EN

KNX eTR Sunblind

Button for Temperature Control and Solar Protection

KNX eTR 201 Sunblind
Item numbers
71090/93 (white), 71092/94 (black)

KNX eTR 202 Sunblind
Item numbers
71100/03 (white), 71102/04 (black)

elsner

Manual

This manual is amended periodically and will be brought into line with new software releases. The change status (software version and date) can be found in the contents footer. If you have a device with a later software version, please check **www.elsner-elektronik.de** in the menu area "Service" to find out whether a more up-to-date version of the manual is available.

Clarification of signs used in this manual

⚠ Safety advice.

⚡ Safety advice for working on electrical connections, components, etc.

DANGER! ... indicates an immediately hazardous situation which will lead to death or severe injuries if it is not avoided.

WARNING! ... indicates a potentially hazardous situation which may lead to death or severe injuries if it is not avoided.

CAUTION! ... indicates a potentially hazardous situation which may lead to trivial or minor injuries if it is not avoided.

STOP **ATTENTION!** ... indicates a situation which may lead to damage to property if it is not avoided.

ETS In the ETS tables, the parameter default settings are marked by underlining.

1. Safety and operating instructions

Installation, testing, operational start-up and troubleshooting should only be performed by a qualified electrician.

CAUTION!
Live voltage!

- Inspect the device for damage before installation. Only put undamaged devices into operation.
- Comply with the locally applicable directives, regulations and provisions for electrical installation.
- Immediately take the device or system out of service and secure it against unintentional switch-on if risk-free operation is no longer guaranteed.

Use the device exclusively for building automation and observe the operating instructions. Improper use, modifications to the device or failure to observe the operating instructions will invalidate any warranty or guarantee claims.
Operate the device only as a fixed-site installation, i.e. only in assembled condition and after conclusion of all installation and operational start-up tasks, and only in the surroundings designated for it.
Elsner Elektronik is not liable for any changes in norms and standards which may occur after publication of these operating instructions.

For information on installation, maintenance, disposal, scope of delivery and technical data, please refer to the installation instructions.

2. Description

The **KNX eTR 201/202 Sunblind push button** has touch-sensitive buttons with which functions can be called up in the KNX building bus system. The glass surface is printed with areas for setting the temperature and solar protection. White LEDs are integrated in these areas and their behaviour can be adjusted.

A temperature sensor is integrated into **KNX eTR 201/202 Sunblind**. An external temperature reading can be received via the bus and processed with its own data to create a total temperature (mixed value).

The **KNX eTR 201/202 Sunblind** has a PI controller for heating and cooling. The setpoint temperature can be changed using the "+" and "-" touch buttons.

Communication objects can be linked via AND and OR logic gates.

Functions:

- **Operating zone for temperature control** with 2 areas (warmer, cooler)

- **LEDs** can be set. All LEDs Off, all LEDs as ambient lighting, all LEDs individually controllable
- **Area function** when touching two or more push buttons. Can be configured as switch, selector switch, as 8 or 16 bit encoder or for scenario recall
- **Temperature** measurements. **Mixed value** from own measured value and external values (proportion can be set by percentage), output of minimum and maximum values
- **PI-controller for heating** (one or two-level) and **cooling** (one or two-level) according to temperature. Regulation according to separate setpoints or basic setpoint temperature
- **2 AND and 2 OR logic gates** each with 4 inputs. 8 logic inputs (in the form of communications objects) can be used as inputs for the logic gates. The output from each gate can be configured optionally as 1-bit or 2 x 8-bit

Additional functions KNX eTR 201 Sunblind:

- **1 operating zone for blinds**, awnings, shutters or windows with 2 areas (up/down with short/long distinction)

Additional functions KNX eTR 202 Sunblind:

- **2 operating zones for blinds**, awnings, shutters or windows with 2 areas (up/down with short/long distinction)

2.0.1. Area function

If the area function in ETS has been activated, another function is available alongside the regular key functions. This is triggered by touching multiple keys, e.g. if you touch the sensor with the palm of your hand.

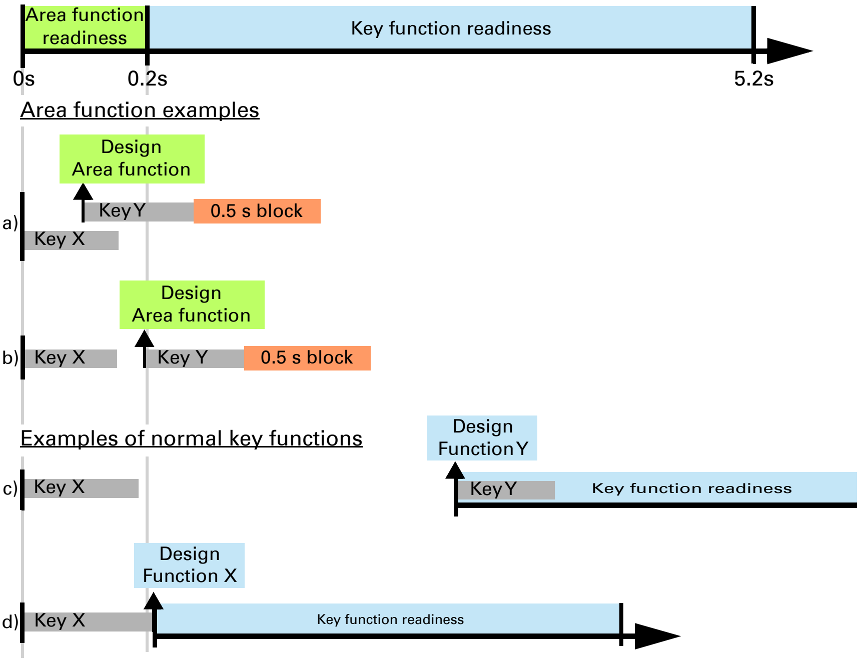
Using the area function

If a key is pressed and another (different) key is touched within 0.2 seconds, the action set in the ETS is performed for the area operation (See Fig. 1 a) and b)). The keys are then blocked for 0.5 seconds.

Using the normal key function

If a key is pressed and no other key is touched within 0.2 seconds, the normal key function is enabled/provided for 5 seconds (See Fig. 1 c) and d)). This is extended for 5 seconds with each push of the button.

Fig. 1



If the area function in the ETS is disabled, the keys can be used normally at any time.

3. Commissioning

Configuration is made using the KNX software as of ETS 5. The **product file** can be downloaded from the ETS online catalogue and the Elsner Elektronik website on **www.elsner-elektronik.de**.

After the bus voltage has been applied, the device will enter an initialisation phase lasting a few seconds. During this phase no information can be received or sent via the bus.

4. Addressing the device

The device is delivered ex works with the bus address 15.15.255. You program a different address in the ETS by overwriting the address 15.15.255 or teach the device using the programming button.

The programming button is located at the bottom outer side of the front panel of the device and is recessed. Use a thin object to reach the button, e. g. a 1.5 mm^2 wire. When the button is pressed, the temperature display on the front flashes.

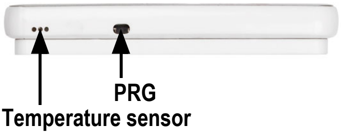


Fig. 2
View from bottom

5. Display and operation at the device

5.1. Adjust room temperature (using the example KNX eTR 201 Sunblind)

Depending on the setting of the "Display mode" parameter in the device application, the **KNX eTR 201/202 Sunblind push button** displays the current room temperature value (or mixed value), the target value or the shift in relation to the basic setpoint. The display can be dimmed and switched off via the bus so that *no* value is displayed.

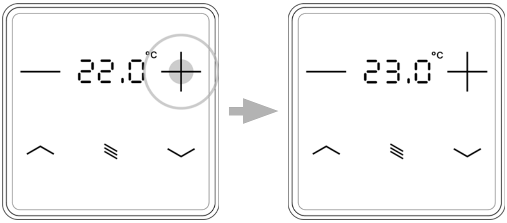
Option A: Display of actual temperature (room temperature)

The current room temperature is displayed. It is *not* possible to change the room temperature manually using the +/- buttons.

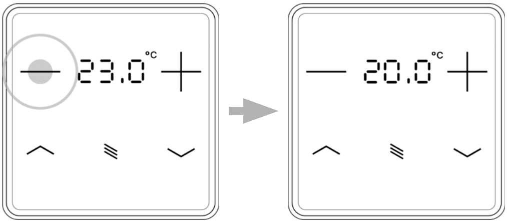
Option B: Display of target temperature or basic setpoint shift

Depending on the setting, the current target value or the shift relative to the base setpoint is displayed. The temperature can be changed by touching the +/- buttons.

Target value display (absolute value):

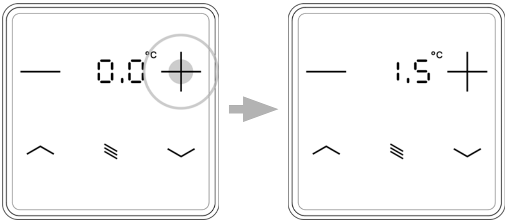


Tap +:
Increase room temperature (target temperature is increased)

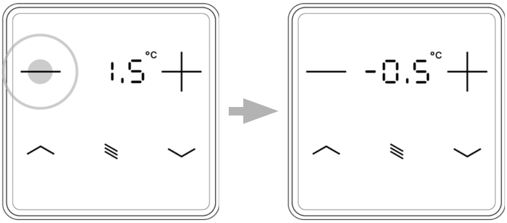


Tap -:
Lower room temperature (target temperature is lowered)

Display of the **basic setpoint shift** (change compared to the basic setpoint of the control):



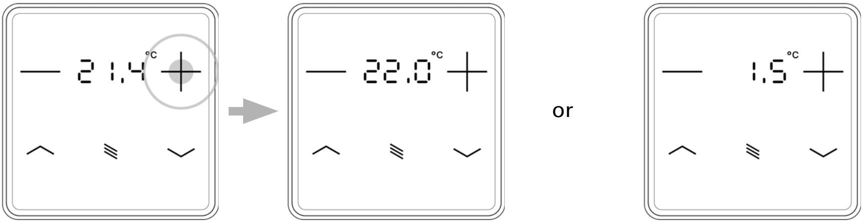
Tap +:
Increase room temperature
(Basic setpoint shift direction PLUS)



Tap -:
Lower room temperature
(Basic setpoint shift direction MINUS)

Option C: Display of actual temperature and target temperature/basic setpoint shift

During normal operation, the current room temperature is displayed. By touching the buttons, the display jumps to the target temperature or to the basic setpoint shift, depending on the presetting. Changes with + or - are visible. The display returns to the room temperature if no button is touched for 7 seconds.



Touch the **+ or -** button briefly: The current **target temperature** (or the basic setpoint shift) is displayed.

Tap +: Increase room temperature
(target temperature/basic setpoint shift is increased).

Tap -: Lower room temperature
(target temperature/basic setpoint shift is lowered).

General:

The step size for the change and the possible setting range are defined in the device application (ETS). There you can also define whether the manually changed values are retained after a mode change (e.g. Eco mode overnight) or reset to the stored values.

The button functions can be disabled in the ETS or locked due to operating mode with priority 1.

6. Transmission protocol

Units:

Temperatures in degrees Celsius

6.1. List of all communication objects

Abbreviations Flags:

C Communication
R Read
W Write
T Transmit
U Update

No	Text	Function	Flags	DPT type	Size
0	Software version	Output	C-WT-	[217.1] DPT_Version	2 Bytes
20	Temperature sensor: malfunction	Output	C-WT-	[1.1] DPT_Switch	1 Bit
21	Temperature sensor: measured value external	Input	-RWT-	[9.1] DPT_Value_Temp	2 Bytes
22	Temperature sensor: measured value	Output	C-WT-	[9.1] DPT_Value_Temp	2 Bytes
23	Temperature sensor: measured value total	Output	C-WT-	[9.1] DPT_Value_Temp	2 Bytes
24	Temperature sensor: measured value min./max. query	Input	-RW--	[1.17] DPT_Trigger	1 Bit
25	Temperature sensor: measured value minimum	Output	C-WT-	[9.1] DPT_Value_Temp	2 Bytes
26	Temperature sensor: measured value maximum	Output	C-WT-	[9.1] DPT_Value_Temp	2 Bytes
27	Temperature sensor: measured value min./max. reset	Input	-RW--	[1.17] DPT_Trigger	1 Bit
30	Temp. thresholdV 1: Absolute value	Input / Output	CRWT-	[9.1] DPT_Value_Temp	2 Bytes
31	Temp. thresholdV 1: (1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
32	Temp. thresholdV 1: Switching delay from 0 to 1	Input	-RW--	[7.5] DPT_TimePeriodSec	2 Bytes
33	Temp. thresholdV 1: Switching delay from 1 to 0	Input	-RW--	[7.5] DPT_TimePeriodSec	2 Bytes
34	Temp. thresholdV 1: Switching output	Output	C-WT-	[1.1] DPT_Switch	1 Bit
35	Temp. thresholdV 1: Switching output block	Input	-RW--	[1.1] DPT_Switch	1 Bit
36	Temp. thresholdV 2: Absolute value	Input / Output	CRWT-	[9.1] DPT_Value_Temp	2 Bytes

No	Text	Function	Flags	DPT type	Size
37	Temp. thresholdV 2: (1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
38	Temp. thresholdV 2: Switching delay from 0 to 1	Input	-RW--	[7.5] DPT_Time-PeriodSec	2 Bytes
39	Temp. thresholdV 2: Switching delay from 1 to 0	Input	-RW--	[7.5] DPT_Time-PeriodSec	2 Bytes
40	Temp. thresholdV 2: Switching output	Output	C-WT-	[1.1] DPT_Switch	1 Bit
41	Temp. thresholdV 2: Switching output block	Input	-RW--	[1.1] DPT_Switch	1 Bit
50	Temp.control: HVAC mode (priority 1)	Input / Output	CRWT -	depending on setting	1 Byte
51	Temp.control: HVAC mode (priority 2)	Input / Output	CRWT -	depending on setting	1 Byte
52	Temp.control: Mode frost/heat protection activt.	Input	CRWT -	[1.1] DPT_Switch	1 Bit
53	Temp.control: Block (1 = Blocking)	Input	-RW--	[1.1] DPT_Switch	1 Bit
54	Temp.control: Current setpoint	Output	C-WT-	[9.1] DPT_Value_Temp	2 Bytes
55	Temp.control: Switch. (0: Heating | 1: Cooling)	Input	-RW--	[1.1] DPT_Switch	1 Bit
56	Temp.control: Setpoint Comfort heating	Input / Output	CRWT -	[9.1] DPT_Value_Temp	2 Bytes
57	Temp.control: Setpoint Comfort heat.(1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
58	Temp.control: Setpoint Comfort cooling	Input / Output	CRWT -	[9.1] DPT_Value_Temp	2 Bytes
59	Temp.control: Setpoint Comfort cool.(1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
60	Temp.control: Basic 16-bit setpoint shift	Input	-RW--	[9.1] DPT_Value_Temp	2 Bytes
61	Temp.control: Setpoint Standby heating	Input / Output	CRWT -	[9.1] DPT_Value_Temp	2 Bytes
62	Temp.control: Setpoint Standby heat.(1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
63	Temp.control: Setpoint Standby cooling	Input / Output	CRWT -	[9.1] DPT_Value_Temp	2 Bytes
64	Temp.control: Setpoint Standby cool. (1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
65	Temp.control: Setpoint Eco heating	Input / Output	CRWT -	[9.1] DPT_Value_Temp	2 Bytes
66	Temp.control: Setpoint Eco heating (1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
67	Temp.control: Setpoint Eco cooling	Input / Output	CRWT -	[9.1] DPT_Value_Temp	2 Bytes

No	Text	Function	Flags	DPT type	Size
68	Temp.control: Setpoint Eco cooling (1:+ | 0:-)	Input	-RW--	[1.1] DPT_Switch	1 Bit
69	Temp.control: Control variable heating (level 1)	Output	C-WT-	[5.1] DPT_Scaling	1 Byte
70	Temp.control: Control variable heating (level 2)	Output	C-WT-	[5.1] DPT_Scaling	1 Byte
71	Temp.control: Control variable cooling (level 1)	Output	C-WT-	[5.1] DPT_Scaling	1 Byte
72	Temp.control: Control variable cooling (level 2)	Output	C-WT-	[5.1] DPT_Scaling	1 Byte
73	Temperature control: Variable for 4/6-way valve	Output	C-WT-	[5.1] DPT_Scaling	1 Byte
74	Temp.control: Status Heat. level 1 (1=ON|0=OFF)	Output	C-WT-	[1.1] DPT_Switch	1 Bit
75	Temp.control: Status Heat. level 2 (1=ON|0=OFF)	Output	C-WT-	[1.1] DPT_Switch	1 Bit
76	Temp.control: Status Cool. level 1 (1=ON|0=OFF)	Output	C-WT-	[1.1] DPT_Switch	1 Bit
77	Temp.control: Status Cool. level 2 (1=ON|0=OFF)	Output	C-WT-	[1.1] DPT_Switch	1 Bit
78	Temp.control: Comfort extension status	Input / Output	CRWT-	[1.1] DPT_Switch	1 Bit
79	Temp.control: Comfort Extension time	Input	CRWT-	[7.5] DPT_TimePeriodSec	2 Bytes
80	Temp. Controller: Fan coil levels 0 to 3	Output	C-WT-	[5.1] DPT_Scaling	1 Byte
81	Temp. Controller: Fan coil level 1	Output	C-WT-	[1.1] DPT_Switch	1 Bit
82	Temp. Controller: Fan coil level 2	Output	C-WT-	[1.1] DPT_Switch	1 Bit
83	Temp. Controller: Fan coil level 3	Output	C-WT-	[1.1] DPT_Switch	1 Bit
84	Temp. Controller: Fan coil auto=1 manual=0	Input / Output	CRWT-	[1.1] DPT_Switch	1 Bit
86	All LEDs On/Off	Input	-RW--	[1.1] DPT_Switch	1 Bit
87	All LEDs Brightness	Input	-RW--	[5.1] DPT_Scaling	1 Byte
88	LEDs Temperature display On/Off	Input	-RW--	[1.1] DPT_Switch	1 Bit
89	LEDs Temperature display Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
90	LED 1 on/off	Input	-RW--	[1.1] DPT_Switch	1 Bit
91	LED 1 Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
92	LED 2 on/off	Input	-RW--	[1.1] DPT_Switch	1 Bit
93	LED 2 Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
94	LED 3 on/off	Input	-RW--	[1.1] DPT_Switch	1 Bit
95	LED 3 Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
96	LED 4 on/off	Input	-RW--	[1.1] DPT_Switch	1 Bit

No	Text	Function	Flags	DPT type	Size
97	LED 4 Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
99	Area operation on/off	Input	-RW--	[1.1] DPT_Switch	1 Bit
100	Area operation Output: Switch	Output	C-WT-	depending on setting	2 Bytes
101	Drive 1 Long-term	Output	C-WT-	[1.8] DPT_Up-Down	1 Bit
102	Drive 1 Short-term	Output	C-WT-	[1.10] DPT_Start	1 Bit
103	Drive 2 Long-term	Output	C-WT-	[1.8] DPT_Up-Down	1 Bit
104	Drive 2 Short-term	Output	C-WT-	[1.10] DPT_Start	1 Bit
107	Logic input 1	Input	-RW--	[1.2] DPT_Bool	1 Bit
108	Logic input 2	Input	-RW--	[1.2] DPT_Bool	1 Bit
109	Logic input 3	Input	-RW--	[1.2] DPT_Bool	1 Bit
110	Logic input 4	Input	-RW--	[1.2] DPT_Bool	1 Bit
111	Logic input 5	Input	-RW--	[1.2] DPT_Bool	1 Bit
112	Logic input 6	Input	-RW--	[1.2] DPT_Bool	1 Bit
113	Logic input 7	Input	-RW--	[1.2] DPT_Bool	1 Bit
114	Logic input 8	Input	-RW--	[1.2] DPT_Bool	1 Bit
117	AND logic 1: 1 bit switching output	Output	C-WT-	[1.2] DPT_Bool	1 Bit
118	AND logic 1: 8 bit output A	Output	C-WT-	depending on setting	1 Byte
119	AND logic 1: 8 bit output B	Output	C-WT-	depending on setting	1 Byte
120	AND logic 1: Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
121	AND logic 2: 1 bit switching output	Output	C-WT-	[1.2] DPT_Bool	1 Bit
122	AND logic 2: 8 bit output A	Output	C-WT-	depending on setting	1 Byte
123	AND logic 2: 8 bit output B	Output	C-WT-	depending on setting	1 Byte
124	AND logic 2: Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
125	OR logic 1: 1 bit switching output	Output	C-WT-	[1.2] DPT_Bool	1 Bit
126	OR logic 1: 8 bit output A	Output	C-WT-	depending on setting	1 Byte
127	OR logic 1: 8 bit output B	Output	C-WT-	depending on setting	1 Byte
128	OR logic 1: Block	Input	-RW--	[1.1] DPT_Switch	1 Bit
129	OR logic 2: 1 bit switching output	Output	C-WT-	[1.2] DPT_Bool	1 Bit
130	OR logic 2: 8 bit output A	Output	C-WT-	depending on setting	1 Byte
131	OR logic 2: 8 bit output B	Output	C-WT-	depending on setting	1 Byte
132	OR logic 2: Block	Input	-RW--	[1.1] DPT_Switch	1 Bit

7. Setting the parameters

7.1. Behaviour on power failure/ restoration of power

Behaviour following a failure of the bus power supply:

The device sends nothing.

Behaviour on bus restoration of power and following programming or reset:

The device sends all outputs according to their send behaviour set in the parameters. Delays established in the "General settings" parameter block are taken into account.

7.2. General settings

Set the basic properties of the data transmission.

Transmission delay after reset/bus restoration	5 s • ... • 300 s
Maximum message rate	• 1 message per second • ... • 10 messages per second • ... • 50 messages per second

7.3. Temperature measured value

Select, whether a **malfunction object** is to be sent if the sensor is faulty.

Use malfunction object	No • Yes

When **measuring temperature**, the self-heating of the device is considered by the electronics. The heating is compensated for in the device.

Use **Offsets** to adjust the readings to be sent.
Measurement variations from permanent sources of interference can be corrected in this way.

Offset in 0.1°C	-50…50; 0

The unit can calculate a **mixed value** from its own reading and an external value. Set the mixed value calculation if desired. If an external portion is used, all of the following settings (threshold values, etc.) are related to the overall reading.

Use external measured value	No • Yes
Ext. Reading proportion of the total reading	5% • 10% • ... • 50% • ... • 100%

Sending pattern for internal and total measured value	• never • periodically • on change • on change and periodically
At and above change of *(if sent on change)*	0.1°C • 0.2°C • 0.5°C • ... • 5.0°C
Send cycle *(if sent periodically)*	5 s • 10 s • ... • 2 h

The **minimum and maximum readings** can be saved and sent to the bus. Use the "Reset temperature min/max. value" objects to reset the values to the current readings. The values are not retained after a reset.

Use minimum and maximum value	No • Yes

7.4. Temperature threshold values

Activate the required temperature threshold values. The menus for the further setting of the threshold values are then displayed.

Use threshold value 1 / 2	Yes • No

7.4.1. Threshold value 1, 2

Threshold value

Decide in which cases **threshold values and delay times** received are to be kept per object. The parameter is only taken into consideration if the setting by object is activated further down. Please note that the setting "After power restoration and programming" should not be used for the initial start-up, as the factory settings are always used until the first call (setting via objects is ignored).

Set the threshold values directly in the application program using parameters, or define them via the bus using a communication object.

Threshold value setpoint using parameter:

Set the threshold values and switching distance (hysteresis) directly.

Threshold value setpoint using	**Parameter** • Communication objects
Threshold in 0.1°C	-300… 800; 200

Threshold value setpoint using a communication object:

Beforehand, enter how the threshold value will be received from the bus. Basically, a new value can be received, or simply a command to increase or decrease.

During initial commissioning, a threshold value must be defined which will be valid until the first call with a new threshold value. For units which have already been taken into service, the last communicated threshold value can be used. Basically, a temperature range is given in which the threshold value can be changed (object value limit).

A set threshold value will be retained until a new value or a change is transferred. The current value is saved in so that it is retained in the event of a power supply failure and will be available again once the power supply is restored.

Threshold value setpoint using	Parameter • **Communication objects**
The last communicated value should	• never • after restoration of power • after power restoration and Programming
Start threshold value in 0.1°C valid until first Communication	-300 … 800; 200
Object value limit (min) in 0.1°C	-300…800
Object value limit (max) in 0.1°C	-300…800
Type of threshold value change	Absolute value • Increase/decrease
Interval *(upon increase/decrease change)*	0.1 °C • ... • 5°C

Set the **switching distance** independently of the type of threshold value setting.

Switching distance setting	in % • absolute
Switching distance in % of the threshold value	0 … 50; 20
Switching distance in 0.1°C	0 … 1100; 50

Switching output

Set the behaviour of the switching output when a threshold value is exceeded/undercut. The output switching delay can be set using objects or directly as a parameter.

When the following conditions apply, the output is (TV = Threshold value) (SD = Switching distance)	• TV above = 1 | TV - SD below = 0 • TV above = 0 | TV - SD below = 1 • TV below = 1 |TV + SD above = 0 • TV below = 0 |TV + SD above = 1
Delays can be set via objects (in seconds)	No • Yes
Switching delay from 0 to 1 *(if delay is adjustable via objects: valid until 1st communication)*	None • 1 s • 2 s • 5 s • 10 s • ... • 2 h
Switching delay from 1 to 0 *(if delay is adjustable via objects: valid until 1st communication)*	None • 1 s • 2 s • 5 s • 10 s • ... • 2 h
Switching output sends	• on change • on change to 1 • on change to 0 • on change and periodically • on change to 1 and periodically • on change to 0 and periodically
Cycle *(is sent only if "periodically" is selected)*	5 s • 10 s • 30 s... • 2 h

Block

The switching output can be blocked using an object.

Use switching output block	No • Yes

Set the cases in which threshold vlaues and delay times received per object are to be retained.

Assessment of the block object	• At value 1: block | At value 0: release • At value 0: block | At value 1: release
Blocking object value before first Communication	0 • 1
Switching output behaviour	
On blocking	• Do not send message • 0 Send • 1 Send
On release (with 2 second release delay)	[Dependent on the "Switching output sends" setting]

The behaviour of the switching output on release is dependent on the value of the parameter "Switching output sends" (see "Switching output")

Switching output sends on change	• Do not send message • Send switching output status
Switching output sends on change to 1	• Do not send message • if switching output = 1 ➔ send 1
Switching output sends on change to 0	• Do not send message • if switching output = 0 ➔ send 0
Switching output sends on change and periodically	Send switching output status
Switching output sends on change to 1 and periodically	if switching output = 1 ➔send 1
Switching output sends on change to 0 and periodically	if switching output = 0 ➔send 0

7.5. Temperature PI controller

Activate the controller if you want to use it.

Use controller	No • Yes

General rules

Decide in which cases **nominal values and delay times** received per object are to be kept. The parameter is only taken into consideration if the setting by object is activated further down. Please note that the setting "After power restoration and programming" should not be used for the initial start-up, as the factory settings are always used until the first call (setting via objects is ignored).

For an adequate regulation of the indoor temperature, comfort, standby, eco and building protection modes may be used.
Comfort when present,
Standby when absent,
Eco as a night-time mode and
Frost / heat protection (building protection) e.g. when the window is open.

The settings for the temperature control include the setpoint temperatures for the individual modes. Objects are used to determine which mode is to be selected. A change of mode may be triggered manually or automatically (e.g. by a timer, window contact).

The **mode** may be switched with two 8 bit objects of different priority. Objects
„... HVAC mode (Prio 2)“ for switching in everyday operation and
„... HVAC mode (Prio 1)“ for central switching with higher priority.
The objects are coded as follows:
0 = Auto
1 = Comfort
2 = Standby
3 = Eco
4 = Building protection

Alternatively, you can use three objects, with one object switching between eco and standby mode and the two others are used to activate comfort mode or frost/heat protection mode. The comfort object then blocks the eco/standby object, and frost/heat protection objects have the highest priority. Objects
„... Mode (1: Eco, 0: Standby)“,
„... comfort activation mode" and
„... frost/heat protection activation mode"
The Eco/Standby object is set to 1 = Eco after a reset. When comfort mode is exited, the system therefore switches to Eco by default. If the system is to switch to standby, the Eco/Standby object must have previously received a 0 = Standby.

Switch mode via	• two 8-bit objects (HVAC modes) • three 1-bit objects

Select the **mode to be activated after reset** (e.g. power failure, reset of the line via the bus). (Default).
Then configure a temperature control **block** using the blocking object.

Mode after reset	• Comfort • Standby • Eco • Building protection
Behaviour of the blocking object with value	• 1 = Block | 0 = release • 0 = block | 1 = release
Value of the blocking object after reset	0 • 1

Specify when the current **control variables** are to be **sent** to the bus. Periodic transmission is safer if a message does not reach a recipient. You may also set up periodical monitoring by the actuator with this setting.

Send control variable	• on change • on change and periodically
from change of (in % absolute)	1...10; 2
Cycle *(if sent periodically)*	5 s • ... • 5 min • ... • 2 h

The **status object** reports the current status of the output (0 = OFF, 0 = ON) and may for example be used for visualisation, or to switch off the heating pump as soon as the heating is switched off.

Send status objects	• on change • on change to 1 • on change to 0 • on change and periodically • on change to 1 and periodically • on change to 0 and periodically
Cycle *(if sent periodically)*	5 s • ... • 5 min • ... • 2 h

Then define the **type of setting**. Heating and/or cooling may be controlled in two stages.

Type of control	• Single stage heating • Dual-stage heating • Single-stage cooling • Single-stage heating + single-stage cooling • Dual-stage heating + single-stage cooling • Dual-stage heating + dual-stage cooling

General setpoint values

You may enter separate setpoint values for each mode or use the comfort setpoint as a basic value.

If you are using the control for both heating *and* cooling, you may also select the setting "separately with switching object". Systems used for cooling in summer and for heating in winter can thus be switched from one to the other.

If no switching object is selected, the actual temperature determines whether heating or cooling takes place. If the actual value is between the heating and cooling setpoint values, the existing operating mode is retained. If heating was previously used, the system remains in heating mode and continues to aim for this set point value. Only when the cooling setpoint is reached does the operating mode switch to cooling.

If cooling has been used so far, the system remains in cooling mode and continues to aim for this set point value. Only when the heating setpoint is reached does the operating mode switch to heating.

If the actual temperature is above the cooling set point value, cooling takes place; if it is below the heating set point value, heating takes place. The difference between the

heating set point value and the cooling set point value or the dead zone should be at least 1 °C. This prevents the control from switching too often between heating and cooling in the event of minor temperature fluctuations.
If you are using the basic value, only the deviation from the comfort setpoint value is listed for the other modes (e.g, 2°C less for standby mode).

Keep modified setpoints after mode change	No • Yes
Setting the nominal values	• separate with switching object • separate without switching object • with comfort setpoint as a basis with switching object • with comfort setpoint as a basis without switching object
Analysis of the status object / Behaviour of the switching object at value	• 0 = Heating | 1 = Cooling • 1 = Heating | 0 = Cooling
Switching object value before first Communication (*only if switching object is used*)	0 • 1

The **grades** for the setpoint changes is predefined. Whether the change remains active only temporarily (do not store) or remains stored even after restoration of power (and programming) is determined in the first section of "General controller". This also applies to a comfort extension.

Grading for setpoint changes (in 0.1 °C)	1... 50; 10
Storage of setpoint(s)	• not be retained • after power restoration • after restoration of power and programming

The controller can be switched from eco mode, i.e. night mode, to comfort mode via the comfort extension. This means that the comfort setpoint can be maintained for longer, for example when having guests. The duration of this comfort extension period is set. After the comfort extension period is terminated, the system returns to eco mode.

Comfort extension time in seconds (can only be activated from eco mode)	1…36000; 3600

Setpoint Comfort

Comfort mode is usually used for daytime mode when people are present. A starting value is defined for the comfort setpoint as well as a temperature range in which the nominal value may be modified.

Initial heating/cooling setpoint (in 0.1 °C) valid until first Communication *(not upon saving the target value after programming)*	-300…800; 210

If setpoint values are entered separately:

Min. object value heating/cooling (in 0.1 °C)	-300…800; 160
Max. object value heating/cooling (in 0.1 °C)	-300…800; 280

If the comfort setpoint value is used as a basis:

If the comfort setpoint is used as the basis, the increase/decrease of this value is indicated.

Minimum base setpoint (in 0.1°C)	-300…800; 160
Maximum base setpoint (in 0.1°C)	-300…800; 280
Reduction by up to (in 0.1°C)	1…100; 50
Increase by up to (in 0.1°C)	1…100; 50

If the comfort setpoint is used as the basis, a dead zone is determined for the control mode "heating *and* cooling" to avoid direct switching from heating to cooling.

Dead zone between heating and cooling *(only if both heating AND cooling are used)*	1…100; 50

Setpoint for standby

Standby mode is usually used for daytime mode when people are absent.

If setpoint values are entered separately:

A starting setpoint value is defined as well as a temperature range in which the setpoint value may be changed.

Heating initial setpoint (in 0.1 °C) valid until first Communication	-300…800; 180
Cooling initial setpoint (in 0.1 °C) valid until first Communication	-300…800; 240
Min. object value heating/cooling (in 0.1 °C)	-300…800; 160
Max. object value heating/cooling (in 0.1 °C)	-300…800; 280

If the comfort setpoint value is used as a basis:

If the comfort setpoint is used as the basis, the increase/decrease of this value is indicated.

Reduce heating setpoint (in 0.1°C) *(for heating)*	0…200; 30
Increase cooling setpoint (in 0.1°C) *(for cooling)*	0…200; 30

Eco setpoint

Eco mode is usually used for night mode.

If setpoint values are entered separately:

A starting setpoint value is defined as well as a temperature range in which the setpoint value may be changed.

Heating initial setpoint (in 0.1 °C) valid until first Communication	-300…800; 160
Cooling initial setpoint (in 0.1 °C) valid until first Communication	-300…800; 280
Min. object value heating/cooling (in 0.1 °C)	-300…800; 160
Max. object value heating/cooling (in 0.1 °C)	-300…800; 280

If the comfort setpoint value is used as a basis:

If the comfort setpoint is used as the basis, the increase/decrease of this value is indicated.

Reduce heating setpoint (in 0.1°C) *(for heating)*	0…200; 50
Increase cooling setpoint (in 0.1°C) *(for cooling)*	0…200; 60

Setpoint values for frost/heat protection (building protection)

The building protection mode is used, for example, as long as windows are open for ventilation. Setpoints for frost protection (heating) and heat protection (cooling) are determined which may not be modified from outside (no access via operating devices etc.). The building protection mode may be activated with delay, which allows you to leave the building before the controls switch to frost/heat protection mode.

Nominal value frost protection\r\n(in 0,1°C)	-300…800; 70
Activation delay	no • 5 s • ... • 5 min • ... • 2 h
Nominal value heat protection (in 0,1°C)	-300…800; 350
Activation delay	no • 5 s • ... • 5 min • ... • 2 h

General variables

This setting appears for the control types "Heating *and* Cooling" only. This is where you can decide whether to use a common variable for heating and cooling. If the 2nd stage has a common variable, this is also where you determine the control mode of the 2nd stage.

For heating and cooling	• separate variables are used • common variables are used for Stage 1 • common variables are used for Stage 2 • common variables are used for Stage 1+2

Use variable for 4/6-way valve *(only with common variable at stage1)*	No • Yes
Control type *(for stage 2 only)*	• 2-point-control • PI control
Regulating variable of the 2nd Stage is on *(only for stage 2 with 2-point control)*	• 1-bit object • 8-bit object

When using the variable for a 4/6 way valve:
0%...100% Heating = 66%...100% variable
OFF = 50% variable
0%...100% Cooling = 33%...0% variable

7.5.1. Heating control stage 1/2

If a heating control mode is configured, one or two setting sections for the heating stages are displayed.

In the first stage, heating is controlled by a PI controller which allows to either enter control parameters or select predetermined applications.

In the second stage (therefore only in case of 2-stage heating), heating is controlled via a PI or a 2-point-control.
In stage 2, the setpoint deviation between the two stages must also be specified, i.e. beyond which setpoint undershoot the second stage is then added.

Setpoint difference between stages 1 and 2 stages (in 0.1°C) *(At stage 2)*	0...100; 40
Control type *(at stage 2, no common variables)*	• 2-point-control • PI control
Control variable is on *(for stage 2 with 2-point control, no common variables)*	• 1-bit object • 8-bit object

PI controller with control parameters:

This setting allows individual input of the parameters for PI control.

Control type	• **PI control**
Setting of the controller by	• **Controller parameter** • specified applications

Specify the deviation from the setpoint value at which the maximum control variable value is reached, i.e. the point at which maximum heating power is activated.
The reset time shows how quickly the controller responds to deviations from the setpoint. In case of a short reset time, the control responds with a fast increase of the control variable. In case of a long reset time, the control responds somewhat less urgently

and needs longer until the necessary control variable for the setpoint deviation is reached.
You should set the time appropriate to the heating system at this point (observe manufacturer's instructions).

Maximum control variable is reached at setpoint/actual difference of (in °C)	1...5
Reset time (in min.)	1...255; 30

Now specify what should be sent when the control is blocked. Set a value greater 0 (=OFF) to receive a basic heating stage, e.g. for floor heating.
On release, the control variable follows the rule again.

When blocked, the control variable should	• not be sent • send a specific value
Value (in %) *(only if a value is sent)*	0...100

In case of a common control variable for heating and cooling, 0 is always transmitted as a fixed value.

PI control with predetermined application:

This setting provides fixed parameters for frequent applications.

Control type	• **PI control**
Setting of the controller by	• Controller parameter • **specified applications**
Application	• Warm water heating • Floor heating • Convection unit • Electric heating
Maximum control variable is reached at setpoint/actual difference of (in °C)	Warm water heating: 5 Floor heating: 5 Convection unit: 4 Electric heating: 4
Reset time (in min.)	Warm water heating: 150 Floor heating: 240 Convection unit: 90 Electric heating: 100

Now specify what should be sent when the control is blocked. Set a value greater 0 (=OFF) to receive a basic heating stage, e.g. for floor heating.
On release, the control variable follows the rule again.

When blocked, the control variable should	• not be sent • send a specific value
Value (in %) *(only if a value is sent)*	0...100

In case of a common control variable for heating and cooling, 0 is always transmitted as a fixed value.

2-point-control (only stage 2):

2-point-control is used for systems which are only set to ON or OFF.

Control type *(is determined at a higher stage for common variables)*	• **2-point-control**

Enter the switching distance that prevents frequent on/off switching of temperatures in the threshold range.

Switching distance (in 0.1°C)	0...100; 20

If separate variables are used, select whether the variable of the 2nd stage is a 1-bit object (on/off) or an 8-bit object (on with percentage value/off).

Control variable is on	• 1-bit object • 8-bit object
Value (in %) *(for 8-bit object)*	0...100

Now specify what should be sent when the control is blocked. Set a value greater 0 (=OFF) to receive a basic heating stage, e.g. for floor heating. On release, the control variable follows the rule again.

When blocked, the control variable should	• not be sent • send a specific value
Value (in %) *only if a value is sent*	0...100

7.5.2. Cooling control stage 1/2

If a cooling control mode is configured, one or two setting sections for the cooling levels are displayed.

In the first stage, cooling is controlled by a PI controller in which either control parameters or predetermined applications can be selected.

In the second stage (therefore only ifor 2-stage cooling), cooling is controlled via a PI or a 2-point-control.
In stage 2, the setpoint deviation between the two stages must also be specified, i.e. beyond which setpoint value undershoot the second stage is then added.

Setpoint difference between stages 1 and 2 stages (in 0.1°C) *(At stage 2)*	0...100; 40
Control type *(at stage 2, no common variables)*	• 2-point-control • PI control
Control variable is on *(for stage 2 with 2-point control, no common variables)*	• 1-bit object • 8-bit object

PI controller with control parameters:

This setting allows individual input of the parameters for PI control.

Control type	• **PI control**
Setting of the controller by	• **Controller parameter** • specified applications

Specify the deviation from the setpoint value which reaches maximum variable value, i.e. the point at which maximum cooling power is activated.
The reset time shows how quickly the controller responds to deviations from the setpoint. In case of a short reset time, the control responds with a fast increase of the control variable. In case of a long reset time, the control responds somewhat less urgently and needs longer until the necessary control variable for the setpoint deviation is reached. You should set the time appropriate to the cooling system at this point (observe manufacturer's instructions).

Maximum control variable is reached at setpoint/actual difference of (in °C)	1...5
Reset time (in min.)	1...255; 30

Now specify what should be sent when the control is blocked.
On release, the control variable follows the rule again.

When blocked, the control variable should	• not be sent • send a specific value
Value (in %) *(only if a value is sent)*	0...100

In case of a common control variable for heating and cooling, 0 is always transmitted as a fixed value.

PI control with predetermined application:

This setting provides fixed parameters for a cooling ceiling

Control type	• **PI control**
Setting of the controller by	• Controller parameter • **specified applications**
Application	• Cooling ceiling
Maximum control variable is reached at setpoint/actual difference of (in °C)	Cooling ceiling: 5
Reset time (in min.)	Cooling ceiling: 30

Now specify what should be sent when the control is blocked.
On release, the control variable follows the rule again.

When blocked, the control variable should	• not be sent • send a specific value
Value (in %) *(only if a value is sent)*	0...100

2-point-control (only stage 2):

2-point-control is used for systems which are only set to ON or OFF.

Control type *is determined at a higher stage for common variables*	• **2-point-control**

Enter the switching distance that prevents frequent on/off switching of temperatures in the threshold range.

Switching distance (in 0.1°C)	0...100; 20

If separate variables are used, select whether the variable of the 2nd stage is a 1-bit object (on/off) or an 8-bit object (on with percentage value/off).

Control variable is on	• 1-bit object • 8-bit object
Value (in %) *(for 8-bit object)*	0...100

Now specify what should be sent when the control is blocked.
On release, the control variable follows the rule again.

When blocked, the control variable should	• not be sent • send a specific value
Value (in %) *(only if a value is sent)*	0...100

In case of a common control variable for heating and cooling, 0 is always transmitted as a fixed value.

7.6. LEDs

Set the LED mode.

LED mode	• All LEDs off • All LEDs as ambient lighting • All LEDs individually controllable

All LEDs as ambient lighting

If all LEDs are to be used as ambient lighting, illuminate all simultaneously. Set the brightness of the LEDs, whether objects are used and whether the LEDs switch off automatically after pushing the button.

LED mode	All LEDs as ambient lighting
Brightness	0 ... 100%; 30%
Use objects	No • Yes
Object value on/off after reset (*if objects are used*)	0 • 1

Use automatic switch-off after pressing a button	No • Yes
Switch off after *(is automatic switch off is used)*	1 ... 255; 30 secs. after pressing

All LEDs individually controllable

This is where you set the brightness of the LEDs, whether objects are used and whether the LEDs switch off automatically after pushing the button.

LED mode	All LEDs individually controllable
Brightness	0 ... 100%; 30%
Use objects	No • Yes
Object value on/off after reset (*if objects are used*)	0 • 1
Function LED 1 (top left) / 2 (top right) / 3 (bottom left) / 4 (bottom right)	• always OFF • On after pressing button for settable time • On for object value = 1 / Off for object value = 0
Lighting period after pressing (*if "On after pressing button for a settable time"*)	1 ... 5 seconds ; 3 seconds
Object value after reset (*if "On for object value = 1 / Off for object value = 0*)	0 • 1
Use block object (*if "On for object value = 1 / Off for object value = 0*)	No • Yes

Specify here what the LEDs for temperature display should indicate.

LEDs for temperature display	• display actual value only • display only target value/base shift • display actual value and target value/base shift

7.7. Buttons

KNX eTR 201/202 Sunblind has an area control, i.e. if you touch 2 or more buttons at the same time, you can activate another function. The value (0 or 1) can be set for the evaluation of the activation of the surface operation.

Set whether you want to use the screen control.

Use screen control	No • Yes
Use screen control	Yes
Activation object assessment	• Value 1 = active | Value 0 = inactive • Value 0 = active | Value 1 = inactive
Object value after reset	0 • 1

Function	• Switch • Selector switch • 8 bit value 0 ... 255 • 8 bit value 0 ... 100% • 16-bit floating point value • Scenario recall
Value (*switch*)	0 • 1
Value (*8 bit value 0 ... 255*)	0 • ... • 255
Value (*8 bit value 0 ... 100%*)	0 • ... • 100
Value in 0.1 (*16 bit value floating point*)	-6707600 • ... • 6707600; 10
Scenario number(*Scenario recall*)	1 • ... • 64

Activate the required drive. The menus for further adjustment of the drive will then be displayed.

Use drive line	No • Yes

The KNX eTR 202 Sunblind has 2 drives that can be activated.

7.7.1. Drive

If the push-button is used to control a drive, select the "blind", "awning", "shutter" or "window" function and define the push-button function and control mode.

Blind

Function	**Shutter**
Swap Up/Down	No • Yes
Control mode	• Standard • Standard inverted • Comfort mode • Dead man switch

Standard:

Behaviour for button actuation (up): short = Stop/Step | long = Up Behaviour for button actuation (down): short = Stop/Step | long = Down	
Time between tap and hold (0.1 sec)	0 ... 50; 10

Standard inverted:

Behaviour for button actuation (up): long = Stop/Step | short = Up Behaviour for button actuation (down): short = Stop/Step | long = Down	
Time between tap and hold (0.1 sec)	0 ... 50; 10
Repetition of the step command upon extended button press	none • every 0.1 s • ... • every 0.5 s • ... • every 2 s

Comfort mode:

Push-button is pressed and	
released before expiry of Time 1 Maintained for longer than Time 1 Triggered between Times 1 and 1 + 2 Triggered after Times 1 + 2	Stop/Step Up | Down Stop No more stop
Time 1 (in 0.1 sec)	0 ... 50; 4
Time 2 (in 0.1 sec)	0 ... 50; 20

Dead man switch:

Press button	Up | Down command
Release button	Stop command

Shutter

Function	**Shutters**
Swap Up/Down	No • Yes
Control mode	• Standard • Standard inverted • Comfort mode • Dead man switch

Standard:

Behaviour for button actuation (up): short = Stop | long = Up Behaviour for button actuation (down): short = Stop | long = Down Behaviour for button actuation (up/down): short = Stop | long = Up/Down	
Time between tap and hold (0.1 sec)	0 ... 50; 10

Standard inverted:

Behaviour for button actuation (up): long = Stop | short = Up Behaviour for button actuation (down): short = Stop | long = Down Behaviour for button actuation (up/down): short = Stop/Step | long = Up/Down	
Time between tap and hold (0.1 sec)	0 ... 50; 10
Repetition of the step command upon extended button press (only for up)	none • every 0.1 s • ... • every 0.5 s • ... • every 2 s

Comfort mode:

Push-button is pressed	
released before expiry of Time 1 Maintained for longer than Time 1 Triggered between Times 1 and 1 + 2 Triggered after Times 1 + 2	Stop Up | Down | Up/Down Stop No more stop
Time 1 (in 0.1 sec)	0 ... 50; 4
Time 2 (in 0.1 sec)	0 ... 50; 20

Dead man switch:

Press button	Up | Down | Up/Down command
Release button	Stop command

Awning

Function	**Awning**
Swap Retract/Extend	No • Yes
Control mode	• Standard • Standard inverted • Comfort mode • Dead man switch

Standard:

Behaviour for button actuation (retract): short = Stop | long = Retract Behaviour for button actuation (extend): short = Stop | long = Extend Behaviour for button actuation (retract/extend): short = Stop | long = Extend/Retract	
Time between tap and hold (0.1 sec)	0 ... 50; 10

Standard inverted:

Behaviour for button actuation (retract): long = Stop | short = Retract Behaviour for button actuation (extend): long = Stop | short = Extend Behaviour for button actuation (retract/extend): hold = long | stop = Extend/Retract	
Time between tap and hold (0.1 sec)	0 ... 50; 10
Repetition of the step command upon extended button press (only for up)	none • every 0.1 s • ... • every 0.5 s • ... • every 2 s

Comfort mode:

Push-button is pressed and	
released before expiry of Time 1	Stop
Maintained for longer than Time 1	Retract • Extend • Retract/Extend
Triggered between Times 1 and 1 + 2	Stop
Triggered after Times 1 + 2	No more stop
Time 1 (in 0.1 sec)	0 ... 50; 4
Time 2 (in 0.1 sec)	0 ... 50; 20

Dead man switch:

Press button	Retract | Extend | Retract/Extend Command
Release button	Stop command

Window

Function	**Window**

Swap Open/Close	No • Yes
Control mode	• Standard • Standard inverted • Comfort mode • Dead man switch

Standard:

Behaviour for button actuation (close): short = Stop | long = Close Behaviour for button actuation (open): short = Stop | long = Open Behaviour for button actuation (open/close): short = Stop | long = Open/Close	
Time between tap and hold (0.1 sec)	0 ... 50; 10

Standard inverted:

Behaviour for button actuation (close): long = Stop | short = Close Behaviour for button actuation (open): long = Stop | short = Open Behaviour for button actuation (open/close): long = Stop | short = Open/Close	
Time between tap and hold (0.1 sec)	0 ... 50; 10
Repetition of the step command upon extended button press (only for up)	none • every 0.1 s • ... • every 0.5 s • ... • every 2 s

Comfort mode:

Push-button is pressed and	
released before expiry of Time 1 Maintained for longer than Time 1 Triggered between Times 1 and 1 + 2 Triggered after Times 1 + 2	Stop Close | Open | Open/Close Stop No more stop
Time 1 (in 0.1 sec)	0 ... 50; 4
Time 2 (in 0.1 sec)	0 ... 50; 20

Dead man switch:

Press button	Close | Open | Open/Close command
Release button	Stop command

7.7.2. Control modes for drive control

Behaviour on button actuation in standard control mode:

	short	**hold**
Blind	Stop/Step	Up or down
Shutters	Stop	Up or down
Awning	Stop	Retract or extend
Windows	Stop	Closed or open

Standard:

If briefly operated, the drive will move incrementally or stops. If operated longer, the drive will move up to the end position. The time difference between "short" and "long" is set individually.

Control mode	**Standard**
Behavior during button operation: short = stop/increment long = Up or Down	
Time between short and long in 0.1 seconds	1...50; 10

Standard inverted:

When pushed shortly, the drive moves up to the end position. When pushed for longer, the drive moves incrementally or stops. The time difference between "short" and "long" and the repeat interval is set individually.

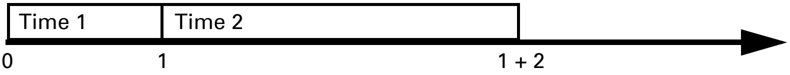
Control mode	**Standard inverted**
Behavior during button operation: short = Up or Down long = Stop/Step	
Time between short and long in 0.1 seconds	1...50; 10
Repeat the step command for a long button press	every 0.1 s • every 2 sec; every 0.5 sec

Comfort mode:

In the **comfort mode** actuating the button briefly, a bit longer and long will trigger different responses of the drive. The time intervals are set individually.

Short actuation (shorter than Time 1): The drive is positioned step-wise and stopped.
Holding it slightly longer (longer than Time 1, but shorter than Time 1+2): Drive running. Drive stops when the button is released.
Long holding (release after Time 1+2 runs out): Drive moves independently to the end position. The movement can be interrupted by a short tap.

Fig. 3
Time interval comfort mode diagram



Point in time 0:	*Actuate of button, start of time 1*
Release before time 1 expired:	*step (or stop if drive is moving)*
Point in time 1:	*End of time 1, start of time 2* *Moving command*
Release after time 1 expired but before time 2 expires:	*Stop*
Release after time 1 + 2 expired:	*Move into end position*

Control mode	**Comfort mode**
Behavior during button operation: Button is pushed and released before time 1 expired = stop/step held longer than time 1 = Up or Down released between time 1 and 1-2= stop released after time 1 +2 = no more stop	
Time 1	0.0s ... • 2 s; 0.4 s
Time 2	0 s • 2 s; 2 s

Dead man's switch:

The drive moves as soon as the button is actuated and stops as soon as the button is released.

Control mode	**Dead man's switch**
Behavior during button operation: Push button = Up or Down command Release button = Stop command	

7.8. Logic

The device has 8 logic inputs, 2 AND and 2 OR logic gates.

Activate the logic inputs and assign object values up to 1st communication.

Use logic inputs	No • Yes
Object value prior to 1st communication for	
- Logic input 1 ... 8	0 • 1

Activate the required logic outputs

AND logic

AND logic 1/2	not active • active

OR logic

OR logic 1/2	not active • active

7.8.1. AND logic outputs 1/2 and OR logic outputs 1/2

The same setting options are available for AND and OR logic.

Each logic output may transmit one 1-bit or two 8-bit objects. Determine what the output should send if logic = 1 and = 0.

1. / 2. / 3. / 4. Input	• Do not use • Logic inputs 1...8 • Logic inputs 1...8 inverted • Temperature sensor malfunction = ON • Temperature sensor malfunction = OFF *only with OR logic:* • Switching output AND logic 1/2 • Switching output AND logic 1/2 inverted
Output type	• a 1-bit-object • sends two 8-bit objects

If the **output type is a 1-bit object**, set the output values for the various conditions.

Output value if logic = 1	1 • 0
Output value if logic = 0	1 • 0
Output value If block active	1 • 0
Output value if monitoring time exceeded	1 • 0

If the **output type is two 8-bit objects**, set the type of object and the output values for the various conditions.

Object type	• value (0...255) • Percent (0...100%) • Angle (0...360°) • Scene call-up (0...63)
Output value object A if logic = 1	0 ... 255 / 100% / 360° / 63; 1
Output value object B if logic = 1	0 ... 255 / 100% / 360° / 63; 1
Output value object A if logic = 0	0 ... 255 / 100% / 360° / 63; 0
Output value object B if logic = 0	0 ... 255 / 100% / 360° / 63; 0

Output value object A If block active	0 … 255 / 100% / 360° / 63; 0
Output value object B If block active	0 … 255 / 100% / 360° / 63; 0
Output value object A if monitoring time exceeded	0 … 255 / 100% / 360° / 63; 0
Output value object B if monitoring time exceeded	0 … 255 / 100% / 360° / 63; 0

Set the output send pattern.

Transmission behaviour	• on change of logic • on change of logic to 1 • on change of logic to 0 • on change of logic and periodically • on change of logic to 1 and periodically • on change of logic to 0 and periodically • on change of logic + object receipt • on change of logic + object receipt and cyclically
Send cycle *(if sent periodically)*	5 s • 10 s • ... • 2 h

Block

If necessary, activate the block for the logic output and set what a 1 or 0 at the block input means and what happens in the event of a block.

Use block	No • Yes
Assessment of the block object	• At value 1: block | At value 0: release • At value 0: block | At value 1: release
Blocking object value before first Communication	0 • 1
Output behaviour when blocking	• Do not send message • Send block value [see above, output value if block active]
when released (with 2 second release delay)	• Do not send message • transmit value for current logic status

Monitoring

If necessary, activate the input monitoring. Set which inputs are to be monitored, at which intervals the inputs are to be monitored and what value the "monitoring status" should have, if the monitoring period is exceeded without a feedback being given.

Use input monitoring	No • Yes
Input monitoring	• 1 • 2 • 3 • 4 • 1 + 2 • 1 + 3 • 1 + 4 • 2 + 3 • 2 + 4 • 3 + 4 • 1 + 2 + 3 • 1 + 2 + 4 • 1 + 3 + 4 • 2 + 3 + 4 • 1 + 2 + 3 + 4

Monitoring period	5 s • ... • 2 h; 1 min
Output behaviour on exceeding the monitoring time	• Do not send message • Send value exceeding [= value of the parameter "Monitoring period"]

Questions about the product?

You can reach the technical service of Elsner Elektronik under
Tel. +49 (0) 70 33 / 30 945-250 or
service@elsner-elektronik.de

We need the following information to process your service request:

- Type of appliance (model name or item number)
- Description of the problem
- Serial number or software version
- Source of supply (dealer/installer who bought the device from Elsner Elektronik)

For questions about KNX functions:

- Version of the device application
- ETS version used for the project

elsner

Elsner Elektronik GmbH Control and Automation Engineering

Sohlengrund 16
75395 Ostelsheim
Germany

Phone +49 (0) 70 33 / 30 945-0
Fax +49 (0) 70 33 / 30 945-20

info@elsner-elektronik.de
www.elsner-elektronik.de