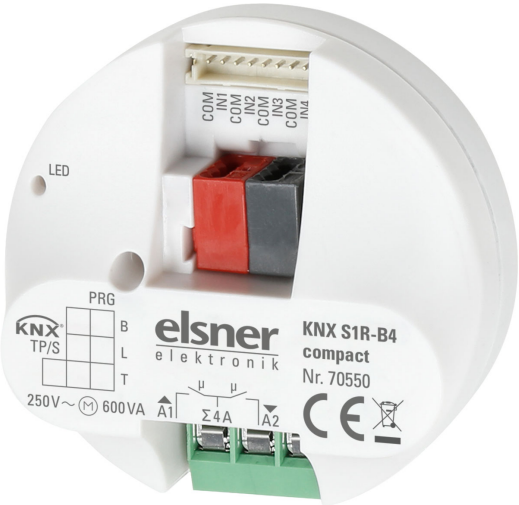
KNX S1R-B4 compact

Multifunctional Actuator

Item number 70550



elsner elektronik

Installation and Adjustment

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 usage instructions

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

DANGER!
Risk to life from live voltage (mains voltage)!
There are unprotected live components inside the device.
• 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.

1.1. Safety notice for automatic functions

WARNING!
Risk of injury from automatically moving components!
Parts of the system can be started by the automatic controls and be a danger to persons.

- No persons may remain in the travelling range of parts driven by an electric motor.
- Adhere to the relevant building regulations.
- Ensure that the return path/access to the building is not blocked if spending time outside the building (danger of being locked out).
- Correctly decommission the system for maintenance and cleaning work.

If there is a power outage, the system does not work. Therefore, shadings should be moved to a save position if there are anticipated weather conditions, for example, if this has not already been done by the automatic function (product protection).

If the power supply is removed, the connected drive switches off. When the power is restored, the consumer remains switched off until a new movement command is received by the actuator.

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

2. Description

The **Actuator KNX S1R-B4 compact** with integrated façade control has got a multifunctional output which can connect to either a drive with Up/Down control (blind, awning, shutter, window) or two switchable devices (On/Off for light and ventilation). Because the output is designed to be free of potential, other systems can also be controlled, e.g. via the manual switch input on a motor control unit.

Automation can be specified externally or internally. Internally, there are numerous options available for blocking, locking (e.g. master-slave) and priority definitions (e.g. manual-automatic). Scenes can be saved and called up via the bus (scene control with 16 scenes).

4 digital inputs are available for the connection of contacts like push-buttons.

Functions:

- Free of potential **multifunctional output** for **a 230 V drive** (shading, window) or connection of two **switchable devices** (light, fan)
- **4 binary inputs**
- **Position feedback** (movement position, also slat position for shutters)
- **Position storage** (movement position) via 1-bit object (storage and call-up e.g. via buttons)
- Parameters for taking drive and mechanics **downtimes** into account
- Control via **internal or external automation functions**
- Integrated **shade control** with **slat adjustment** for shutters based on the position of the sun
- **Scene control** for movement position with 16 scenes (also slat position for shutters)
- Mutual locking of two drives using **zero position sensors** prevents collisions e.g. of shade and window (master–slave)
- **Blocking objects and alarm reports** have different priorities, so safety functions always take precedence (e.g. wind block)
- Manual or automatic **priority** setting via time or communication object
- **Brief time limit** (movement command blocked) and **2 movement limits**

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** in the “Service” menu.

After the bus voltage has been applied, the device will enter an initialisation phase lasting approx. 5 seconds. During this phase no information can be received or sent via the bus.

3.1. Addressing of the device at the bus

The equipment is delivered with the individual address 15.15.255. This can be changed via the ETS. A button and a control LED are located on the unit for this purpose.

3.2. Monitor safety functions

For KNX devices with safety functions (e.g. wind or rain block), periodical monitoring of the safety objects must be set up.

For further explanations of KNX integration, see the manual chapters *Transfer protocol* and *Setting the parameters*.

4. Transfer protocol

4.1. List of all communication objects

Abbreviations:
R Read
W Write
C Communication
T Transmit
DPT Data Point Type

No	Text	Function	Flags	DPT type	Size
0	Software version	readable	R-C-	[217.1] DPT_Version	2 Bytes
1	Channel A - Automatic or manual status	Output	R-CT	[1.1] DPT_Switch	1 Bit
2	Channel A - Manual extended	Input	RWC-	[1.8] DPT_UpDown	1 Bit
3	Channel A - Manual short time	Input	RWC-	[1.8] DPT_UpDown	1 Bit
4	Channel A - Manual movement position	Input	RWC-	[5.1] DPT_Scaling	1 Byte
5	Channel A - Manual blind position	Input	RWC-	[5.1] DPT_Scaling	1 Byte
6	Channel A - Automatic extended	Input	RWC-	[1.8] DPT_UpDown	1 Bit
7	Channel A - Automatic short time	Input	RWC-	[1.8] DPT_UpDown	1 Bit
8	Channel A - Automatic movement position	Input	RWC-	[5.1] DPT_Scaling	1 Byte
9	Channel A - Automatic blind position	Input	RWC-	[5.1] DPT_Scaling	1 Byte
10	Channel A - Switch from manual to automatic	Input	RWC-	[1.1] DPT_Switch	1 Bit
11	Channel A - Automatic blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
12	Channel A - Current movement position	Output	R-CT	[5.1] DPT_Scaling	1 Byte
13	Channel A - Current blind position	Output	R-CT	[5.1] DPT_Scaling	1 Byte
14	Channel A - Status object	Output	R-CT	[1.1] DPT_Switch	1 Bit
15	Channel A - Manually move to position memory	Input	RWC-	[1.1] DPT_Switch	1 Bit

No	Text	Function	Flags	DPT type	Size
16	Channel A - Learn object position memory manually 0	Input	RWC-	[1.1] DPT_Switch	1 Bit
17	Channel A - Learn object position memory manually 1	Input	RWC-	[1.1] DPT_Switch	1 Bit
18	Channel A - Automatically move to position memory	Input	RWC-	[1.1] DPT_Switch	1 Bit
19	Channel A - Learn object position memory automatically 0	Input	RWC-	[1.1] DPT_Switch	1 Bit
20	Channel A - Learn object position memory automatically 1	Input	RWC-	[1.1] DPT_Switch	1 Bit
21	Channel A - Call up / saving scenes	Input	RWC-	[18.1] DPT_SceneControl	1 Byte
22	Channel A - Outdoor temperature blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
23	Channel A - Outdoor temperature block measurement	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
24	Channel A - Outdoor temperature blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
25	Channel A - Twilight object	Input	RWC-	[1.1] DPT_Switch	1 Bit
26	Channel A - Twilight measurement	Input	RWC-	[9.4] DPT_Value_Lux	2 Bytes
27	Channel A - Twilight status	Output	R-CT	[1.1] DPT_Switch	1 Bit
28	Channel A - Time control	Input	RWC-	[1.1] DPT_Switch	1 Bit
29	Channel A - Inside temperature release object	Input	RWC-	[1.1] DPT_Switch	1 Bit
30	Channel A - Inside temperature release measurement	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
31	Channel A - Inside temperature release setpoint	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
32	Channel A - Inside temperature release status	Output	R-CT	[1.1] DPT_Switch	1 Bit
33	Channel A - Shading object	Input	RWC-	[1.1] DPT_Switch	1 Bit
34	Channel A - Shading brightness measurement 1	Input	RWC-	[9.4] DPT_Value_Lux	2 Bytes
35	Channel A - Shading brightness measurement 2	Input	RWC-	[9.4] DPT_Value_Lux	2 Bytes
36	Channel A - Shading brightness measurement 3	Input	RWC-	[9.4] DPT_Value_Lux	2 Bytes

No	Text	Function	Flags	DPT type	Size
37	Channel A - Shading threshold value	Input / Output	RWCT	[9.4] DPT_Value_Lux	2 Bytes
38	Channel A - Shading threshold value 1 = + | 0 = -	Input	RWC-	[1.1] DPT_Switch	1 Bit
39	Channel A - Shading threshold value +	Input	RWC-	[1.1] DPT_Switch	1 Bit
40	Channel A - Shading threshold value -	Input	RWC-	[1.1] DPT_Switch	1 Bit
41	Channel A - Shading status	Output	R-CT	[1.1] DPT_Switch	1 Bit
42	Channel A - Shading position learning object	Input	RWC-	[1.1] DPT_Switch	1 Bit
43	Channel A - Azimuth	Input	RWC-	[9.7] DPT_Value_ Humidity	2 Bytes
44	Channel A - Elevation	Input	RWC-	[9.7] DPT_Value_ Humidity	2 Bytes
45	Channel A - Cold air intake blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
46	Channel A - Cold air intake outside temp. measurement value	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
47	Channel A - Cold air intake blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
48	Channel A - Forced ventilation	Input	RWC-	[1.1] DPT_Switch	1 Bit
49	Channel A - Warm air intake blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
50	Channel A - Warm air intake inside temp. measurement value	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
51	Channel A - Warm air intake outside temp. measurement value	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
52	Channel A - Warm air intake blocking target value	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
53	Channel A - Warm air intake blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
54	Channel A - Inside temperature opening object	Input	RWC-	[1.1] DPT_Switch	1 Bit
55	Channel A - Inside temperature opening measurement	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes
56	Channel A - Inside temperature opening setpoint	Input	RWC-	[9.1] DPT_Value_Temp	2 Bytes

No	Text	Function	Flags	DPT type	Size
57	Channel A - Inside temp. opening threshold value	Input / Output	RWCT	[9.1] DPT_Value_Temp	2 Bytes
58	Channel A - Inside temp. opening threshold value 1 = +	Input	RWC-	[1.1] DPT_Switch	1 Bit
59	Channel A - Inside temp. opening threshold value +	Input	RWC-	[1.1] DPT_Switch	1 Bit
60	Channel A - Inside temp. opening threshold value -	Input	RWC-	[1.1] DPT_Switch	1 Bit
61	Channel A - Inside temperature opening status	Output	R-CT	[1.1] DPT_Switch	1 Bit
62	Channel A - Inside humidity opening object	Input	RWC-	[1.1] DPT_Switch	1 Bit
63	Channel A - Inside humidity opening measurement	Input	RWC-	[9.7] DPT_Value_ Humidity	2 Bytes
64	Channel A - Inside humidity opening status	Output	R-CT	[1.1] DPT_Switch	1 Bit
65	Channel A - Zero position reached	Input	RWC-	[1.1] DPT_Switch	1 Bit
66	Channel A - Zero position sensor disrupted	Output	R-CT	[1.1] DPT_Switch	1 Bit
67	Channel A - Master zero position status	Output	R-CT	[1.1] DPT_Switch	1 Bit
68	Channel A - Master zero position command	Output	R-CT	[1.1] DPT_Switch	1 Bit
69	Channel A - Slave zero position status	Input	RWC-	[1.1] DPT_Switch	1 Bit
70	Channel A - Master zero position status	Input	RWC-	[1.1] DPT_Switch	1 Bit
71	Channel A - Master zero position command	Input	RWC-	[1.1] DPT_Switch	1 Bit
72	Channel A - Slave zero position status	Output	R-CT	[1.1] DPT_Switch	1 Bit
73	Channel A - Drive is moving	Output	R-CT	[1.1] DPT_Switch	1 Bit
74	Channel A - Object malfunction	Output	R-CT	[1.1] DPT_Switch	1 Bit
75	Channel A - Blocking 1 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
76	Channel A - Blocking 1 - Wind blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
77	Channel A - Blocking 1 - Wind blocking measurement	Input	RWC-	[9.5] DPT_Value_Wsp	2 Bytes

No	Text	Function	Flags	DPT type	Size
78	Channel A - Blocking 1 - Wind blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
79	Channel A - Blocking 1 - Rain blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
80	Channel A - Blocking 2 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
81	Channel A - Blocking 2 - Wind blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
82	Channel A - Blocking 2 - Wind blocking measurement	Input	RWC-	[9.5] DPT_Value_Wsp	2 Bytes
83	Channel A - Blocking 2 - Wind blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
84	Channel A - Blocking 2 - Rain blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
85	Channel A - Blocking 3 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
86	Channel A - Blocking 3 - Wind blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
87	Channel A - Blocking 3 - Wind blocking measurement	Input	RWC-	[9.5] DPT_Value_Wsp	2 Bytes
88	Channel A - Blocking 3 - Wind blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
89	Channel A - Blocking 3 - Rain blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
90	Channel A - Blocking 4 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
91	Channel A - Blocking 4 - Wind blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
92	Channel A - Blocking 4 - Wind blocking measurement	Input	RWC-	[9.5] DPT_Value_Wsp	2 Bytes
93	Channel A - Blocking 4 - Wind blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
94	Channel A - Blocking 4 - Rain blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
95	Channel A - Blocking 5 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
96	Channel A - Blocking 5 - Wind blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
97	Channel A - Blocking 5 - Wind blocking measurement	Input	RWC-	[9.5] DPT_Value_Wsp	2 Bytes

No	Text	Function	Flags	DPT type	Size
98	Channel A - Blocking 5 - Wind blocking status	Output	R-CT	[1.1] DPT_Switch	1 Bit
99	Channel A - Blocking 5 - Rain blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
100	Channel A - Movement restriction 1 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
101	Channel A - Movement restriction 2 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
102	Channel A - Short time restriction	Input	RWC-	[1.1] DPT_Switch	1 Bit
103	Channel A1 - Switch	Input	RWC-	[1.1] DPT_Switch	1 Bit
104	Channel A1 - Feedback	Output	R-CT	[1.1] DPT_Switch	1 Bit
105	Channel A1 - Status	readable	R-C-	[1.1] DPT_Switch	1 Bit
106	Channel A1 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
107	Channel A1 - Start staircase light function	Input	-WC-	[1.10] DPT_Start	1 Bit
108	Channel A1 - Start/stop staircase light function	Input	RWC-	[1.10] DPT_Start	1 Bit
109	Channel A1 - Link	Input	RWC-	[1.2] DPT_Bool	1 Bit
110	Channel A1 - Call up / saving scenes	Input	RWC-	[18.1] DPT_SceneControl	1 Byte
111	Channel A2 - Switch	Input	RWC-	[1.1] DPT_Switch	1 Bit
112	Channel A2 - Feedback	Output	R-CT	[1.1] DPT_Switch	1 Bit
113	Channel A2 - Status	readable	R-C-	[1.1] DPT_Switch	1 Bit
114	Channel A2 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
115	Channel A2 - Start staircase light function	Input	-WC-	[1.10] DPT_Start	1 Bit
116	Channel A2 - Start/stop staircase light function	Input	RWC-	[1.10] DPT_Start	1 Bit
117	Channel A2 - Link	Input	RWC-	[1.2] DPT_Bool	1 Bit
118	Channel A2 - Call up / saving scenes	Input	RWC-	[18.1] DPT_SceneControl	1 Byte
119	Channel A - Local control temperature blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
120	Input 1 - Extended	Input / Output	RWCT	[1.8] DPT_UpDown	1 Bit
121	Input 1 - Short	Output	R-CT	[1.8] DPT_UpDown	1 Bit
122	Input 1 - Switch	Input / Output	RWCT	[1.1] DPT_Switch	1 Bit

No	Text	Function	Flags	DPT type	Size
123	Input 1 - Relative dimming	Input / Output	RWCT	[3.7] DPT_Control_ Dimming	4 Bit
124	Input 1 - Encoder 8 Bit	Output	R-CT	[5] 5.xxx	1 Byte
125	Input 1 - Temperature encoder	Output	R-CT	[9.1] DPT_Value_Temp	2 Bytes
126	Input 1 - Brightness encoder	Output	R-CT	[9.4] DPT_Value_Lux	2 Bytes
127	Input 1 - Scene	Output	R-CT	[18.1] DPT_SceneControl	1 Byte
128	Input 1 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
133	Input 2 - Extended	Input / Output	RWCT	[1.8] DPT_UpDown	1 Bit
134	Input 2 - Short	Output	R-CT	[1.8] DPT_UpDown	1 Bit
135	Input 2 - Switch	Input / Output	RWCT	[1.1] DPT_Switch	1 Bit
136	Input 2 - Relative dimming	Input / Output	RWCT	[3.7] DPT_Control_ Dimming	4 Bit
137	Input 2 - Encoder 8 Bit	Output	R-CT	[5] 5.xxx	1 Byte
138	Input 2 - Temperature encoder	Output	R-CT	[9.1] DPT_Value_Temp	2 Bytes
139	Input 2 - Brightness encoder	Output	R-CT	[9.4] DPT_Value_Lux	2 Bytes
140	Input 2 - Scene	Output	R-CT	[18.1] DPT_SceneControl	1 Byte
141	Input 2 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
154	Input 3 - Extended	Input / Output	RWCT	[1.8] DPT_UpDown	1 Bit
155	Input 3 - Short	Output	R-CT	[1.8] DPT_UpDown	1 Bit
156	Input 3 - Switch	Input / Output	RWCT	[1.1] DPT_Switch	1 Bit
157	Input 3 - Relative dimming	Input / Output	RWCT	[3.7] DPT_Control_ Dimming	4 Bit
158	Input 3 - Encoder 8 Bit	Output	R-CT	[5] 5.xxx	1 Byte
159	Input 3 - Temperature encoder	Output	R-CT	[9.1] DPT_Value_Temp	2 Bytes
160	Input 3 - Brightness encoder	Output	R-CT	[9.4] DPT_Value_Lux	2 Bytes
161	Input 3 - Scene	Output	R-CT	[18.1] DPT_SceneControl	1 Byte
162	Input 3 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit
163	Input 4 - Extended	Input / Output	RWCT	[1.8] DPT_UpDown	1 Bit
164	Input 4 - Short	Output	R-CT	[1.8] DPT_UpDown	1 Bit

No	Text	Function	Flags	DPT type	Size
165	Input 4 - Switch	Input / Output	RWCT	[1.1] DPT_Switch	1 Bit
166	Input 4 - Relative dimming	Input / Output	RWCT	[3.7] DPT_Control_ Dimming	4 Bit
167	Input 4 - Encoder 8 Bit	Output	R-CT	[5] 5.xxx	1 Byte
168	Input 4 - Temperature encoder	Output	R-CT	[9.1] DPT_Value_Temp	2 Bytes
169	Input 4 - Brightness encoder	Output	R-CT	[9.4] DPT_Value_Lux	2 Bytes
170	Input 4 - Scene	Output	R-CT	[18.1] DPT_SceneControl	1 Byte
171	Input 4 - Blocking object	Input	RWC-	[1.1] DPT_Switch	1 Bit

5. Parameter setting

The default settings of the parameter are labelled by an underscore.

5.1. General settings

First set the general parameters for the bus communication (telegram rate, transmission delay). Additionally, you can indicate if for the programming of scenes all, or only the changed settings are applied to the bus.

Maximum telegram rate	1 • 2 • 5 • 10 • 20 telegrams per second
Send delay of switching and status outputs after voltage returns	5 s ... 2 h
For the use of scenes:	
Application when programming	all parameters • only changed parameters

Parameter „Application when programming“

If "all parameters" is selected, the values stored in the device (e.g. positions) are overwritten by the ETS download in any case.

If "only changed parameters" is selected, only the parameters in the unit that have been changed *in the ETS* compared to the previous download will be overwritten.
In this way, for example, movement positions that were stored by the user with the memory function via the bus are retained. If a value is changed via the bus (not in the ETS!) then this value is retained with "only changed parameters".

5.2. Output

State here what is connected to the output channel.

Operating mode	
Channel A controls	• shutter • blind • awning • window • double switch function

Thereafter, the setting options will appear:

Settings for drives (channel A):

- General specifications for the connected drive
  (see *Channel settings – drives*, page 15)
- Control functions: Movement range limit, blocking, type of automatic
  (see *Control (drives)*, page 17)
- Automatic functions: Automatic can be specified externally or internally
  (see *Movement limits*, page 24 or *Automatic for windows (drives)*, page 31)
- Scenes: Movement positions (see *Scenes (drives)*, page 35)

- Button inputs: Configuration as actuator button, bust button or for zero position sensor (see *Button inputs (drives)*, page 35)

Settings for switch functions
(Channel is divided into two switches A1|A2):

- General specifications for the switch function
  (see *Channel settings – switch functions*, page 44)
- Connecting different communication objects
  (see *Connection (switch functions)*, page 45)
- On/Off switch delays or time switching
  (see *On/Off switch delays, time switching (switch functions)*, page 45)
- Block function(see *Blocking function (switch functions)*, page 46)
- Button input: Configuration as actuator button or bus button
  (see *Button input (switch functions)*, page 46)

5.2.1. Channel settings – drives

If a drive is connected to the output channel, set first the general specifications for the drive.

Driving direction:
Up/down, on/off or open/close can be exchanged.

Exchange UP/DOWN *(shutter, blinds)* Exchange RETRACT/EXTEND *(awning)* Exchange OPEN/CLOSE *(window)*	no • yes

Runtime:
The runtime between the end positions is the basis for moving into intermediate positions (e.g. for movement range limits and scenes). Enter the runtime numerically (in seconds).

Runtime DOWN in sec *(shutter, blinds)* Runtime EXTEND in sec *(awning)* Runtime OPEN in sec *(window)*	1 … 320; 60
Runtime UP in sec *(shutter, blinds)* Runtime RETRACT in sec *(awning)* Runtime CLOSE in sec *(window)*	1 … 320; 65

If a dead time is observed while starting the curtain, then this can be entered manually at this point or calculated automatically. Obey the manufacturer's instructions for the curtain.

Use dead times	• no • yes, enter by hand
during the position travel from closed position in 10 ms *(only for manual input)*	0 ... 600
for position movement from all other positions in 10 ms *(only for manual input)*	0 ... 600

for slat movement from closed position in 10 ms *(only for manual input)*	0 ... 600
for movement with change of direction in 10 ms *(only for manual input)*	0 ... 600
for slat movement from all other positions in 10 ms *(only for manual input)*	0 ... 600

Runtime zero position and step setting of slats:
(only for shutters)
Through the runtime in which the drive continues moving in the zero position (i.e. after reaching the top end position), different curtain lengths or assembly positions of the end position switch may be balanced. The shading of a facade is completely retracted by adjusting the zero position runtimes, and thus provides a better overall image.
Step time x step number determines the turning time of the slats.

Runtime zero position in 0.1 sec	0 ... 255
Step time in 10 ms	1 ... 100; 20
Step number slats	1 ... 255; 5

If the short time command for shutters (step command) is used only for slat adjustment, but not for positioning the curtain, the following parameter is set to "Yes". The parameter appears only for shutters.

Allow step commands only for slat adjustment	no • yes

Break time:
The required break times during a change of direction of the drive should be adjusted according to the specifications of the motor manufacturer.

Break time for a change of direction in 0.1 sec	5 ... 100; 10

Reference movement:
With the regular movement to the two end positions, the runtime and zero position are adjusted again. Therefore, it can be set here after how many movements before a positioning movement a reference movement will be performed. The reference movement is always in the direction of the secure position (retracting when shading, closing windows).

Perform a reference movement	no • yes

Perform a reference movement	**yes**
for more than movements before an auto positioning movement	1 ... 255; 10

Slat turning:
(only for shutters)
The slat turning should be adjusted according to the specifications of the motor manufacturer.

Turn slats	• never • only after positioning movement • after each movement

Status object and drive position:
The status and current position can be sent to the bus. By sending of 1, the status object indicates that the retracted or closed position has been exited and it is suitable for example for monitoring windows.
The exact drive position can be sent on the bus if required. The variable delay ensures that the bus is not blocked by too many data packets during a longer movement. The position can also be transmitted cyclically.

Use status object	no • yes
Use drive position feedback	no • yes
Position transmit delay after change in 0.1 s *(only for feedback)*	0...50; 10
Transmit drive position cyclically *(only for feedback)*	no • 5 s • 10 s • ... • 2 h

Scenes:
Here the scene menu is activated for this output channel.

Use scenes	no • yes

See *Scenes:*, page 17.

5.2.1.1. Control (drives)

Set the behaviour of the drive here.

Movement range limit:

The operating range limit is used in order to avoid that two units collide with each other (e.g. an awning and a window which is about to open).

One of two drive mechanisms is prioritised and is parameterised as master and the other one as slave. By means of zero position sensors, both actuators know the own current status and the current status of the other one. This one is either “in a safe position“ or “not in a safe position“. The safe position is reached as soon as the drive mechanism is in a sector where a collision is not possible (for an awning, for example, this might be an extension of 0 to 30%). In order to report the safe position of the drive mechanism, either a zero position sensor (e.g. final position switch or light barrier) may be connected at an input of the actuator (this must be set in the example, if the awning is used as slave 30% may be open, at position 31% it should be mounted), or the actuator receives the message of its zero position sensor by the bus (see graphic in chapter *Connection options for zero position sensors* in the general part).

Before the drive mechanism of the master actuator is moved, the slave actuator receives the command to move its drive mechanism to the safe position. As a consequence, the slave remains in safe position or it moves back if it is not within the safe range.

The master actuator knows from the communication object „Slave zero position status" whether the drive mechanism connected to the slave actuator is already in a safe position (then the master moves immediately) or not (then the master waits). Only if the master actuator is informed that the slave drive mechanism is in a safe position, it moves its drive mechanism beyond its own safe position.

Example:

The ventilation with the window shall take priority over the shading with the awning. Therefore, the window is parameterised as master, the awning as slave. Both are provided with a zero position sensor which reports whether the drive mechanism is in a safe position or not.

The awning is now extended and the window shall be opened. The window knows the status of the awning ("not safe position") and therefore submits a master command to the awning. This is the signal for the awning, to retract a little bit. As soon as the awning has reached a safe position, there is an according feedback signal of the zero position sensor of the awning. Only now the window opens.

Master and slave regularly exchange their positions ("safe" or "not safe"). By means of the monitoring period, you may adjust the frequency of information retrieval. The selected period should be shorter than the period which the monitored drive mechanism needs to travel from the limit of the safe range (last reported safe position) to a position where there is risk of collision.

If the drive mechanism does not receive a master/slave or zero position object, it moves to the safe position. The same holds true for a bus voltage breakdown or for a malfunction message from the zero position sensor (is valid for the parameterisation as master and as slave).

Without movement range limitation:

Use movement range limit	**no**
Behaviour on bus voltage restoration and after programming	• no action • Up command (or Retract/Close) • Down command (or Extend/Open)

With movement range limit:
Set if the zero position sensor of the drive is directly connected to the actuator (input channel) or if the zero position is received via the bus (communication object).

Use movement range limit	**yes**

Zero position sensor	• using communication object • using input channel • do not use
Actuator is	master • slave

Actuator as master:
For non-receipt of Slave status object move in secure position.

Actuator is	**master**
Send repetition for master command in sec	1 ... 255; 10
Monitoring period for slave status (and zero position) object in sec	1 ... 255; 10

Actuator as slave:
For non-receipt of Master status object move in secure position.

Actuator is	**slave**
Send repetition for slave commands in sec	1 ... 255; 10
Monitoring period for master status (and zero position) object in sec	1 ... 255; 10
Movement position for slave in % if input "Master zero position command" = 1	0 ... 100

Reference travel direction:

If the travel range is limited, the direction of the reference travel is fixed (safe position). The direction can be set without limiting the travel range.

Direction of reference travel	• in safe position • in closed position *(move out shading)* • in open position *(window)* • shortest route

Blocking objects:
The output channel can be blocked in case of rain, wind or other events. The manual operation is then not possible. Blocking and monitoring are configured here first. For setting the individual blocks, separate menu items "Blocking X" will appear (see chapter *Block – blocking objects*, page 21, *Block – wind blocking*, page 22 and *Block – rain blocking*, page 23).
The priorities of the blocking objects correspond to the sequence listed (Block 1 has the highest priority, Block 5 the lowest).

Use Block 1 (high priority)	• no • yes, with blocking object • yes, as wind blocking • yes, as rain blocking
Use block 2	• no • yes, with blocking object • yes, as wind blocking • yes, as rain blocking

Use block 3	• no • yes, with blocking object • yes, as wind blocking • yes, as rain blocking
Use block 4	• no • yes, with blocking object • yes, as wind blocking • yes, as rain blocking
Use Block 5 (low priority)	• no • yes, with blocking object • yes, as wind blocking • yes, as rain blocking
Priority is	• Block 5 over Manual • Manual over Block 5
Use monitoring of blocking objects	No • Yes
Monitoring period for blocking objects *(only if using monitoring of the blocking objects)*	5s... • 2 h; 5 min
Behaviour if a blocking object is not received *(only if blocking object monitoring is used)*	• Stop • Up command • Down command *(Shutters/roller blinds)* • Retract command • Extend command *(Awnings)* • Close command • Open command *(Windows)*

Use movement limit 1/2:

The movement limits are activated here, and can them be configured in their own menu items. See 'Movement limits' on Page 30.

Short time restriction (for blinds):

If short time restriction is active, only short time movement commands are still possible manually. If the function "Allow step commands only for blind adjustment" is activated simultaneously, (see *Channel settings – drives*, page 15) only the slats can still be adjusted by hand but no longer the movement position of the shutter.
Restriction is active for object value 1.

Use short time limit	no • yes
Value of the object in front of 1. communication and bus voltage restoration *(if short time restriction is used)*	0 • 1

Automatic reset:
With the manual operation the automatic of the drive is deactivated. Here it is set when the automatic is reactivated.

Manual switches to automatic after	• expiry of a waiting period • reception of an object • reception of an object or expiry of a waiting period
Waiting period in min *(if "Expiration of a waiting period" was chosen)*	1...255; 20
Switch to automatic for an object value *(if "Receipt of an object" was chosen)*	0 • 1 • 0 or 1

Automatic blocking object:
With the automatic blocking object, the automatic can be deactivated for a short term (e.g. if present or during speeches in conference rooms).
Here it is also specified in which mode the channel is found when the voltage returns, i.e. after a power failure. The mode (manual or automatic) is send as a status object to the bus.

Use automatic blocking object	no • yes
Automatic is blocked when *(if „yes“ was chosen)*	0 • 1
Value of the blocking object after resumption of power *(if „yes“ was chosen)*	0 • 1
Operating mode after power returns	• Automatic • Manual
Send status object	• 1 for automatic | 0 for manual • 0 for automatic | 1 for manual
Send delay of the status output Automatic or Manual in 0.1 sec	0...50

Type of automatic:
The automatic for the connected drive can be specified externally, however all the settings can also be configured internally. If "internal automatic" is chosen, a separate menu item "Automatic" (see chapter *Movement limits*, page 24 or *Automatic for windows (drives)*, page 31) appears.

Type of automatic	external automatic • internal automatic

Block – blocking objects

The menu item only appears if a block with blocking object was configured for "control". Here it is specified was happens for object value 1 and 0. Via the free blocking object, a fire alarm scenario may be configured for example (create escape routes by retracting the shading, smoke extraction via windows). This can prevent being locked

out on the patio (opened window contact of the patio door blocks the shutter in front of the door).

Designation	[Block 1 ... 5] Enter a designation here!
If blocking object has value =1	• no action • stop • move into position • up-command • down-command *(shutter/blind)* • retract-command • extend-command *(awning)* • close-command • open-command *(window)*
Position in % (only if by using a block, a specific position is achieved)	0...100
Slat position in % (only if by using a block, a specific shutter position is achieved)	0...100
If blocking object has value =0	
For manual operation before and after blocking	• no action • move into last position
For automatic operation after blocking	follow automatic
Value of the object before the 1st communication and bus voltage return	0...1

Block – wind blocking

The menu item only appears if a wind blocking was configured for "control". The input object "wind blocking" is linked with the output object of a wind sensor. The input can be a 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value).

Designation	[Wind blocking] Enter a designation here!
Type of input object	1 bit • 16 bit

1 bit input object:

Type of input object	**1 bit**
If blocking object has value =1	• no action • stop • move into position • up-command • down-command *(shutter/blind)* • retract-command • extend-command *(awning)* • close-command • open-command *(window)*

Position in % (only if by using a block, a specific position is achieved)	0...100
Slat position in % (only if by using a block, a specific shutter position is achieved)	0...100
Waiting period in secure position in min after blocking	1...255; 5
Behaviour after waiting period	
For manual operation before and after blocking	• no action • move into last position
For automatic operation after blocking	follow automatic

16 bit input object:

Type of input object	**16 bit**
As of wind speed in m/s blocking	2...30; 5
If blocking is active	• no action • stop • move into position • up-command • down-command *(shutter/blind)* • retract-command • extend-command *(awning)* • close-command • open-command *(window)*
Waiting period in secure position in min after blocking	1...255; 5
Behaviour after waiting period	
For manual operation before and after blocking	• no action • move into last position
For automatic operation after blocking	follow automatic
Send current blocking status	no • yes

Block – rain blocking

The menu item only appears if a rain blocking was configured for "control". The input object "rain blocking" is linked with the output object of a rain sensor.

Designation	[rain blocking] Enter a designation here!
If blocking object has value =1	• no action • stop • move into position • up-command • down-command *(shutter/blind)* • retract-command • extend-command *(awning)* • close-command • open-command *(window)*
Position in % (only if by using a block, a specific position is achieved)	0...100
Slat position in % (only if by using a block, a specific shutter position is achieved)	0...100
Waiting period in secure position in min after blocking	1...255; 5
Behaviour after waiting period	
For manual operation before and after blocking	• no action • move into last position
For automatic operation after blocking	follow automatic

Movement limits

The menu item appears only if a movement limit was activated in 'Control'. Movement limits can be used to restrict manual movement. The limit is active for object value 1

Limitation type	• full • movement position • slat angle (for shutters) • allow UP only • allow DOWN only
Value of the object in front of 1. Communication and bus voltage restoration	0 • 1

If limiting the movement position:

Limitation type	• movement position
Movement allowed in the position range	
from (in %)	0...100
to (in %)	0...100

If limiting the slat angle (shutters only):

Limitation type	• slat angle
Movement allowed in the angle range	
from (in %)	0...100
to (in %)	0...100

5.2.1.2. Manual

The actuator has two movement position memories for manual movement. These positions can be

- be called up via object 15 "Move to manual position memory", for example with a push-button.
- be retrieved by a specific telegram sequence. The command sequence of these telegrams can be sent, for example, with the down button of an appropriately configured bus push button (long button press followed immediately by a short button press). In this way, only one position can be recalled.

Use position memory	no • yes

Object 15 can be used to move to one or two (object value 0 and 1) positions.

Use different positions for object values 0 and 1	no • yes (if 'yes' is selected, there will be a division into positions for object value 0 and object value 1)

For shutters, both the movement and the slat position can be stored.

Position in %	0...100

If retrieval via the command sequence is activated, the position is retrieved as soon as a "1" is received via the long-term object and within 2 seconds a "1" is also received via the short-term object. If "Use different positions for object values 0 and 1" is selected, the position for object value 0 is retrieved.

Allow calling via command sequence: long-term = 1, short-term = 1	no • yes

The stored positions can be changed via the learning object. The learning object ensures that the current position is stored in the memory. All parameters (height and slat in the case of blinds) or only changed parameters can be adopted (see *Parameter „Application when programming“*, page 14).

Use learning object for new position	no • yes
Transfer when programming (when learning object is used)	• all parameters • changed parameters only

5.2.1.3. Automation - external

The 'External automation' menu item appears if the external automation is selected in 'Control'. In this case, the position memory can be activated for the automatic movement. The position set here can be overwritten via a learning object at any time. The memorised position can be retrieved again at a later time. For configuration options, see Chapter 'Manual' on Page 31.

5.2.1.4. Automatic - internal for shading (drives)

The menu item "Automatic internal" appears if internal automatic is selected for "control". The internal automatic functions take into account the brightness/position of the sun, outdoor and indoor temperature and allow a time and dimming control. A shading position can be specified or taught.

To be able to fully utilize the internal shading automatic, information about brightness/ twilight, outdoor and indoor temperature, time and position of the sun must be present in the bus system.

Outdoor temperature block:

The input object "outdoor temperature block" is linked with the output object of a temperature sensor. The input object can be a 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value).

Use external temperature shut-off	no • yes

Use external temperature shut-off	**yes**
Type of temperature input object	1 bit • 16 bit

1 bit input object:

Type of temperature input object	**1 bit**

Shading is allowed if the bit is 0 and blocked if the bit is 1.

16 bit input object:

Type of temperature input object	**16 bit**
Threshold value in 0.1°C	-300 … 800; 50
Hysteresis in 0.1°C	1 … 100; 20
Send current blocking status	no • yes

Shading is allowed
if the measurement value is larger than the threshold value+hysteresis
and blocked
if the measurement value is smaller than or equal to the threshold value.

Twilight/time control:

The time control is provided via a communication object. The input object "twilight control" is linked with the output object of a brightness sensor. A 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value) can be used for the twilight control.

Use twilight/time control	• no • only twilight control • only time control • both (OR linking)

Use twilight/time control	**only twilight control / both**
Type of twilight object	1 bit • 16 bit

16 bit input object:

Type of twilight object	**16 bit**
Twilight threshold value in lux	1 ... 1000; 10
Switching delay	1 minute
Send current twilight status	no • yes

Indoor temperature release:

The input object "indoor temperature release" is linked with the output object of a temperature sensor. The input object can be a 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value or target and actual value).

Use inside temperature release	no • yes

Type of input object	1 bit • 16 bit • 16 bit target/actual temperature

16 bit input object:

Type of input object	**16 bit**
Threshold value in 0.1°C	-300 ... 800; 200
Hysteresis in 0.1°C	1 ... 100; 20
Send current blocking status	no • yes

16 bit input object (target/actual temperature):
For this function the target value and actual value (measurement values) are imported from the 16bit object and evaluated.

Type of input object	**16 bit target/actual temperature**
Target value (SW) – actual value (MW) Difference in 0.1°C	1 ... 100; 20
Hysteresis in 0.1°C	1 ... 100; 20
Send current blocking status	no • yes

Shading is allowed if the measurement value is greater than or equal to the target value+difference
and blocked if the measurement value is smaller than the target value+hysteresis difference.

Automatic shading:

The automatic shading evaluates the input objects "brightness" and "position of the sun" of a weather station. The moving position for the automatic shading is specified here as well.

Use automatic shading	no • yes

Brightness:

For controlling brightness, a 1bit object (smaller or larger than a threshold value), as well as two or three 16bit objects (measurement values, e.g. East, South and West sun) can be used.

Type of shading input	1 x 1 bit • 1 x 16 bit • 2 x 16 bit • 3 x 16 bit

1 x 1 bit input object:
Set the delay times for shading (prevents constant opening and closing when light conditions change quickly).

Type of shading input	**1 x 1 bit**
Drive up delay in min	0 ... 255; 12
Departure delay in min	0 ... 30; 1

1 x 16 bit, 2 x 16 bit or 3 x 16 bit as an input object:
The brightness threshold value can be specified per parameter or communication object. For several brightness measurement values (2 x 16 bit or 3 x 16 bit) only the maximum brightness value is compared to the threshold value.

Type of shading input	**1 x 16 bit • 2 x 16 bit • 3 x 16 bit**
Shading threshold specification per	parameter • communication object

Threshold value per parameter:
Set the threshold value and delay times for shading (prevents constant opening and closing when light conditions change quickly).

Shading threshold specification per	**Parameter**
Shading threshold value in klux	0 ... 100; 30
Hysteresis threshold in	percent (%) • kLux
Hysteresis (in %)	0 ... 100; 10
Hysteresis (in kLux)	1 ... 120; 20
Drive up delay in min	0 ... 255; 12
Drive down delay in min	0 ... 30; 1
Send current shading status	No • Yes

Threshold value per communication object:
The threshold value is received via the communication object and can be changed additionally (e.g. button for "more sensitive" and "less sensitive"). Set the delay times for shading here (prevents constant opening and closing when light conditions change quickly).

Shading threshold specification per	**communication object**
The value communicated last shall be retained	• not • after voltage returns • after voltage returns and programming

Start threshold value in klux valid until 1st communication	0 ... 100; 30
Type of limit value change	• Absolute value with a 16bit comm. object • Lifting/lowering with a comm. object • Lifting/lowering with two comm. objects
Hysteresis threshold in	percent (%) • kLux
Hysteresis (in %)	0 ... 100; 10
Hysteresis (in kLux)	1 ... 120; 20
Increments in klux *(only when "lifting/lowering with comm. object")*	1 ... 5; 2
Drive up delay in min	0 ... 255; 12
Drive down delay in min	0 ... 30; 1
Send current shading status	no • yes

Position of the sun:

Assess position of the sun	no • yes

Assess position of the sun	**yes**
Position of the sun is defined via	• Discreet value of azimuth and elevation • Directions (regarding azimuth and elevation)

Defining position of sun via values:
Enter the range (direction and height) in which the sun must be located for the shading to be active.

Position of the sun is defined via	**discreet value** **of azimuth and elevation**
Azimuth from	0 ... 360
Azimuth to	0 ... 360
Elevation from	0 ... 90
Elevation to	0 ... 90

Defining position of the sun via directions:
Enter the direction in which the sun must be positioned so that the shading is active.

Position of the sun is defined via	**directions** **(regarding azimuth and elevation)**
Directions	• East (azimuth: 0° ... 180°) • South east (azimuth: 45° ... 225°) • South (azimuth: 90° ... 270°) • South west (azimuth: 135° ... 315°) • West (azimuth: 180° ... 360°)

Slats and moving position (for shutters):

For shutters the angle of the slats can be firmly set, or the slats can automatically follow the elevation. This rule applies: Slats are closed at 100%, horizontal at 50%.

Should the slats follow the elevation	no • yes

The slats should **not** follow the elevation (fixed reversing angle):
Adjust the desired position of the slats and the curtain.

Should the slats follow the elevation	**no**
Slat position in %	0 ... 100; 75
Shutter position in %	0 ... 100; 75
Use teaching object for new shading position *(curtain and slat positions will be saved, see info below)*	no • yes

The slats shall follow the elevation:
Three different elevation ranges can be set. A fixed curtain and slat position is specified for each.

Should the slats follow the elevation	**yes**
For an elevation less than (in degrees)	0 ... 90; 10
Slat position in %	0 ... 100; 95
For an elevation less than (in degrees)	0 ... 90; 30
Slat position in %	0 ... 100; 80
For an elevation less than (in degrees)	0 ... 90; 45
Slat position in %	0 ... 100; 65
For an elevation less than (in degrees)	90
Slat position in %	0 ... 100; 50
Shutter position in %	0 ... 100; 75
Use teaching object for new shading position *(only the curtain position will be saved, see info below)*	no • yes

Moving position (for awnings and blinds):

Awning position in % or blind position in %	0 ... 100; 75
Use teaching object for new shading position	no • yes

Use teaching object for new shading position: The curtain position it can be specified numerically or taught manually. For teaching set "use teaching object: Yes" and the "channel X shading position teaching object" is used for saving the position reached. Saving occurs for value = 1 and can for example be realized via a button linked to the teaching object. Numerical specifications already set are overwritten by the teaching object.

5.2.1.5. Automatic for windows (drives)

The menu item "Automatic" only appears if internal automatic is selected for "Control". Depending on the setting, the internal automatic functions take the outdoor temperature, indoor temperature and room air humidity into account, and allow forced ventilation via a communication object.

In order to fully utilize the internal ventilation automatic, information about the outdoor and indoor temperature and the inside air humidity must be present in the bus system.

Cold supply air lock:

The input object "cold supply air block" is linked with the output object of a temperature sensor. The input object can be a 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value).

Use cold supply air block	no • yes

Use cold supply air block	**yes**
Type of temperature input object	1 bit • 16 bit

1bit input object:

Type of temperature input object	**1 bit**

Ventilation is allowed if the bit is 0 and blocked if the bit is 1.

16bit input object:

Type of temperature input object	**16 bit**
Threshold value in 0.1°C	-300 … 800; 50
Hysteresis in 0.1°C	1 … 100; 20
Send current blocking status	no • yes

Ventilation is allowed if the measurement value is larger than the threshold value+hysteresis
and blocked if the measurement value is smaller than or equal to the threshold value.

Forced ventilation:

Use forced ventilation	no • yes

If forced ventilation is active ("use forced ventilation: Yes"), ventilation is started as soon as the communication object "forced ventilation" = 1.

Warm supply air block:

The input object "warm supply air block" is linked with the output object of one or more temperature sensors. The input object can be a 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value indoor/outdoor or target and actual value).

Use warm supply air block	no • yes

Use warm supply air block	**yes**
Type of input object	1 bit • 16 bit • 16 bit target/actual temperature

1bit input object:

Type of input object	**1 bit**

Ventilation is allowed if the bit is 0 and blocked if the bit is 1.

16bit input object:

Type of input object	**16 bit**
Threshold value in 0.1°C	-100 ... 200; 50
Hysteresis in 0.1°C	1 ... 100; 20
Send current blocking status	no • yes

Ventilation is allowed if the outdoor measurement value is smaller than the indoor measurement value+difference-hysteresis and blocked if the outdoor measurement value is greater than or equal to the indoor measurement value+difference.

16bit input object (target/actual temperature):
For this function the target value and actual value (measurement values) are imported from the 16bit object and evaluated.

Type of input object	**16 bit target/actual temperature**
Close if outdoor temperature exceeds the target value by (in 0.1°C)	0...255; 50
Hysteresis in 0.1°C	1...100; 20
Send current blocking status	no • yes

Ventilation is allowed if the outdoor measurement value is smaller than the target value+difference-hysteresis and blocked if the outdoor measurement value is greater than or equal to the target value+difference.

Open by temperature/humidity:

Open window	• never • if too high temperature • if too high room air humidity • if too high temperature or room air humidity

Indoor temperature:

These parameters appear if ventilated at "too high temperature" / "too high temperature or room air humidity". The input object can be a 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value or target and actual value).

Type of temperature input object	1 bit • 16 bit • 16 bit target/actual temperature

1 bit input object:

Type of temperature input object	**1 bit**

Ventilation is activated if the bit is 0 and blocked if the bit is 1.

16 bit input object:
The threshold value specification can be provided via a parameter or communication object.

Type of temperature input object	**16 bit**
Indoor temperature of threshold specification via	parameter • communication object

Threshold value per parameter:

Indoor temperature of threshold specification via	**parameter**
Indoor temperature threshold value in 0.1?	100 … 500; 300
Hysteresis in 0.1?	0 … 100; 20
Send current temperature status	no • yes

Threshold value per communication object:
The threshold value is received via the communication object and can be changed additionally (e.g. button for target temperature + and -).

Indoor temperature threshold specification via	**communication object**
The value communicated last shall be retained	• not • after voltage returns • after voltage returns and programming
Start threshold value in 0.1°C valid until 1st communication	100 … 500; 300
Type of limit value change	• Absolute value with a 16bit comm. object • Lifting/lowering with a comm. object • Lifting/lowering with two comm. objects
Increments *(only when "lifting/lowering with comm. object")*	0.1°C ... 5°C; 1°C
Hysteresis in 0.1?	0 … 100; 20
Send current temperature status	no • yes

16 bit input object (target/actual temperature):
For this function the target value and actual value (measurement values) are imported from the 16bit object and evaluated.

Type of temperature input object	**16 bit target / actual temperature**
Open if actual value exceeds the target value (in 0.1°C)	0...255; 20
Hysteresis in 0.1°C	0...100; 20
Send current blocking status	no • yes

Room air humidity:

These parameter appear if ventilated at "too high room air humidity" / "too high temperature or room air humidity". The input object can be a 1bit object (smaller or larger than a threshold value), as well as a 16bit object (measurement value).

Type of humidity input object	1 bit • 16 bit

1 bit input object:

Type of humidity input object	**1 bit**

Ventilation is activated if the bit is 0 and blocked if the bit is 1.

16 bit input object:

Type of humidity input object	**16 bit**
Indoor humidity threshold value in %	0 ... 100; 60
Hysteresis in 0.1°C	0 ... 100; 5
Send current humidity status	no • yes

Window opening:

If the ventilation by temperature or humidity is controlled via a 1bit input object, then enter the opening position in %.

Window opening in %	1...100

If the ventilation is controlled by temperature and humidity via a 16bit input object, then you can either set an opening position or open the windows incrementally. In the step operation the temperature/humidity deviation is checked after a specified period of time, and may be increased/decreased by one step.

Window opening	absolute in % • incrementally
Window opening in % *(only if "window opening is absolute in %")*	1...100
incrementally by (in %) *(only if "window opening is in increments")*	1...100; 25
every (in minutes) *(only if "window opening is in increments")*	1...60; 3

5.2.1.6. Scenes (drives)

A group address for scenes must be filed in the KNX system to control the scenes. The input object 'Channel X: call/save scenes' of the actuator is linked to this group address.

A scene is **called**, then the **scene number** is communicated to the actuator. The movement position saved for this scene number in the actuator is then taken.
If the scene **saving** function is used, then the current movement position is saved for this scene number in the actuator.

The 'Scenes' menu item of the actuator is only shown if 'Use scenes: yes' was selected in the settings for the drive channel. Every drive can **have up to 16 scene save points** for movement positions.

Activate a scene save point.

Use scene save point X	no • **yes**

Assign a scene number to the scene save point. Use this scene number to call/save the movement position stored in the actuator. Make sure that every scene number is used only once per drive channel.

Scene number	0...127

Set the movement position. If it is allowed to save scenes via the bus, this position only applies after the ETS download until the first manual saving. Afterwards, the new movement position saved in the actuator is used.

Shutter position in % or Blind position in % or Awning position in % or Window position in %	0...100; 50
Slat position in % *(only for shutters)*	0...100; 70

5.2.1.7. Button inputs (drives)

The inputs can be used as actuator button or bus button.

For connected drives the input 1 can be used *alternatively* for a zero position sensor.

Operating mode	
Use input 1	• no • as a bus button • as an actuator switch • as a zero position sensor
Use input 2 (3 / 4)	• no • as a bus button • as an actuator switch

Input as bus button

If an input is used as a free bus button, it will send a previously set value to the bus when activated. In the program file of the actuator different parameters are integrated for frequently needed bus functions. Thus, the inputs can easily be configured as a switch, drive control, dimmer for sending values and for the scene calls.

Bus function	• Switch • Selector switch • Shutter • Blind • Awning • Window • Dimmer • 8 bit encoder • Temperature encoder • Brightness encoder • Scenes

Input as switch:

If a button with switch function is assigned to the input, select the bus function "Switch" and specify which value is sent when pressing/releasing the button and when it will be sent.

Function	**Switch**
Command when pressing the button	• send 0 • send 1 • do not send telegram
Command when releasing the button	• send 0 • send 1 • do not send telegram
Send value	• no change • for change to 1 • for change to 0 • for change and cyclical • for change to 1 and cyclical • for change to 0 and cyclical
Cycle *(if sent cyclical)*	5 s • 10 s • 30 s • 1 min • 2 min • 5 min • 10 min • 20 min • 30 min • 45 min • 1 h • 1,5 h • 2 h

The input can be blocked using a blocking object. Set what is transmitted to the bus when (de)activating blocking.
For active blocking there is *no* cyclical transmission.

Use blocking object	No • Yes

Once when activating the blocking	• send 0 • send 1 • do not send telegram
Once when deactivating the blocking	• send 0 • send 1 • do not send telegram • send current state

Input as changeover switch:

If a button with switch function is assigned to the input, select the bus function "Changeover Switch" and specify if the button should switch when pressed/released.

Function	**Changeover Switch**
Command when pressing the button	• Switching • do not send telegram
Command when releasing the button	• Switching • do not send telegram

The input can be blocked using a blocking object. For active blocking there is *no bus communication*.

Use blocking object	No • Yes

Input to shutter, blinds, awning or window control:

If the input to the drive control is used via the bus, select the bus function "shutter", "awning", "blinds" or "window" and specify the button function and control mode.

Function	**Shutter / blinds / awning / window**	
Button function	Up • Down Up • Down • Up/ Down Retract • Extend • Retract/Extend Close • Open • Close/Open	(blind) (shutter) (awning) (window)
Control mode*	• Standard • Standard inverted • Comfort mode • Dead man's switch	

*A detailed description of the setting options for the individual control modi can be found in the general part of chapter *Control modi for drive control*, page 40.

The input can be blocked using a blocking object. For active blocking there is *no bus communication*.

Use blocking object	No • Yes

Input as dimmer:

If the input is used as a dimmer, select the bus function "Dimmer" and specify the button function, time interval (switching/dimming) and if requested, the repeat interval for a long button press.

Function	**Dimmer**
Button function	brighter • darker • brighter/darker
Time between switching and dimming (in 0.1 s)	1...50; 5
Repeat the dimm command	no • yes
Repeat the dimm command for a long button press *(if dimm command is repeated)*	every 0.1 s • every 2 sec; every 1 sec
Dim by *(if dimm command is repeated)*	1,50% • 3% • 6 % • 12,50% • 25% • 50%

The input can be blocked using a blocking object. For active blocking there is *no bus communication*.

Use blocking object	No • Yes

Input 8 bit encoder:

If the input is to be used as an 8bit encoder, select the "8 bit encoder" bus function and specify which value will be sent.

Function	**8 bit encoder**
Value	0...255

The input can be blocked using a blocking object. For active blocking there is *no bus communication*.

Use blocking object	No • Yes

Input as temperature encoder:

If the input is used as a temperature encoder, then choose the bus function "Temperature encoder" and specify which value between -30°C and +80°C will be sent. By sending a temperature value, the target value of the temperature control may be changed for example.

Function	**Temperature encoder**
Temperature in 0.1°C	-300...800; 200

The input can be blocked using a blocking object. For active blocking there is *no bus communication*.

Use blocking object	No • Yes

Input as brightness encoder:

If the input is assigned and shall be used as a brightness encoder (e.g. threshold value of a sun sensor), select "brightness encoder" and specify which value will be sent.

Function	**Brightness encoder**
Brightness in klux	0...100; 20

The input can be blocked using a blocking object. For active blocking there is *no bus communication*.

Use blocking object	No • Yes

Input for scene control:

If scenes are called and saved with the input, then choose the "Scenes" bus function and specify the saving, time difference (call/save) and scene number.

Function	**Scenes**
Button operation	• without saving • with saving
Time between calling and saving in 0.1 seconds *(only if selected "with saving")*	1...50; 20
Scene No.	0...127

The input can be blocked using a blocking object. For active blocking there is *no bus communication*.

Use blocking object	No • Yes

Input as actuator button

If this channel is used for the input to the control of the drive, then specify the button function and the control mode.

Button function	Up • Down Up • Down • Up/Down Retract • Extend • Retract/Extend Close • Open • Close/Open	(blind) (shutter) (awning) (window)
Control mode*	• Standard • Standard inverted • Comfort mode • Dead man's switch	

*A detailed description of the setting options for the individual control modi can be found in the chapter *Control modi for drive control*, page 40.

The input can be blocked using a blocking object. No operation is possible for an active block.

Use blocking object	No • Yes

If monitoring periods or movement range limits are used, no operation via the local button is possible in case of a bus voltage failure.

Input as zero position sensor

The zero position sensor is used for the movement range limit of the respective drive (see *Channel settings – drives*, page 15). In case of a defect zero position sensor a malfunctioning message can be sent to the bus.

Send malfunction message when zero position sensor is defective	No • Yes

Control modi for drive control

If inputs are used as buttons for operating shading or windows, then different control modi can be set.

Control mode	• Standard • Standard inverted • Comfort mode • Dead man's switch

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	

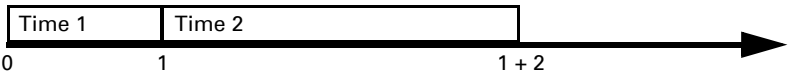
Time between short and long in 0.1 seconds	1...50; 10
Repeat the step command for a long button press	every 0.1 s • every 2 sec; every 0.5 sec

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. 1
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:

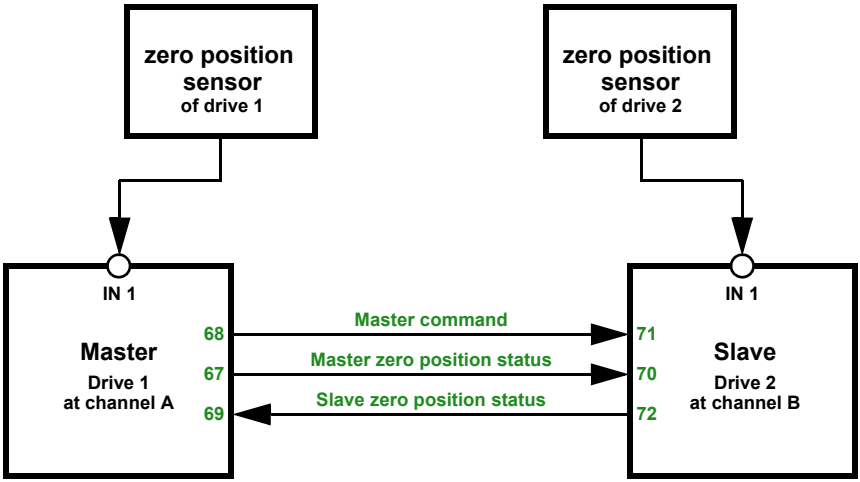
The drive moves as soon as the button is actuated and stops as soon as the button is released.

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

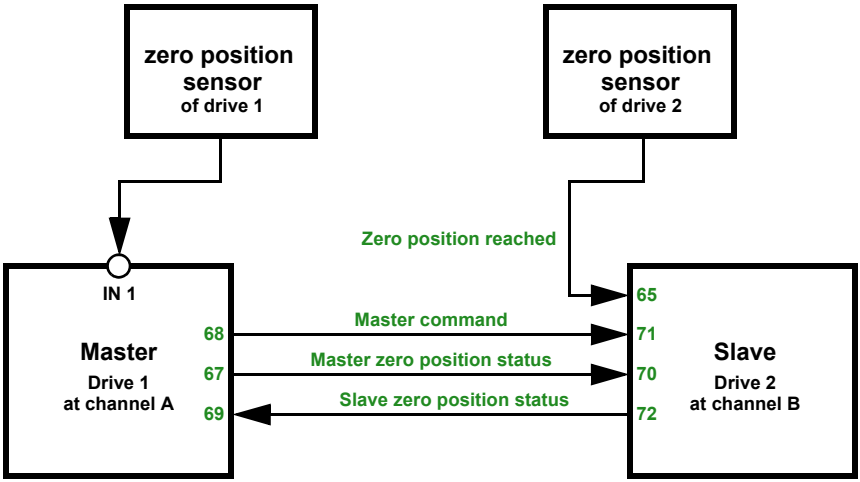
5.2.2. Connection option for zero position sensors

See also section *Movement Range Limit* in chapter *Control (drives)*, page 17. The examples and the communication object numbers refer to the mutual master-slave coupling of drives at the output channel A and channel B.

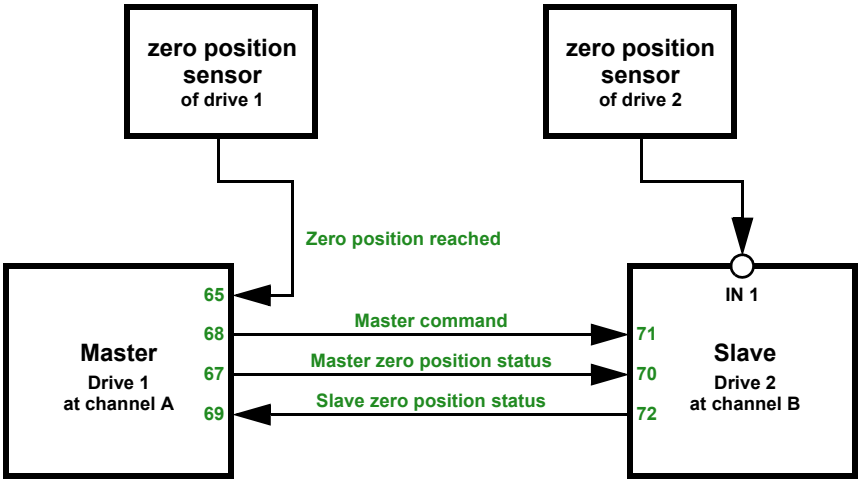
Actuator A is Master, zero position sensor at input 1 of the actuator, Actuator B is Slave, zero position sensor at input 1 of the actuator:



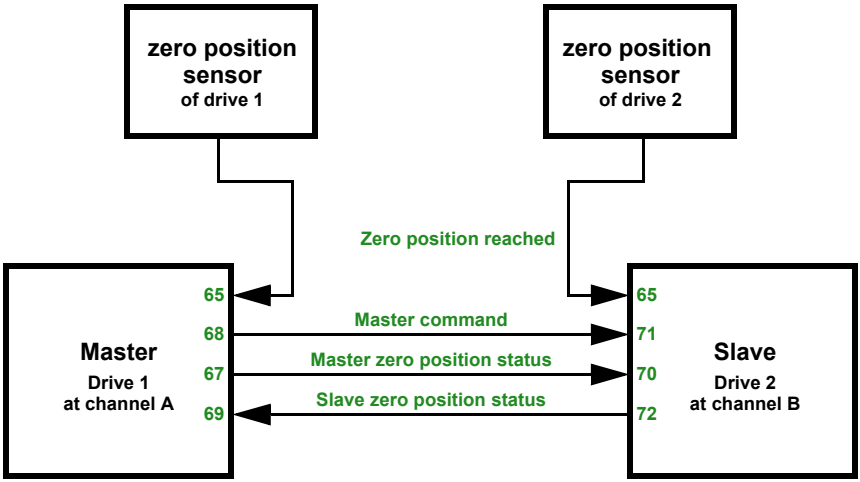
Actuator A is Master, zero position sensor at input 1 of the actuator, Actuator B is Slave, zero position sensor via bus:



Actuator A is Master, zero position sensor via bus, Actuator B is Slave, zero position sensor at input 1 of the actuator:



Drive channel A is Master, zero position sensor via bus, drive channel B is Slave, zero position sensor via bus:



5.2.3. Channel settings – switch functions

If two switchable devices are connected to the output channel, two separate channels will appear (e.g. "Channel A1 – switch function" and "Channel A2 – switch function"). First set the general specifications for the connected device and, if necessary, activate the connections, time functions and blocking objects. A diagram is found in chapter *Channel settings – switch functions*, page 44.

Relay operation	closer • opener
Behavior after reset and ETS download	• opened • closed
Use status object	• no • as an active feedback object • as a passive status object
Use connection function (see *Connection (switch functions)*, page 45)	no • yes
Use time function (see *On/Off switch delays, time switching (switch functions)*, page 45)	• no • as a switch on delay • as a switch off delay • as a switch on and off delay • as a staircase light timer
Use blocking object	no • yes
Use scenes	no • yes

5.2.3.1. Connection (switch functions)

The menu item "connection" appears only for the settings for the switch function channel if selected "Use switch functions: Yes".

In the connection object ("Channel X connection") different communication objects can be linked with AND or OR. E.g. a light can only be switched on if the button input is active AND twilight is active.

Connection type	AND • OR
Value of the connection object after bus voltage returns	0 • 1

5.2.3.2. On/Off switch delays, time switching (switch functions)

The menu item appears only for the settings for the switch function channel if a time function is chosen. The menu item has the same name as the selected function.

With the switch on and off delay, a switch can be used for example for a HVAC unit and light. Through the switch on delay the ventilator will only start if the light has already been turned on for a few minutes. The switch off delay effects that the ventilator will follow up if the button was operated again and the light is already off.

The staircase timer function makes sure for example that the light is on for a defined period of time and then turns off automatically.

Switch on delay

The switch on delay is set with a time basis and time factor (e.g. 1 min × 4 corresponds to 4 minutes). Additionally it is specified if the time interval for a repeat receipt of a switch-on telegram is extended ('triggered again", e.g. by pressing the button again) and what happens when a switch off telegram arrives from the bus.

Time basis	0.1 s • 1 s • 1 min • 1 h
Time factor	4...255; 10
Switch on delay cannot	be triggered again • can be triggered again
Off telegram during switch on-delay effects	nothing • direct turn off

Switch off delay

The switch off delay is set with a time basis and time factor (e.g. 1 min × 4 corresponds to 4 minutes). Additionally it is specified if the time interval for a repeat receipt of a switch-off telegram is extended ("can be triggered again", e.g. by pressing the button again) and what happens when a switch off telegram arrives from the bus.

Time basis	0.1 s • 1 s • 1 min • 1 h
Time factor	4...255; 10
Switch off delay cannot	be triggered again • can be triggered again
On telegram during switch off-delay effects	nothing • direct turn on

Staircase lighting timer

The staircase time switch sets with a time basis and time factor how long the light will remain on (e.g. 1 s × 10 corresponds to 10 seconds). Additionally it is specified if the time interval for a repeat receipt of a switch-on telegram is extended ("triggered again", e.g. by pressing the button again) and what happens when a switch off telegram arrives from the bus.

Time basis	0.1 s • 1 s • 1 min • 1 h
Time factor	4...255; 10
Staircase light time can	not be triggered again • can be triggered again
Off telegram during staircase light period affects	nothing • direct turn off

5.2.3.3. Blocking function (switch functions)

The menu item "blocking function" appears only for the settings for the switch function channel if selected "Use blocking functions: Yes".

The output channel can be blocked by a block telegram. What happens during the blocking, for bus voltage return and after the blocking is set here. The manual operation is then not possible for an active block.
The function can be used for example for a light, which is turned on when pressing a "panic button" (=trigger for blocking function) and cannot be turned off any longer.

Blocking function blocks for	0 • 1
Value of the blocking object after reset and ETS download	0 • 1
Response when blocking	no change • opened • closed
Response upon release	follows switch command • opened • closed

5.2.3.4. Button input (switch functions)

The inputs can be used as actuator button or bus button.

Operating mode	
Use input 1 / 2 / 3 / 4	• no • as a bus button • as an actuator switch

Input as bus button

See *Input as bus button*, page 36.

Input as actuator button

If the input to the control of the device is used at this channel, then specify the button function.

Button function	Switch• Selector switch

If a button with switch function is assigned to the input, select the button function "Switch" and specify what happens when pressing/releasing the button and when to send.

Button function	**Switch**
Command when pressing the button	• switch on • switch off • nothing
Command when releasing the button	• switch on • switch off • nothing

The input can be blocked using a blocking object. Set what happens when (de)activating the block. No operation is possible for an active block.

Use blocking object	No • Yes

Use blocking object	**Yes**
Once when activating the blocking	• switch on • switch off • nothing
Once when deactivating the blocking	• switch on • switch off • nothing • evaluate current state

If a button with selector switch function is assigned to the input, select the bus function "Selector switch" and specify what happens when pressing and releasing the button.

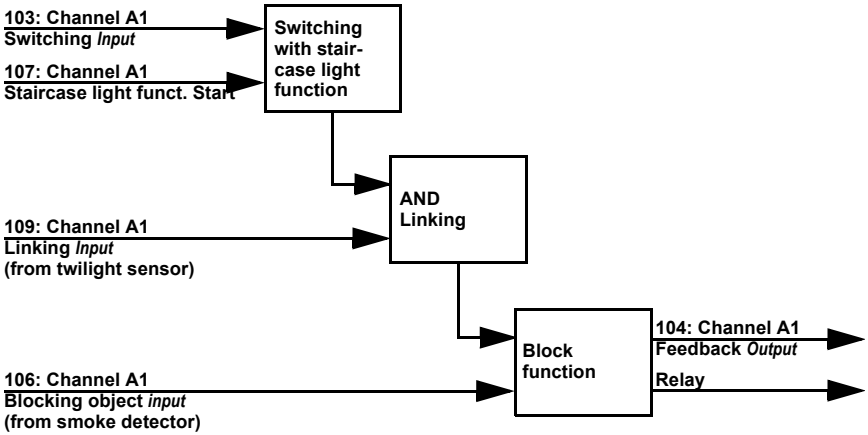
Button function	**Selector switch**
Command when pressing the button	• switch over • nothing
Command when releasing the button	• switch over • nothing

The input can be blocked using a blocking object. No operation is possible for an active block.

Use blocking object	No • Yes

Correlation connection – time switch – block

Application 1: Staircase light at channel A1, that can only be switchable at twilight/ night (linking) and that is turned on during a fire alarm (blocking).



When switching via communication object "Channel A1 switch" (103), the light is turned on or off normally. When switching via object "Channel A1 staircase light function start" (107), the staircase light time function is activated. The time function has priority, i.e. the status triggered by normal switching is overwritten.

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 Elektronik GmbH Control and Automation Engineering

Sohlengrund 16
75395 Ostelsheim
Germany

Phone +49 (0) 70 33 / 30 945-0 info@elsner-elektronik.de
Fax +49 (0) 70 33 / 30 945-20 www.elsner-elektronik.de