status display (switching object)	In the "Switching" and "Dimming" push-button functions, the status LED signals the inverted status of the "Switching" object. The object value is evaluated as follows: "ON" -> "LED illuminated / OFF" -> LED goes out. This setting causes the additional parameter "colour of the LED" to be shown.
Activation via separate LED object	The status LED indicates with this configuration the state of 3 of its own, separate 1-bit LED objects. Three objects are generated for the 3 colours, status LED red, status LED green and

---

	<p>status LED blue (B.LED 'n'). The object which received its value last determines the colour that the status LED lights up in.</p> <p>This setting causes the additional parameter "Activation of the status LED via object value" to be shown.</p>
Operating mode display (KNX controller)	<p>The status LED indicates the state of a KNX room temperature controller via a separate 1-byte communication object. This setting causes the additional parameters "Status LED ON with" and "Colour of the LED" or "LED colours" for Comfort/Standby/Night mode to be shown.</p>
Controller status general (General/...activate)	<p>The status LED indicates the state of a KNX room temperature controller via two separate 1-byte communication object (Extension) This setting causes the additional parameters "Status LED ON with" and "Colour of the LED" or "LED colours" for Comfort/Standby/Night mode or for Heating/Cooling to be shown.</p>
Controller status KNX compliant (General/...activate)	<p>The status LED indicates the state of a KNX room temperature controller via a separate 2-byte communication object (Extension). This setting causes the additional parameters "Status LED ON with" and "Colour of the LED" or "LED colours" for Heating/Cooling to be shown.</p>
Comparator without sign (1-byte)	<p>The status LED is activated depending on a comparison. In this configuration there is a separate 1-byte communication object available via which the unsigned reference value (0...255) is received.</p> <p>This setting causes the additional parameters "Status LED ON with", "Type of reference value", "Reference value" and "Colour of the LED" to be shown.</p>
Comparator without sign (2-byte)	<p>The status LED is activated depending on a comparison. In this configuration there is a separate 2-byte communication object available via which the unsigned reference value (0...65535) is received.</p>

---

	<p>This setting causes the additional parameters "Status LED ON with", "Type of reference value", "Reference value" and "Colour of the LED" to be shown.</p>
Comparator with sign (1-byte)	<p>The status LED is activated depending on a comparison. In this configuration there is a separate 1-byte communication object available via which the positive or negative reference value (-128...127) is received. This setting causes the additional parameters "Status LED ON with", "Type of reference value", "Reference value" and "Colour of the LED" to be shown.</p>
Comparator with sign (2-byte)	<p>The status LED is activated depending on a comparison. In this configuration there is a separate 2-byte communication object available via which the positive or negative reference value (-32768...32767) is received.</p> <p>This setting causes the additional parameters "Status LED ON with", "Type of reference value", "Reference value" and "Colour of the LED" to be shown.</p>
Limit value display	<p>This function compares an external object value with two internally configurable limiting values. The three-colour status LED changes the colour. Three cases can be distinguished (on condition that limiting value 1 is greater than limiting value 2):</p> <ul style="list-style-type: none"><li>The object value is less than limiting value 1</li><li>The object value is equal to or greater than limiting value 1</li><li>The object value is equal to or greater than limiting value 2.</li></ul> <p>This setting causes the additional parameters "Type of limiting values" and "LED colours" to be shown.</p>
Fault message display	<p>Two objects are generated, "B.Fault message input" and "B.Fault message acknowledgement. The colours that the three-colour status LEDs light up in are predefined for the different phases of a fault message.</p> <ol style="list-style-type: none"><li>1. Flashing in red for fault coming,</li><li>2. Continuously in red for acknowledged</li></ol>

		<p>fault,          3. going          3.1. Blue unacknowledged          3.2. Green unacknowledged.</p>
	No function indication	The status LED indicates a button actuation by flashing red three times. This setting can only be configured for the pushbutton function "No function".
<p>All functions of the status LED indicator except for "No function", "Always OFF" and "Limit value display"...</p>		
Colour of the status LED	<p><b>red</b>          green          blue</p>	Selection of the colour that the status LED lights up in when the configured condition is met.
<p>If status LED indicator "Operating mode display (KNX controller)" and "Controller status indication status general"...</p>		
Comfort mode	<p><b>red</b>          green          blue</p>	Selection of the colour that the status LED lights up in when the controller works in comfort mode.
Standby mode	<p>red  <b>green</b>          blue</p>	Selection of the colour that the status LED lights up in when the controller works in standby mode.
Night mode	<p>red          green  <b>blue</b></p>	Selection of the colour that the status LED lights up in when the controller works in night mode.
<p>If status LED indicator "Controller status indication status general" and "Controller status indication status KNX compliant"...</p>		
Heating	<p><b>red</b>          green          blue</p>	Selection of the colour that the status LED lights up in when the controller works in "Heating" operating mode.

Cooling	red green <b>blue</b>	Selection of the colour that the status LED lights up in when the controller works in "Cooling" operating mode.
<p>If Function of status LED = "Limit value display" -&gt; LED colours ...</p>		
Limiting value 1 undershot	red <b>green</b> blue	Selection of the colour that the status LED lights up in when the limiting value 1 compared with the object value is undershot.
Limiting value 1 is equal to or exceeded	red green <b>blue</b>	Selection of the colour that the status LED lights up in when the limiting value 1 compared with the object value is equal to or has been exceeded.
Limiting value 2 is equal to or exceeded	<b>red</b> green blue	Selection of the colour that the status LED lights up in when the limiting value 2 compared with the object value is equal to or has been exceeded.
<p>The function of the status LED = "Display via separate LED object"...</p>		
Activation of the status LED via object value	<b>1 = LED ON /</b> <b>0 = LED OFF</b>  1 = LED OFF / 0 = LED ON  1 = LED flashes / 0 = LED OFF  1 = LED OFF / 0 = LED flashes	<p>If the "Function of status LED ..." is set to "Control via separate LED object", then the telegram polarity of the 1-bit object "Status LED" can be specified at this point. The LED can be switched on or off statically. In addition, the received switching telegram can be evaluated in such a way that the LED flashes.</p>
<p>If the function of status LED = "Operating mode display (KNX controller)"...</p>		
Status LED ON with	Automatic mode <b>Comfort mode</b> Standby mode Night mode Frost / heat protection Comfort/Standby/Night mode	<p>The values of a communication object with data type 20.102 "HVAC Mode" are defined as follows: 0 = Automatic 1 = Comfort 2 = Standby 3 = Night</p>

4 = Frost/heat protection

The value "Automatic" is used only by the "forced operating mode switchover" objects.

The status LED is illuminated when the object receives the value configured here.

If the "Comfort/Standby/Night mode" is selected, an LED colour can then be selected for each of the three states.

If function of the status LED = "Controller status indication status general" (General/... activate"...

Status LED ON with

**Comfort mode**  
 Standby mode  
 Night mode  
 Frost/heat protection mode  
 Controller disabled  
 Heating / cooling (Heating = 1 / Cooling = 0)  
 Controller inactive (deadband operation)  
 Frost alarm  
 Normal/Forced operating mode (Forced = 1 / Normal = 0)  
 Comfort mode extension  
 Open window  
 Additional level active  
 Dew point alarm  
 Comfort/Standby/Night mode

The status LED is lights up if the controller is in the configured state here.

If the "Comfort/Standby/Night mode" is selected, an LED colour can then be selected for each of the three states.

If the "Heating / Cooling (Heating = 1 / Cooling = 0)" parameter is selected, an LED colour can then be selected for each of the two states.

If function of the status LED = "Controller status indication KNX compliant" (General/... activate"...

Status LED ON with

**Heating / cooling (Heating = 1 / Cooling = 0)**  
 dew point alarm  
 Controller fault  
 Frost protection  
 temperature undershot  
 Heat protection  
 temperature exceeded

The status LED is lights up if the controller is in the configured state here.

If the "Heating / Cooling (Heating = 1 / Cooling = 0)" parameter is selected, an LED colour can then be selected for each of the two states.

If the function of status LED = "Comparator without sign"...

Status LED ON with

	<p><b>Reference value greater than received value</b></p> <p>Reference value less than received value</p> <p>Reference value equal to received value</p>	<p>The status LED indicates whether the configured reference value is greater or less than or equal to the value of the "B.LED" object n".</p>
Reference value type	<p><b>1 byte</b></p> <p>2 byte</p>	<p>This parameter defines the number range of the comparator.</p>
If reference value type = 1 byte Reference value (0 ... 255)	<p><b>0 ... 255</b></p>	<p>This parameter defines the exact reference value to which the value of the "B.LED n" object is compared.</p>
If reference value type = 2 byte Reference value (0 ... 65535)	<p><b>0 ... 65535</b></p>	<p>This parameter defines the exact reference value to which the value of the "B.LED n" object is compared.</p>
If the function of status LED = "Comparator with sign"...		
Status LED ON with	<p><b>Reference value greater than received value</b></p> <p>Reference value less than received value</p> <p>Reference value equal to received value</p>	<p>The status LED indicates whether the configured reference value is greater or less than or equal to the value of the "B.LED" object n".</p>
Reference value type	<p><b>1 byte</b></p> <p>2 byte</p>	<p>This parameter defines the number range of the comparator.</p>
If reference value type = 1 byte Reference value (-128 ... 127)	<p><b>-128 ... 0 ... 127</b></p>	<p>This parameter defines the exact reference value to which the value of the "B.LED n" object is compared.</p>
If reference value type = 2 bytes Reference value (-128 ... 127)	<p><b>-32768 ... 0 ... 32667</b></p>	<p>This parameter defines the exact reference value to which the value of the "B.LED n" object is compared.</p>



## 4.1.5.3 Parameter of the disabling functions

Description	Values	Comment
<p>☐ Disabling</p> <p>Disabling function?</p>	<p>Yes</p> <p><b>No</b></p>	<p>With this parameter, the disabling function of the panel can be centrally activated.</p> <p>If "Yes", the ETS shows further communication object and parameters.</p>
<p>Polarity of disabling object</p>	<p><b>disable = 1 / enable = 0</b></p> <p>Disable = 0 / enable = 1</p>	<p>This parameter defines the value of the disabling object at which the disabling function is active.</p>
<p>Reaction of the panel at the beginning of the disabling function</p>	<p><b>No reaction</b></p> <p>Reaction as button &gt;&gt;X&lt;&lt; when pressed</p> <p>Reaction as button &gt;&gt;X&lt;&lt; when released</p> <p>Reaction as disabling function 1 when pressed</p> <p>Reaction as disabling function 1 when released</p> <p>Reaction as disabling function 2 when pressed</p> <p>Reaction as disabling function 2 when released</p> <p>...</p> <p>Reaction as disabling function 6 when pressed</p> <p>Reaction as disabling function 6 when released</p> <p>Internal scene recall scene 1</p> <p>Internal scene recall scene 2</p> <p>...</p> <p>Internal scene recall scene 8</p>	<p>Besides disabling the button functions, a pushbutton can also trigger a specific function at the time when the disabling state occurs.</p> <p>This function can... correspond to the function assigned to any of the buttons in the non-disabled state ("Reaction as button &gt;&gt;X&lt;&lt; ..."), be defined on the following parameter pages ("Reaction as disabling function ..."), recall a scene stored internally in the panel ("Internal scene recall ...").</p>
<p>Button &gt;&gt;X&lt;&lt;</p>	<p><b>Button 1</b></p> <p>Button 2</p>	<p>If the panel is to perform the function of a specific button at the beginning of the</p>

	<p>... Button 24</p>	<p>disabling state, this button will be selected here.</p> <p>Visible only if "Reaction of pushbutton sensor at the beginning of the disabling function = Reaction as button &gt;&gt;X&lt;&lt; on pressing / releasing"!</p>
Behaviour during active disabling	<p><b>All buttons without function</b></p> <p>All buttons behave as</p> <p>Individual buttons without function</p> <p>Individual buttons behave as</p> <p>Individual columns without function</p> <p>Individual columns behave as</p>	<p>While disabling is active... all buttons, individual columns or only individually selected buttons can be disabled ("... no function"), all buttons, individual columns or only individually selected buttons can be restricted to a specific function ("... behave as"). In this case, the ETS shows further parameters. (see advanced parameters -&gt; Behaviour during active disabling).</p>
Reaction of pushbutton sensor at the end of disabling	<p><b>No reaction</b></p> <p>Reaction as button &gt;&gt;Y&lt;&lt; when pressed</p> <p>Reaction as button &gt;&gt;Y&lt;&lt; when released</p> <p>Reaction as disabling function 1 when pressed</p> <p>Reaction as disabling function 1 when released</p> <p>Reaction as disabling function 2 when pressed</p> <p>Reaction as disabling function 2 when released</p> <p>...</p> <p>Reaction as disabling function 6 when pressed</p> <p>Reaction as disabling function 6 when released</p> <p>Internal scene recall scene 1</p> <p>Internal scene recall scene 2</p>	<p>Besides disabling of button functions, the panel can also trigger a special function immediately at the end of disabling.</p> <p>This function can... correspond to the function assigned to any of the buttons in the non-disabled state ("Reaction as button &gt;&gt;X&lt;&lt; ..."), be defined on the following parameter pages ("Reaction as disabling function ..."), recall a scene stored internally in the panel ("Internal scene recall ...").</p>

	...	
	Internal scene recall scene 8	
Button >>Y<<	<b>Button 1</b> Button 2 ... Button 24	If the panel is to perform the function of a specific button at the end of the disabling state, this button will be selected here.  Only visible if "Reaction of pushbutton sensor at the end of disabling = Reaction as button >>Y<< on pressing / releasing"!
□-  Disabling advanced parameters		
Behaviour during active disabling all buttons without function ...		All buttons of the panel are set to inoperable, i.e. disabled.
Behaviour during active disabling all buttons behave as ...		
Button column 1 behaves during disabling as	<b>Button 1</b> Button 2 ... Button 24 Disabling function 1 Disabling function 2 ... Disabling function 6	If a specific button or disabling function is to be assigned when disabling all buttons of this column, the desired button or disabling function can be selected here, the function of which will then be executed. During disabling, all buttons of this column behave as the button or disabling function configured here. There is <u>one</u> disabling object for all three columns.
During disabling, button column 2 behaves as	Button 1 Button 2 ... <b>Button 9</b> ... Button 24 Disabling function 1 Disabling function 2 ... Disabling function 6	See button column 1
During disabling, button column 3 behaves as	Button 1 Button 2 ... <b>Button 17</b> ... Button 24 Disabling function 1 Disabling function 2	See button column 1

...  
Disabling function 6

Behaviour during active  
disabling  
individual buttons  
without function ...

Depending on the settings for the  
buttons in the parameter group "Disable  
- Button selection", individual buttons  
are set to inoperable, i.e. disabled.

Button 1 (parameter  
field "Disable - Button  
selection")

Yes  
**No**

If the parameter is set to "No", the  
button retains its function.

Button 2...24 behave  
as "button 1"

The parameter field "Disable - Button  
selection" is only available when  
selecting "individual button without  
function or "individual buttons behave  
as".

Behaviour during active  
disabling  
individual buttons  
behave as ...

Button column 1  
behaves during  
disabling as

**Button 1**  
Button 2  
...  
Button 24  
Disabling function 1  
Disabling function 2  
...  
Disabling function 6

If a specific button or disabling function  
is to be assigned to all or individual  
buttons of the 1st column during  
disabling, the desired button or disabling  
function can be selected here, the  
function of which will then be executed.  
During disabling, all disabled buttons or  
selected buttons of this column behave  
as the button or disabling function  
configured here.

Individual buttons of this column in the  
parameter group "Disable - Button  
selection" can be removed from this  
functionality by setting the button 'n' (n =  
1 to 8) = "No".

Visible only if "Behaviour during active  
disabling = all buttons behave as" or  
"Behaviour during active disabling =  
individual buttons behave as" or  
"Behaviour during active disabling =  
individual columns behave as!"  
There is one disabling object for all three  
columns.

During disabling, button  
column 2 behaves as

Button 1  
Button 2  
...  
**Button 9**  
...  
Button 24  
Disabling function 1  
Disabling function 2

See button column 1.  
Individual buttons of this column in the  
parameter group "Disable - Button  
selection" can be removed from this  
functionality by setting the button 'n' (n =  
9 to 16) = "No".

	<p>... Disabling function 6</p>	
<p>During disabling, button column 3 behaves as</p>	<p>Button 1 Button 2 ... <b>Button 17</b> ... Button 24 Disabling function 1 Disabling function 2 ... Disabling function 6</p>	<p>See button column 1. Individual buttons of this column in the parameter group "Disable - Button selection" can be removed from this functionality by setting the button 'n' (n = 17 to 24) = "No".</p>
<p>Behaviour during active disabling individual columns without function...</p>		<p>Depending on the status of the disabling object of this column all buttons of this column are set to inoperable, i.e. disabled.</p> <p><b>i</b> There is a separate disabling object for <u>each column</u> (three disabling objects).</p>
<p>Behaviour during active disabling individual columns behave as ...</p>		
<p>During disabling, button column 1 behaves as</p>	<p><b>Button 1</b> Button 2 ... Button 24 Disabling function 1 Disabling function 2 ... Disabling function 6</p>	<p>If a specific button or disabling function is to be assigned to all buttons of this column during disabling, the desired button or disabling function can be selected here, the function of which will then be executed. During disabling, all buttons of this column behave as the button or disabling function configured here. There is a <u>separate</u> disabling object "B.Disable button column 1" available for the 1st column.</p>
<p>During disabling, button column 2 behaves as</p>	<p>Button 1 Button 2 ... <b>Button 9</b> ... Button 24 Disabling function 1 Disabling function 2 ... Disabling function 6</p>	<p>If a specific button or disabling function is to be assigned to all buttons of the 2nd column during disabling, the desired button or disabling function can be selected here, the function of which will then be executed. During disabling, all buttons of the 2nd column behave as the button or disabling function configured here. There is a <u>separate</u> disabling object "B.Disable button column 2" available for the 2nd column.</p>
<p>During disabling, button column 3 behaves as</p>	<p>Button 1 Button 2 ... <b>Button 17</b></p>	<p>If a specific button or disabling function is to be assigned to all buttons of the 3rd column during disabling, the desired button or disabling function can be</p>

...  
 Button 24  
 Disabling function 1  
 Disabling function 2  
 ...  
 Disabling function 6

selected here, the function of which will then be executed.  
 During disabling, all buttons of the 3rd column behave as the button or disabling function configured here.  
 There is a separate disabling object "B.Disable button column 3" available for the 3rd column.

☐ Disabling function 1

Function

No function  
**Switching**  
 Dimming  
 Venetian blind  
 1-byte value transmitter  
 2-byte value transmitter  
 Scene extension  
 2-channel operation

This parameter is used to define the basic function of the disabling function. The configured functions correspond exactly to those of the push-button functions and can be looked up there.

**i** Depending on the basic function setting the ETS displays different communication objects and parameters for a disabling function.

☐ For disabling function 2 ... 6 see disabling function 1!

## 4.1.5.4 Parameters of the Piezo signal device

Description	Values	Comment
□ Sound 1		
Pitch	Low <b>Medium</b> High	This parameter specifies the pitch.
Volume	Level 1 <b>Level 2</b> Level 3	This parameter specifies the volume. Level 1 is quiet, level 3 is loud.
On-time (0...255)*100ms	0 ... <b>3</b> ... 255	This parameter specifies the pulse time, i.e. the time in which the sound can be heard. Together with the off-time, a variety of sound intervals can be generated. Thus, for example, a discreet "beep" with longer pause times can be used for a less important signal, or an annoying, longer sound with short pause times can be used for a signal that should definitely be noticed.
Off-time (0...255)*100ms	0 ... <b>3</b> ... 255	This parameter specifies the pause time of the acoustic signal. The setting to "0"= corresponds to a continuous sound if at least one value greater than "1" is set for the on-time.
□ Sound 2 ... 4 see sound 1!		

## 4.1.5.5 Parameters of the dismantling alarm

Description	Values	Comment
<input type="checkbox"/> Dismantling alarms		
Dismantling alarms	Yes <b>No</b>	With this parameter, the disabling alarms of the panel can be centrally activated. If "Yes", the ETS shows further communication object and parameters.
Type of dismantling alarms	<b>Switching telegram</b> Value telegram	This parameter specifies whether a 1-bit switching telegram or 1-byte switching telegram should be transmitted during dismantling.
Switching command on dismantling alarm	<b>ON</b> OFF	This parameter sets the polarity of the alarm signalling switching command This parameter is only visible with the configuration "Type of dismantling alarms" = "Switching telegram".
Value telegram for dismantling alarm	1 ... 255	This parameter sets the 1-byte value of the value telegram for the dismantling alarm message. This parameter is only visible with the configuration "Type of dismantling alarms" = "Value telegram".
Automatic reset of the dismantling alarm	Yes <b>No</b>	If, after a dismantling message and proper subsequent mounting, the parameter selected is - "Yes", an inverted dismantling alarm telegram is transmitted automatically. - "No", the panel remains disabled until an enabling telegram with an inverted alarm value (switching command) or value telegram with the value "0" is received.
Alarm sound	No sound <b>Sound 1</b> Sound 2 Sound 3 Sound 4	This parameter specifies whether, and possibly, which signal is played back. The signals (sounds) can vary in the "Piezo" parameter group.



## 4.1.5.6 Parameters of the logic gate

Description	Values	Comment
<input type="checkbox"/> Logic gate 1 Logic gate 1	Yes  <b>No</b>	The device has logic gates (OR, AND, exclusive OR, AND with feedback) with up to 8 inputs and one output. This parameter activates the logic gate 1 and the other parameters and communication objects, if needed.
<input type="checkbox"/> Logic gate 1 advanced parameters		
Logic gate 1 Type of logic operation	AND <b>OR</b> Exclusive OR AND with feedback	Set here which logic operation is created by the gate  <input type="checkbox"/> <b>i</b> NAND and NOR gates can be implemented by inversion of the output
Logic gate 1 Number of inputs	1 <b>2</b> 3 4 5 6 7 8	This parameter specifies how many inputs the gate has with its own communication object.
Logic gate 1 Behaviour input 1	<b>Normal</b> Inverted	This parameter allows inversion of the input value.
Logic gate 1 Behaviour input for 2...8 as behaviour of input 1	<b>Each input event</b> Change the output	With the "Each input event" setting, the output status is transmitted for each telegram at an input as well as for a telegram repetition with the same value. If "Change the output" is selected, the status is only transmitted when there is an actual change in the output value.
Logic gate 1 Behaviour output	<b>Normal</b> Inverted	This parameter allows inversion of the output value.
Logic gate 1 Filter	<b>No</b> Only transmit ON telegrams Only transmit OFF telegrams	Filtering allows ON or OFF telegrams to be filtered out so that only ON or only OFF telegrams can be transmitted.

☐ For logic gate 2...5 see logic gate 1!

## 4.1.5.7 Parameters of the timers

Description	Values	Comment
☐ Timer 1		
Timer 1	Yes <b>No</b>	The device has timers that cause the Off telegrams, ON telegrams and/or ON and OFF telegrams - to be delayed - to be optionally inverted - or not to be transmitted. This parameter activates the timer 1 and the other parameters and communication objects, if needed.
☐ Timer 1 advanced parameters		
Timer 1 Function input -> output	On -> --- / Off -> --- On -> On / Off -> --- On -> --- / Off -> Off <b>On -&gt; On / Off -&gt; Off</b> On - / Off -> Off -> --- On -> --- Off -> On On -> Off / Off -> On On -> Toggle / Off -> --- On -> --- / Off -> Toggle On -> Toggle / Off -> Toggle	The timer can output a value settable here, depending on the value of the input object (filter function).
Timer 1 Time function	<b>No delay</b> ON telegram delayed OFF telegram delayed ON and OFF telegram delayed	The input telegram can be forwarded to the output of the filtering/timer after a delay (time delay function). It is possible to specify here whether the time delay is active and, if so, which switching state (ON and / or OFF) should be delayed. With "ON and OFF telegram delayed", the delay times can be set individually.
Timer 1 Time function Delay base for ON telegrams	<b>100 ms</b> 1 sec 1 min	This parameter defines the time base for the transmission delay of ON telegrams. This parameter is only visible on "Time function = ON telegram delayed" or "Time function = ON and OFF telegram delayed".
Timer 1 Time function	0...1...255	This parameter defines the time factor for the transmission delay of ON

Delay ON telegrams  
factor (0...255)

telegrams.  
This parameter is only visible on "Time  
function = ON telegram delayed" or  
"Time function = ON and OFF telegram  
delayed".

Timer 1                                   **100 ms**  
Time function                           1 sec  
Delay base for OFF                   1 min  
telegrams

This parameter defines the time base for  
the transmission delay of OFF  
telegrams.  
This parameter is only visible on "Time  
function = OFF telegram delayed" or  
"Time function = ON and OFF telegram  
delayed".

Timer 1                                   **0...1...255**  
Time function  
Delay base for OFF  
telegrams (factor  
(0...255).

This parameter defines the time factor  
for the transmission delay of OFF  
telegrams.  
This parameter is only visible on "Time  
function = OFF telegram delayed" or  
"Time function = ON and OFF telegram  
delayed".

Timer 1                                   **Yes**  
Disabling object                       **No**

The filtering/timer can also be fitted with  
a disabling object. If the disabling object  
is used, the timer ignores changes to the  
input when the disabling is active.

Timer 1                                   **1 = enabled /**  
Disabling object                       **0 = disabled**  
Polarity of disable  
object                                   0 = enabled /  
  1 = disabled

The telegram polarity of the disabling  
object can be set here.  
This parameter is only visible on  
"disabling object = Yes".

Timer 1                                   **Yes**  
Disabling object                       **No**  
Transmit after disabling  
is enabled

After the block, the current input value  
can immediately be processed further  
("Yes" setting), or the timer waits for the  
next input telegram ("No" setting).  
This parameter is only visible on  
"disabling object available = Yes".

☐ Timer 2...5 see Timer 1!

## 4.1.5.8 Parameters of the scene functions

Description	Values	Comment
<p>☐ Scene</p> <p>Scene function ?</p>	<p>Yes</p> <p><b>No</b></p>	<p>The device can internally handle eight scenes with eight actuator groups. This parameter activates the scene function and the other parameters and communication objects, if needed.</p>
<p>Overwrite scene values during ETS download?</p>	<p><b>Yes</b></p> <p>No</p>	<p>If the values of the actuator groups that have been changed on site by the used are to be reset to the values preset in the ETS during an application download by the ETS, the setting "Yes" must be chosen. If "No" is selected, the ETS values will not overwrite the scene values stored in the push button sensor, if any.</p>
<p>Scene 1 Recall via extension object with scene number</p>	<p>1 ... 64</p>	<p>If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the first scene.</p>
<p>Scene 2 Recall via extension object with scene number</p>	<p>1...2 ... 64</p>	<p>If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the second scene.</p>
<p>Scene 3 Recall via extension object with scene number</p>	<p>1...3 ... 64</p>	<p>If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the third scene.</p>
<p>Scene 4 Recall via extension object with scene number</p>	<p>1...4 ... 64</p>	<p>If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the fourth scene.</p>
<p>Scene 5 Recall via extension object with scene number</p>	<p>1...5 ... 64</p>	<p>If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the fifth scene.</p>

Scene 6 Recall via extension object with scene number	1... <b>6</b> ... 64	If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the sixth scene.
Scene 7 Recall via extension object with scene number	1... <b>7</b> ... 64	If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the seventh scene.
Scene 8 Recall via extension object with scene number	1... <b>8</b> ... 64	If the internal scenes are to be recalled via the extension object, a definite number is required for each of them. This parameter serves to specify the extension number of the eighth scene.
<input type="checkbox"/> Scene output 1 Data type	<b>Switching</b>  Value (0 ... 255)  Value / position of Venetian blind (0 ... 100%)	Selection of the data format of the scene output.
Scene 1 Switching command	<b>ON</b>  <b>OFF</b>	The switching command of the first scene can be predefined here. This parameter is only visible if "Data type = Switching".
Scene 1 Value (0 ... 255)	0... <b>204</b> ...255	The value of the first scene can be predefined here. This parameter is only visible if "Data type = Value (0...255)".
Scene 1 Value / position of Venetian blind (0 ... 100 %)	0... <b>80</b> ...100	The value of the first scene can be predefined here. This parameter is only visible if "Data type = Value / Venetian blind (0...100%)".
Scene 1 Allow save?	<b>Yes</b>  <b>No</b>	If the user is to be given the possibility of changing the value of the scene and of storing it while the system is running, this parameter must be set to "Yes".
Scene 1 Allow transmission?		If the state of an actuator group is to remain unchanged during the recall of a

---

	<b>Yes</b>	scene, this parameter can be set to "No". In this case, the push button sensor does not transmit a telegram via the scene output concerned during the recall of the scene. The scene output is deactivated for this scene.
	<b>No</b>	
Scene 1 Transmit delay (1 ... 1200 * 100 ms) (0 = deactivated)	<b>0...1200</b>	When the panel sends the telegrams to the various scene outputs, it can insert a presettable waiting time of 2 min. max. before each telegram. The bus load can be reduced by this. In this way, it is possible to have certain illumination switched on only after the shutters are really closed. If no delay is selected ("0" setting), the push button sensor sends the output telegrams with maximum speed. With this setting it may happen in some cases that the telegram sequence is not compatible with output numbering.

☐ Scene output 2 ... 8 see scene output 1!

## 5 Appendix

### 5.1 Index

<b>A</b>	
Activation via separate LED object.....	49
AND .....	62
AND with feedback.....	62
<b>B</b>	
button-click.....	59
<b>C</b>	
commissioning.....	11
communication objects.....	18
Comparator with sign.....	49
Comparator without sign.....	49
control surfaces.....	13
Controller status general.....	49
Controller status KNX.....	49
<b>D</b>	
disabling function.....	53
<b>E</b>	
ETS search paths.....	16
exclusive OR.....	62
<b>F</b>	
Fault message display.....	49
fault messages.....	59
<b>I</b>	
internal scene.....	41
<b>L</b>	
Limit value display.....	49
logic gates.....	62
Long-time operation.....	38
<b>O</b>	
Objects for the timers.....	34
Operating mode display (KNX controller) .....	49
Operation concept "long – short or short" .....	40
Operation concept "long – short".....	39
Operation concept "short – long – short" .....	38
Operation concept "short – long".....	39
OR .....	62
<b>P</b>	
pitch .....	59
<b>S</b>	
Scene control.....	56
Scene definition.....	56
scene recall.....	56
Scope of functions.....	17
Short-time operation.....	38
sound- /pause time.....	59
status LED.....	47
Status-LEDs.....	13-14
storage function.....	41
Storing scenes.....	57
<b>T</b>	
theft .....	60
timers.....	63
Toggle (Buttons).....	36
<b>V</b>	
volume.....	59



**ALBRECHT JUNG GMBH & CO. KG**

Volmestraße 1  
58579 Schalksmühle  
GERMANY

Telefon: +49 2355 806-0  
Telefax: +49 2355 806-204  
kundencenter@jung.de  
www.jung.de