KNX manual
CO_2 sensor AMUN 716 S

Set basic KNX Multi
(Basic device for Multisensor KNX)

Amun 716 S 7169230

Set basic KNX Multi 9070900

Contents

1 Function description

- CO2 room air sensor with integrated individual room thermostat.
- Measures CO2 concentration, relative humidity, temperature, and barometric air pressure.
- Three independent, configurable thresholds for CO2 concentration and relative humidity.
- For controlling heating actuators or motorised actuators
- Two front panels are included in the scope of supply: an absolute and a relative scale. [1]
- Buttons for presence or operating modes: comfort, standby, temperature reduction at night, frost protection. [2]
- 4 binary inputs for conventional switches/push buttons (switching, dimming, blinds), also for external temperature sensor, window contact or presence signal.
- Actions can be carried out if thresholds are exceeded or fallen below (send, priority, switching, value).
- Display of current operating mode and heating/cooling by multi-coloured LED. [3]
- CO2 setting range of 500 - 2250 ppm (thresholds).
- Relative humidity of 1% to 100%.
- Basic module for multi-sensor KNX: For a complete multi-sensor KNX, thePrema KNX is always additionally required [4]
- Comparator for internal actuating values or external values.
- CO_2 calibration possible via object.

[1] Only Amun 716 S
[2] Only Amun 716 S
[3] Only Amun 716 S
[4] Only Set basic KNX Multi

2 Operation

The device has the following operating and display elements: [5]

1. One LED for display of the CO_2 content.
2. One rotary knob for the base setpoint of the room temperature controller, or for set point offset.
3. One LED for display of the relative humidity.
4. One button for the selection of the operating mode, or for presence.
5. One four-coloured LED for display of the operating mode.
   Red: comfort, **yellow:** standby, **green:** eco, **blue:** frost.
6. One LED for display of heating and cooling mode:
   Red: heating, **orange:** comfort extension, **blue:** cooling.

[5] Only Amun 716 S

3 Technical data

Operating voltage KNX	Bus voltage, IBus ≤ 15 mA
Type of connection	Bus connection: KNX bus terminal
Installation type	Wall-mounted [6] / Ceiling installation [7]
Display [8]	LEDs multi-coloured
Interface extension	max. 30 m
Ambient temperature	+5 °C ... +40 °C
Measurement range CO2	300 – 5000 ppm
Accuracy	0-1000 +/- (50 ppm + 3%) of the measured value 1001-2000 +/- (50 ppm + 5%) of the measured value >2000 +/- (100 ppm + 5%) of the measured value
Measurement range humidity	1 – 100 % rh (± 3 %)
Measurement range air pressure	30000 – 110000 Pa (± 100 Pa)
Measurement range temperature	-5 °C ... +45 °C (± 0,8 K)
Setting range temperature	+5 °C ... +32 °C
Number of external inputs	4
Contact voltage	5 V, provided internally
Contact current	0.5 mA / 5 mA (peak)
Protection rating	IP 20
Protection class	III in accordance with EN 60 730-1

[6] Only Amun 716 S
[7] Only Set basic KNX Multi
[8] Only Amun 716 S

4 The AMUN 716 S / Set basic KNX Multi application programme

4.1 Selection in the product database

Manufacturer	Theben AG
Product family	Heating, ventilation, air conditioning
Product type	Room air sensor with controller
Program name	AMUN 716 S

Number of communication objects	90 [9] / 89 [10]
Number of group addresses	255
Number of associations	255

The ETS database can be found on our website: www.theben.de/en/downloads_en

[9] Only Amun 716 S
[10] Only Set basic KNX Multi

4.2 Overview of communication objects

4.2.1 General

No.	Object name	Function	Length	R	W	C	T	DPT
1	*CO_2 value*	*Send*	2 bytes	R	-	C	T	9.008
2	*Relative humidity*	*Send*	2 bytes	R	-	C	T	9.007
3	*Temperature value*	*Send*	2 bytes	R	-	C	T	9.001
4	*Air pressure*	*Send*	4 bytes	R	-	C	T	14.058
5	*Degree of comfort*	*Send*	1 byte	R	-	C	T	5.001
6	*Device LEDs* [11]	*Reduced*	1 bit	-	W	C	-	1.001
		Brightness	1 byte	-	W	C	-	5.001
7	*CO_2 fresh air calibration*	*Release*	1 bit	-	W	C	-	1.001

[11] Only Amun 716 S

4.2.2 CO2 sensor

No.	Object name	Function	Length	R	W	C	T	DPT
8	*CO_2 threshold 1*	*Switching*	1 bit	-	-	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		0-100%	1 byte	-	-	C	T	5.001
		0-255	1 byte	-	-	C	T	5.010
9	*CO_2 threshold 1*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
10	*CO_2 threshold 1*	*Setting/reading threshold*	2 bytes	R	W	C	T	9.008
11	*CO_2 threshold 2*	*Switching*	1 bit	-	-	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		0-100%	1 byte	-	-	C	T	5.001
		0-255	1 byte	-	-	C	T	5.010
12	*CO_2 threshold 2*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
13	*CO_2 threshold 2*	*Setting/reading threshold*	2 bytes	R	W	C	T	9.008
14	*CO2 threshold 3*	*Switching*	1 bit	-	-	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		0-100%	1 byte	-	-	C	T	5.001
		0-255	1 byte	-	-	C	T	5.010
15	*CO_2 threshold 3*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
16	*CO2 threshold 3*	*Setting/reading threshold*	2 bytes	R	W	C	T	9.008
17	*Ventilating CO_2*	*Actuating value 0-100%*	1 byte	-	-	C	T	5.001
		Actuating value 0-255	1 byte	-	-	C	T	5.010
18	*Ventilating CO_2*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
19	*Ventilating CO_2*	*Setting/reading CO2 setpoint*	2 bytes	R	W	C	T	9.008
20	*CO_2 scenes*	*Send*	1 byte	-	-	C	T	17.001
21	*CO_2 scenes*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003

4.2.3 Humidity sensor

No.	Object name	Function	Length	R	W	C	T	DPT
22	*Humidity threshold 1*	*Switching*	1 bit	-	-	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		0-100%	1 byte	-	-	C	T	5.001
		0-255	1 byte	-	-	C	T	5.010
23	*Humidity threshold 1*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
24	*Humidity threshold 1*	*Setting/reading threshold*	2 bytes	R	W	C	T	9.007
25	*Humidity threshold 2*	*Switching*	1 bit	-	-	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		0-100%	1 byte	-	-	C	T	5.001
		0-255	1 byte	-	-	C	T	5.010
26	*Humidity threshold 2*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
27	*Humidity threshold 2*	*Setting/reading threshold*	2 bytes	R	W	C	T	9.007
28	*Humidity threshold 3*	*Switching*	1 bit	-	-	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		0-100%	1 byte	-	-	C	T	5.001
		0-255	1 byte	-	-	C	T	5.010
29	*Humidity threshold 3*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
30	*Humidity threshold 3*	*Setting/reading threshold*	2 bytes	R	W	C	T	9.007
31	*Ventilating humidity*	*Actuating value 0-100%*	1 byte	-	-	C	T	5.001
		Actuating value 0-255	1 byte	-	-	C	T	5.010
32	*Ventilating humidity*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
33	*Humidity scenes*	*Send*	1 byte	-	-	C	T	17.001
34	*Humidity scenes*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
35	*Dew point alarm*	*Send*	1 bit	R	-	C	T	1.005

4.2.4 Room temperature controller (RTC)

No.	Object name	Function	Length	R	W	C	T	DPT
40	*Base setpoint*	*Defining the set point temperature*	2 bytes	-	W	C	-	9.001
	Base setpoint at rotary control [12]	*Send*	2 bytes	R	-	C	T	9.001
41	*Manual set point offset*	*Receive*	2 bytes	-	W	C	-	9.002
	Set point offset at rotary control [13]	*Send*	2 bytes	R	-	C	T	9.002
42	*Outdoor temperature compensation*	*Send*	2 bytes	R	-	C	T	9.001
		Adjust setpoint	2 bytes	-	W	C	-	9.002
43	*Operating mode preset*	*Receive*	1 byte	-	W	C	-	20.102
	Night <-> standby	*Receive*	1 bit	-	W	C	-	1.001
44	*Comfort*	*Receive*	1 bit	-	W	C	-	1.003
	Presence	*Receive*	1 bit	-	W	C	-	1.018
45	*Window status*	*Closed=0, open=1*	1 bit	-	W	C	-	1.019
	Frost	*Receive*	1 bit	-	W	C	-	1.003
46	*Current operating mode*	*Send*	1 byte	R	-	C	T	20.102
47	*Operating mode as scene*	*Save/ call up*	1 byte	-	W	C	T	18.001
48	*Heating actuating value*	*Send*	1 bit	-	-	C	T	1.001
		Send	1 byte	-	-	C	T	5.001
48	*Heating/cooling actuating value*	*Send*	1 bit	-	-	C	T	1.001
		Send	1 byte	-	-	C	T	5.001
49	*PWM heating additional stage*	*Send*	1 bit	-	-	C	T	1.001
	Actuating value additional heating stage	*Send*	1 byte	-	-	C	T	5.001
50	*Cooling actuating value*	*Send*	1 bit	-	-	C	T	1.001
		Send	1 byte	-	-	C	T	5.001
51	*PWM cooling additional stage*	*Send*	1 bit	-	-	C	T	1.001
	Actuating value additional cooling stage	*Send*	1 byte	-	-	C	T	5.001
52	*Send heating mode/cooling mode*	*0 = heating, 1 = cooling*	1 bit	R	-	C	T	1.001
		0 = cooling, 1 = heating	1 bit	R	-	C	T	1.100
	Switching between heating and cooling	*0 = heating, 1 = cooling*	1 bit	-	W	C	-	1.001
		0 = cooling, 1 = heating	1 bit	-	W	C	-	1.100
53	*Current setpoint*	*Setting/sending*	2 bytes	-	W	C	T	9.001
54	*Control actual value*	*Send*	2 bytes	R	-	C	T	9.001
55	*External actual value*	*Receive*	2 bytes	-	W	C	-	9.001

[12] Only Amun 716 S
[13] Only Amun 716 S

No.	Object name	Function	Length	R	W	C	T	DPT
56	*Actual value failure*	*Send*	1 bit	R	-	C	T	1.001
57	*Outdoor temperature*	*Receive*	2 bytes	-	W	C	-	9.001
58	*Dew point alarm*	*Receive*	1 bit	-	W	C	-	1.005

4.2.5 External inputs I1-I4: Switch function

No.	Object name	Function	Length	R	W	C	T	DPT
60	*Channel I1.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
61	*Channel I1.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
62	*Channel I1.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
64	*Channel I1*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
65	*Channel I2.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
66	*Channel I2.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
67	*Channel I2.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
67	*Channel I2.3*	*Send value*	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx

No.	Object name	Function	Length	R	W	C	T	DPT
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
69	*Channel I2*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
70	*Channel I3.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
71	*Channel I3.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
72	*Channel I3.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
74	*Channel I3*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
75	*Channel I4.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
76	*Channel I4.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
77	*Channel I4.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001

No.	Object name	Function	Length	R	W	C	T	DPT
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
79	*Channel I4*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003

4.2.6 External inputs I1-I4: Button function

No.	Object name	Function	Length	R	W	C	T	DPT
60	*Channel I1.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
61	*Channel I1.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
62	*Channel I1.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
64	*Channel I1*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
65	*Channel I2.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
66	*Channel I2.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
67	*Channel I2.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
67	*Channel I2.3*	*Send value*	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx

No.	Object name	Function	Length	R	W	C	T	DPT
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
69	*Channel I2*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
70	*Channel I3.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
71	*Channel I3.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
72	*Channel I3.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
74	*Channel I3*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
75	*Channel I4.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
76	*Channel I4.2*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
77	*Channel I4.3*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001

No.	Object name	Function	Length	R	W	C	T	DPT
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
79	*Channel I4*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003

4.2.7 External inputs I1-I4: Dimming function

No.	Object name	Function	Length	R	W	C	T	DPT
60	*Channel I1*	*Switching*	1 bit	-	W	C	T	1.001
61	*Channel I1*	*Brighter/darker*	4 bit	-	-	C	T	3.007
62	*Channel I1.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
64	*Channel I1*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
65	*Channel I2*	*Switching*	1 bit	-	W	C	T	1.001
		Switching	1 bit	-	-	C	T	1.001
66	*Channel I2*	*Brighter/darker*	4 bit	-	-	C	T	3.007
67	*Channel I2.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
69	*Channel I2*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
70	*Channel I3*	*Switching*	1 bit	-	W	C	T	1.001
		Switching	1 bit	-	-	C	T	1.001
71	*Channel I3*	*Brighter/darker*	4 bit	-	-	C	T	3.007
72	*Channel I3.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
74	*Channel I3*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
75	*Channel I4*	*Switching*	1 bit	-	W	C	T	1.001
		Switching	1 bit	-	-	C	T	1.001
76	*Channel I4*	*Brighter/darker*	4 bit	-	-	C	T	3.007
77	*Channel I4.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001

No.	Object name	Function	Length	R	W	C	T	DPT
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
79	*Channel I4*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003

4.2.8 External inputs I1-I4: Blinds function

No.	Object name	Function	Length	R	W	C	T	DPT
60	*Channel I1*	*Step/stop*	1 bit	-	-	C	T	1.010
61	*Channel I1*	*UP/DOWN*	1 bit	-	W	C	T	1.008
		UP	1 bit	-	-	C	T	1.008
		DOWN	1 bit	-	-	C	T	1.008
62	*Channel I1.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Height %	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
63	*Channel I1.2*	*Slat %*	1 byte	-	-	C	T	5.001
64	*Channel I1*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
65	*Channel I2*	*Step/stop*	1 bit	-	-	C	T	1.010
66	*Channel I2*	*UP/DOWN*	1 bit	-	W	C	T	1.008
		UP	1 bit	-	-	C	T	1.008
		DOWN	1 bit	-	-	C	T	1.008
67	*Channel I2.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Height %	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
68	*Channel I2.2*	*Slat %*	1 byte	-	-	C	T	5.001
69	*Channel I2*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
70	*Channel I3*	*Step/stop*	1 bit	-	-	C	T	1.010
71	*Channel I3*	*UP*	1 bit	-	-	C	T	1.008
		UP/DOWN	1 bit	-	W	C	T	1.008
		DOWN	1 bit	-	-	C	T	1.008
72	*Channel I3.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Height %	1 byte	-	-	C	T	5.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
72	*Channel I3.1*	*2 byte 9.x*	2 bytes	-	-	C	T	9.xxx

No.	Object name	Function	Length	R	W	C	T	DPT
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
73	*Channel I3.2*	*Slat %*	1 byte	-	-	C	T	5.001
74	*Channel I3*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003
75	*Channel I4*	*Step/stop*	1 bit	-	-	C	T	1.010
76	*Channel I4*	*UP*	1 bit	-	-	C	T	1.008
		UP/DOWN	1 bit	-	W	C	T	1.008
		DOWN	1 bit	-	-	C	T	1.008
77	*Channel I4.1*	*Switching*	1 bit	-	W	C	T	1.001
		Priority	2 bit	-	-	C	T	2.001
		Send percentage value	1 byte	-	-	C	T	5.001
		Height %	1 byte	-	-	C	T	5.001
		Send value	1 byte	-	-	C	T	5.010
		2 byte 9.x	2 bytes	-	-	C	T	9.xxx
		4 byte 14.x	4 bytes	-	-	C	T	14.xxx
78	*Channel I4.2*	*Slat %*	1 byte	-	-	C	T	5.001
79	*Channel I4*	*Block = 1*	1 bit	-	W	C	-	1.001
		Block = 0	1 bit	-	W	C	-	1.003

4.2.9 External inputs I3-I4: Temperature sensor function

No.	Object name	Function	Length	R	W	C	T	DPT
70	*Channel I3.1*	*Temperature actual value*	2 bytes	R	-	C	T	9.001
75	*Channel I4.1*	*Temperature actual value*	2 bytes	R	-	C	T	9.001

4.2.10 Diagnosis and alarm objects

No.	Object name	Function	Length	R	W	C	T	DPT
80	*Firmware*	*Version*	2 bytes	R	-	C	T	217.001
81	*Alarm*	*Info*	6 bytes	R	-	C	T	219.001
83	*Alarm*	*Error text*	14 bytes	R	-	C	T	16.000

4.2.11 Measurement value calibration CO_2

No.	Object name	Function	Length	R	W	C	T	DPT
84	*CO_2 offset*	*Measurement value offset*	2 bytes	-	W	C	-	9.* [14]
85	*CO_2 reference*	*Receive measurement value*	2 bytes	-	W	C	-	9.008

4.2.12 Comparator

No.	Object name	Function	Length	R	W	C	T	DPT
86	*Comparator*	*Input 1*	1 byte	-	W	C	-	5.001
87	*Comparator*	*Input 2*	1 byte	-	W	C	-	5.001
88	*Comparator*	*Input 3*	1 byte	-	W	C	-	5.001
89	*Comparator*	*Output*	1 byte	R	-	C	T	5.001

[14] DPT9.* 2 byte floating value.

4.3 Description of communication objects

4.3.1 General objects

Object 1: CO_2 value
Sends the measured CO_2 content.

Object 2: Relative humidity
Sends the measured relative humidity and percent.

Object 3: Temperature value
Sends the room temperature in °C, measured with the temperature sensor inside the device.

Object 4: Air pressure
Sends the measured air pressure of the atmosphere in Pascal (Pa).
In meteorology, the air pressure is usually stated in Hectopascal (hPa).
1 Hectopascal (hPa) = 100 Pa

Note: 1 hPa corresponds to 1 mbar.

Object 5: Degree of comfort
The degree of comfort comprises 3 states, which are reported as a percentage value in fixed increments (see in the Appendix: *Comfort*)

Status	Value
Comfortable	100%
Still comfortable	50%
Uncomfortable	10%

Object 6: Device LEDs [15]
The brightness of the device LEDs can be changed via bus, if desired.
(See ***General*** parameter page).
Depending on the parameter setting (*object type*), 2 formats are possible.

Object type	Format	Object function
Via switch object	1 bit	Activate preset reduced brightness.
Via percentage value	1 byte	Set LED brightness individually via bus telegram.

[15] Only Amun 716 S

Object 7: Fresh air calibration
For initialising a calibration procedure of the device
(see in the Appendix: *Fresh air calibration*).

If the device is properly used, a calibration procedure is not necessary.

A wrong calibration can cause malfunctions of the device and the connected systems.
Please use the calibration only if absolutely necessary.

4.3.2 Objects for the CO2 sensor

Object 8: Threshold 1 CO_2
Initial object for the first CO_2 threshold.
Depending on the parameter setting, a telegram can be sent when exceeding or falling below the set threshold.
The format can be configured as a switch, priority, percentage, or value telegram (see above, *Overview of communication objects, CO_2 sensor*).

Object 9: Block CO_2 threshold 1
Block object for the threshold.
The acting direction (block with 1 or with 0) can be configured.

Object 10: Threshold 1 CO_2 – setting/reading threshold
Allows a subsequent change of the threshold via bus telegram at any time.

Object 11: CO_2 threshold 2
Output object for the second CO_2 threshold.
Depending on the parameter setting, a telegram can be sent when exceeding or falling below the set threshold.
The format can be configured as a switch, priority, percentage, or value telegram (see above, *Overview of communication objects, CO_2 sensor*).

Object 12: Block CO_2 threshold 2
Block object for the threshold.
The acting direction (block with 1 or with 0) can be configured.

Object 13: CO_2 threshold 2 – setting/reading threshold
Allows a subsequent change of the threshold via bus telegram at any time.

Object 14: CO_2 threshold 3
Initial object for the first CO_2 threshold.
Depending on the parameter setting, a telegram can be sent when exceeding or falling below the set threshold.
The format can be configured as a switch, priority, percentage, or value telegram (see above, *Overview of communication objects, CO_2 sensor*).

Object 15: Block CO_2 threshold 3
Block object for the threshold.
The acting direction (block with 1 or with 0) can be configured.

Object 16: CO_2 threshold 3 – setting/reading threshold
Allows a subsequent change of the threshold via bus telegram at any time.

Object 17: Ventilation of CO_2 - actuating value

Actuating value for ventilation actuator (1 byte telegram).
This object is used if ventilation is only to be activated by the CO_2 content, such as in conference rooms.
The format can be configured as a percentage, or value telegram.

Object 18: Block CO_2 ventilation

Block object for CO_2 dependent ventilation
The acting direction (block with 1 or with 0) can be configured.

Object 19: Ventilation of CO_2 – setting/reading setpoint

Only with *fan control = via PI controller*.
Allows a subsequent change of the CO_2 control setpoint via bus telegram at any time.
Permissible values: 400–1000 ppm. Higher or lower values are not accepted.

Deleting value set via object and restoring ETS value:
When receiving a 0, the previously downloaded ETS setpoint will be restored.

Object 20: Send CO_2 scenes

Sends scene numbers depending on the CO_2 thresholds.

Scene	Telegram		Scene	Telegram		Scene	Telegram		Scene	Telegram	
No.	Hex.	Dec.	No.	Hex.	Dec.	No.	Hex.	Dec.	No.	Hex.	Dec.
1	$00	0	17	$10	16	33	$20	32	49	$30	48
2	$01	1	18	$11	17	34	$21	33	50	$31	49
3	$02	2	19	$12	18	35	$22	34	51	$32	50
4	$03	3	20	$13	19	36	$23	35	52	$33	51
5	$04	4	21	$14	20	37	$24	36	53	$34	52
6	$05	5	22	$15	21	38	$25	37	54	$35	53
7	$06	6	23	$16	22	39	$26	38	55	$36	54
8	$07	7	24	$17	23	40	$27	39	56	$37	55
9	$08	8	25	$18	24	41	$28	40	57	$38	56
10	$09	9	26	$19	25	42	$29	41	58	$39	57
11	$0A	10	27	$1A	26	43	$2A	42	59	$3A	58
12	$0B	11	28	$1B	27	44	$2B	43	60	$3B	59
13	$0C	12	29	$1C	28	45	$2C	44	61	$3C	60
14	$0D	13	30	$1D	29	46	$2D	45	62	$3D	61
15	$0E	14	31	$1E	30	47	$2E	46	63	$3E	62
16	$0F	15	32	$1F	31	48	$2F	47	64	$3F	63

Object 21: Block CO_2 scenes

Block object for the CO_2 dependent scenes
The acting direction (block with 1 or with 0) can be configured.

4.3.3 *Objects for the humidity sensor*

Object 22: Humidity threshold 1
Initial object for the first humidity threshold.
Depending on the parameter setting, a telegram can be sent when exceeding or falling below the set threshold.
The format can be configured as a switching, priority, percentage, or value telegram (see above, *Overview of communication objects, humidity sensor*).

Object 23: Block humidity threshold 1
Block object for the threshold.
The acting direction (block with 1 or with 0) can be configured.

Object 24: Humidity threshold 1 – setting/reading threshold
Allows a subsequent change of the threshold via bus telegram at any time.

Object 25: Humidity threshold 2
Initial object for the second humidity threshold.
Depending on the parameter setting, a telegram can be sent when exceeding or falling below the set threshold.
The format can be configured as a switching, priority, percentage, or value telegram (see above, *Overview of communication objects, humidity sensor*).

Object 26: Block humidity threshold 2
Block object for the threshold.
The acting direction (block with 1 or with 0) can be configured.

Object 27: Humidity threshold 2 – setting/reading threshold
Allows a subsequent change of the threshold via bus telegram at any time.

Object 28: Humidity threshold 3
Initial object for the first humidity threshold.
Depending on the parameter setting, a telegram can be sent when exceeding or falling below the set threshold.
The format can be configured as a switching, priority, percentage, or value telegram (see above, *Overview of communication objects, humidity sensor*).

Object 29: Block humidity threshold 3
Block object for the threshold.
The acting direction (block with 1 or with 0) can be configured.

Object 30: Humidity threshold 3 – setting/reading threshold
Allows a subsequent change of the threshold via bus telegram at any time.

Object 31: Ventilation of humidity - actuating value

Actuating value for ventilation actuator (1 byte telegram).
This object is used if ventilation is to be activated only because of the humidity.
The format can be configured as a percentage, or value telegram.

Object 32: Block ventilation of humidity

Block object for humidity-dependent ventilation
The acting direction (block with 1 or with 0) can be configured.

Object 33: Send humidity scenes

Sends scene numbers depending on the humidity thresholds.

Scene	Telegram		Scene	Telegram		Scene	Telegram		Scene	Telegram	
No.	Hex.	Dec.	No.	Hex.	Dec.	No.	Hex.	Dec.	No.	Hex.	Dec.
1	$00	0	17	$10	16	33	$20	32	49	$30	48
2	$01	1	18	$11	17	34	$21	33	50	$31	49
3	$02	2	19	$12	18	35	$22	34	51	$32	50
4	$03	3	20	$13	19	36	$23	35	52	$33	51
5	$04	4	21	$14	20	37	$24	36	53	$34	52
6	$05	5	22	$15	21	38	$25	37	54	$35	53
7	$06	6	23	$16	22	39	$26	38	55	$36	54
8	$07	7	24	$17	23	40	$27	39	56	$37	55
9	$08	8	25	$18	24	41	$28	40	57	$38	56
10	$09	9	26	$19	25	42	$29	41	58	$39	57
11	$0A	10	27	$1A	26	43	$2A	42	59	$3A	58
12	$0B	11	28	$1B	27	44	$2B	43	60	$3B	59
13	$0C	12	29	$1C	28	45	$2C	44	61	$3C	60
14	$0D	13	30	$1D	29	46	$2D	45	62	$3D	61
15	$0E	14	31	$1E	30	47	$2E	46	63	$3E	62
16	$0F	15	32	$1F	31	48	$2F	47	64	$3F	63

Object 34: Block humidity scenes

Block object for the humidity-dependent scenes
The acting direction (block with 1 or with 0) can be configured.

Object 35: Send dew point alarm

Only available if *monitor dew point = yes* is set (***humidity thresholds*** parameter page).
Sends a 1 if the temperature approaches the dew point, due to the air humidity.

> For use with the RTC, the object has to be connected with the RTC object *dew point alarm – receive* via group address.

4.3.4 Objects for the room temperature controller (RTC)

Object 40: Base setpoint
The function of the object is defined by the parameter *Rotary control function*. [16]

Parameter: *Rotary control function*	Object function
Base setpoint [17]	Sends the base setpoint set at the rotary control.
Manual offset or blocked	Receives the base setpoint from the bus. The base setpoint is first specified via the application at start-up and stored in the *base setpoint* object. Afterwards, it can be specified again at any time this object (limited by the *minimum* or *maximum valid setpoint*).

Object 41: Manual set point offset / set point offset at rotary control [18]
The function of the object is defined by the parameter *Rotary control function*. [19]

Rotary control function	Object function	Data direction
Base setpoint [20], or blocked [21]	Receive manual set point offset. The object receives a temperature difference. The desired room temperature (current setpoint) can be adjusted against the base setpoint by this difference. The following applies in comfort mode (heating): Current setpoint = base setpoint + manual set point offset. Values outside the configured range are limited to the highest or lowest value. If a 0 is received, a previously entered set point offset is reset to 0 K. ⓘ The offset always refers to the set base setpoint and not to the current setpoint.	Receive
Manual offset [22]	Sends the set point offset set at the rotary control.	Send

[16] Only Amun 716 S
[17] Only Amun 716 S
[18] Only Amun 716 S
[19] Only Amun 716 S
[20] Only Amun 716 S
[21] Set basic KNX Multi
[22] Only Amun 716 S

Object 42: Outdoor temperature compensation / adjust setpoint

The function of the object is defined by the parameter *setpoint adjustment at high outside temperature*.

Setpoint adjustment at high outside temperature	Object function	Data direction
Receive only	Receives set point correction for outdoor temperature compensation.	Receive
Calculate internally and send	Reports the current set point correction as an amount or as a differential. The format of the correction value (see following table) is set on the ***set point adjustment*** parameter page.	Send
None		

Format of correction value	Object function	Example
Absolute	Sends the amount: Unadjusted base setpoint + set point correction as setpoint for additional temperature controllers.	Unadjusted base setpoint = 20 °C. Setpoint correction = +2 K The object transmits: 22 °C
Relative	Calculated setpoint correction (in Kelvin) based on outside temperature.	Unadjusted base setpoint = 20 °C. Setpoint correction = +2 K The object sends: 2 K

Object 43: Operating mode preset or night <-> standby

The function of the object is defined by the parameter *Objects for determining the operating mode*.

Objects for determining the operating mode	Object function
new: operating mode, presence, window status	Here, it is a 1 byte object. One of 4 operating modes can be directly activated. 1 = Comfort 2 = Standby 3 = Night, 4 = Frost protection (heat protection) The configured *operating mode after reset* is active until a new valid operating mode is received or changed at the device by the user.
old: comfort, night, frost	With this setting, the object is a 1 bit object. It can be used to activate the operating mode Night or Standby 0=Standby 1=Night

Object 44: Presence or comfort.
The function of the object is defined by the parameter *Objects for determining the operating mode*.

Objects for determining the operating mode	Object function
new: operating mode, presence, window status	Presence: The status of a presence detector (e.g. push button, motion detector) can be received via this object. 1 on this object activates the comfort operating mode.
old: comfort, night, frost	Comfort: 1 on this object activates the comfort operating mode. This operating mode takes priority over night and standby modes. Comfort mode is disabled again by sending a 0 to the object. ⓘ It should not be sent cyclically on this object, since a comfort extension (via the button at the device) will be deleted if a 0 is a received.

Object 45: Window status, or frost/heat protection
The function of the object is defined by the parameter *Objects for determining the operating mode*.

Objects for determining the operating mode	Object function
new: operating mode, presence, window status	Window setting: The status of a window contact can be received via this object. 1 on this object activates the frost / heat protection operating mode.
old: comfort, night, frost	Frost/heat protection: 1 on this object activates the frost protection operating mode. During cooling mode, the heat protection operating mode is activated. The frost/heat protection operating mode has highest priority. Frost/heat protection mode remains active, until it is cleared again by a 0.

Object 46: Current operating mode.
Sends the current HVAC operating mode.
The transmission behaviour can be defined on the ***Settings*** parameter page.

Value	HVAC operating mode
1	Comfort
2	Standby
3	Night
4	Frost protection/heat protection

Object 47: Operating mode as scene.
Teaching in and calling up scenes.
A scene only consists of the current operating mode preset.
Saving scenes: The current value of the object *operating mode preset* is saved together with the corresponding scene number.
Calling up scenes: The content of the object *operating mode preset* will be overwritten by the saved value, and the new operating mode will be accepted by the RTC.
See in the Appendix, *Operating mode as scene*

Object 48: Heating actuating value or heating/cooling actuating value.
Sends the current heating actuating value (0...100%) or heating or cooling if the *output of cooling actuating value* parameter has been set to *together with heating actuating value*. (parameter page ***Cooling control***).

Type of control	Object format
Continuous	1 byte
2-point	1 bit

Object 49: Additional heating stage actuating value or PWM additional heating stage
Sends the actuating value for the additional heating stage depending on the configuration, as 1 bit PWM, or 1 byte percentage telegram.
This object is only available if the additional stage is used.

Object 50: Cooling actuating value
Sends the current actuating value or cooling switching command depending on the type of control selected on the ***cooling control*** parameter page.
The object is only available if the cooling function has been selected on the ***Settings*** parameter page (*control* = heating and cooling).

Object 51: Additional cooling stage actuating value or PWM additional cooling stage
Sends the actuating value for the additional cooling stage depending on the configuration, as 1 bit PWM, or 1 byte percentage telegram.
This object is only available if the additional stage is used.

Object 52: Send heating/cooling operation, or change over between heating and cooling

The object is available if the cooling function has been selected on the ***Settings*** parameter page (*control = heating and cooling*).
The function of the object depends on the *change over between heating and cooling* parameter on the ***cooling control*** parameter page.

Parameter: Change over between heating and cooling	Function
Automatic	Reports whether the room thermostat is currently operating in heating or cooling mode.
Via object	Receives the switching command for switching between heating and cooling mode.

The telegram format can be set on the ***cooling control*** parameter page:

Parameter: *Format object heating/cooling*	Telegram format
DPT1.100	Heating = 1, Cooling = 0
Inverted	Heating = 0, Cooling = 1

Object 53: Current setpoint

Sends the currently set temperature.
The transmission behaviour can be set on the ***heating setpoints*** parameter page.

Object 54: Control actual value

Sends the actual value actually used by the room temperature controller.

ⓘ The control actual value might, depending on the *source for the actual value*, deviate from the internally measured temperature (object *temperature value*).

Object 55: External actual value

Only available if *external actual value* has been selected as a source.

Receives the room temperature from another measurement point via the bus.
This object can be activated on the ***actual value*** parameter page.

Object 56: Send actual value failure

Only available if the actual value monitoring is activated (*Monitor actual value = yes*).
Sends a 1, as soon as one of the selected sources for the actual value provides an unusable value, or (if selected) if no new actual value telegram has been received by the *external actual value* object within the actual value monitoring time.

Unusable temperature values might occur if a temperature sensor is mechanically damaged, or if the electrical connection is interrupted or short-circuited.

ⓘ As long as at least one valid actual value remains available, this will be continued to be used for control. This is the case if the average value is determined out of 2 or 3 sources.

Object 57: Receive outdoor temperature

Only available if the *set point correction at high outdoor temperature* parameter has been set to *calculate internally and send*.
Receives the outdoor temperature for internal set point adjustment in cooling mode

Object 58: Receive dew point alarm

The object is available if the cooling function has been selected on the ***Settings*** parameter page (*control = heating and cooling*).
When receiving a 1, cooling will be stopped, so the temperature cannot drop down to the dew point.

4.3.5 Objects for the external inputs: Switch function

Object 60: Channel I1.1
First initial object of the channel (First telegram).
6 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x, 4 byte DPT 14.x.

Object 61: Channel I1.2
Second initial object of the channel (Second telegram).
6 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x, 4 byte DPT 14.x.

Object 62: Channel I1.3
Third initial object of the channel (Third telegram).
6 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x, 4 byte DPT 14.x.

Object 64: Channel I1 block = 1, or block = 0
The channel is blocked via this object.
The acting direction of the block object and behaviour when setting or cancelling the block can be set on the ***Channel 1*** parameter page.

Objects 65-79
Objects for channels I2-I4.

4.3.6 Objects for the external inputs: Button function

Object 60: Channel I1.1
First initial object of the channel (First telegram).
6 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x, 4 byte DPT 14.x.

Object 61: Channel I1.2
Second initial object of the channel (Second telegram).
6 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x, 4 byte DPT 14.x.

Object 62: Channel I1.3
Third initial object of the channel (Third telegram).
6 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x, 4 byte DPT 14.x.

Object 64: Channel I1 block = 1, or block = 0
The channel is blocked via this object.
The acting direction of the block object and behaviour when setting or cancelling the block can be set on the ***Channel 1*** parameter page.

Objects 65-79
Objects for channels I2-I4.

4.3.7 Objects for the external inputs: Dimming function

Object 60: Channel I1.1 switching
Switches the dimmer on and off.

Object 61: Channel I1.1 brighter, darker, brighter/darker
4-bit dim commands.

Object 62: Channel I1.1 – switching, priority, percentage value..
Initial object for the additional function with double-click.
6 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x,
4 byte DPT 14.x.

Object 64: Channel I1 block = 1, or block = 0
The channel is blocked via this object.
The acting direction of the block object and behaviour when setting or cancelling the block can be configured.

Objects 65-79
Objects for channels I2-I4.

4.3.8 Objects for the external inputs: Blinds function

Object 60: Channel I1 UP/DOWN, UP, DOWN
Sends operating command to the blind actuator.

Object 61: Channel I1 Step/Stop
Sends Step/Stop commands to the blind actuator.

Object 62: Channel I1.1 – switching, priority, percentage value.., height % + slat %
Initial object for the additional function with double-click.
7 telegram formats can be set:
Switching ON/OFF, priority, send percentage value, send value, 2 byte DPT 9.x,
4 byte DPT 14.x, height % + slat %.

Object 64: Channel I1 block = 1, or block = 0
The channel is blocked via this object.
The acting direction of the block object and behaviour when setting or cancelling the block can be configured.

Objects 65-79
Objects for channels I2-I4.

4.3.9 Objects for the external inputs I3 and I4: Temperature sensor function

The external inputs I3 and I4 can be used as analogue inputs for temperature measurement via remote sensor.

This function is activated on the ***General*** parameter page with the parameter *function of the external inputs I3 + I4*.

Object 70: Channel I3 temperature actual value
Sends the temperature measured by the external sensor at I3.

Object 75: Channel I4 temperature actual value
Sends the temperature measured by the external sensor at I4.

4.3.10 Diagnosis and alarm objects

Object 80: Firmware version
Sends firmware version information as DPT version (DPT217.001).
Can be read out for diagnostic purposes via the ETS.
See in the Appendix: Firmware version.

Object 81: Alarm info
Reports error or alarm as DPT_AlarmInfo (DPT219.001).

Object 83: Alarm error text
Diagnostic object: Sends a short alarm text in case of an error (14 characters) as DPT_String_ASCII (DPT16.000).

Alarm reason	Error text
CO_2 sensor failure	CO_2 Fault
Humidity/pressure sensor failure	Humidity Fault
Internal temperature sensor failure	Temp Fault
Rotary control failure [23]	Wheel Fault
Light sensor failure	ALS Fault
Sensor failure external analog input 3	I3 Temp Fault
Sensor failure external analog input 4	I4 Temp Fault
CO_2 thresholds invalid	Thresholds CO_2
Humidity thresholds invalid	Thresholds Hum

If an alarm or an error occurs, this will be indicated by sending object *Alarm info*. Additionally, the object *Alarm error text* will be sent, which show the error code and a short error text.
If several alarms are active, the object *Alarm error text* will be cyclically sent with the alarm information at an interval of 10 s.
If all active alarms have been processed, the cyclical sending will be repeated after a pause of 30 s.
If no alarms are active anymore, the object *Alarm error text* (blank string) will be sent once.

[23] Only Amun 716 S

4.3.11 Objects for measurement value calibration

Object 84: CO_2 offset - measurement value offset

Receives a value by which the current CO_2 measurement value is to be offset.
Format: DPT9.* 2 byte floating value. (max. ±1000 ppm).

Object 85: CO_2 offset – receive measurement value

Receives an absolute CO_2 measurement value, and adjusts the internally measured CO_2 value to the received CO_2 value.

Format: DPT9.008.

4.3.12 Objects for the comparator

Objects 86, 87, 88: input 1, input 2, input 3
Available if the respective parameter, *input 1, 2* or *3* is set to *input object (0-100%)*.
This allows the use of external actuating values for the determination of the comparison value.

Object 89: output
Sends the determined comparison value.

4.4 Parameter pages overview

The device consists of one general block and 5 main functional blocks.

Parameter page	Description
General functional block	
General	LED settings [24], activation of the temperature sensor inputs.
Measurement values	Settings for sending CO_2, humidity, temperature, air pressure, and comfort degree.
CO_2 sensor functional block	
CO_2 thresholds	Setting the 3 CO2 thresholds.
CO_2 threshold 1	Setting of response to exceeding or falling below the respective CO_2 threshold.
CO_2 threshold 2	
CO_2 threshold 3	
Ventilating CO_2	Setting of ventilation speed depending on CO_2 content
CO_2 scenes	Setting of scene numbers to be sent depending on CO_2 content
Humidity sensor functional block	
Humidity thresholds	Setting of the 3 humidity thresholds
Humidity threshold 1	Setting of response to exceeding or falling below the respective humidity threshold
Humidity threshold 2	
Humidity threshold 3	
Ventilating humidity	Setting of ventilation speed depending on relative humidity
Humidity scenes	Setting of scene numbers to be sent depending on relative humidity
RTC functional block	
Setting	General settings for operation and temperature control
Actual value	Source for actual value measurement, actual value monitoring, etc.
Operating Mode	Operating mode after reset, presence sensor etc.
Heating control	Control parameters, installation type etc. for heating mode.
Heating setpoints	Base setpoint value, lowering, frost protection etc.
Additional stage heating	Type of actuating value, proportional band, transmission behaviour.
Cooling control	Control parameters, installation type etc. for cooling mode.
Cooling setpoints	Dead zone, standby, heat protection etc.
Set point adjustment	Setting maximum adjustment.
Additional stage cooling	Type of actuating value, proportional band, transmission behaviour.
Comparator functional block	
Comparator	Compares up to 3 actuating values and determines the minimum, maximum or mean value from them.
External inputs functional block	
Channel I1	Function of the input, debounce time, number of telegrams, block function, etc. Additionally at I3 and I4: Selection of the temperature sensor, temperature calibration, etc.
Channel I2	
Channel I3	
Channel I4	
Switch object 1	Object type, transmission behaviour, etc. can be set for each object individually.
Switch object 2	
Switch object 3	
Button object 1	Object type, transmission behaviour, etc. can be set for each object individually.
Button object 2	
Button object 3	
Dimming	Type of control

[24] Only Amun 716 S

Parameter page	Description
Blinds	Type of control
Double-click	Additional telegrams for *dimming* and *blinds*

4.5 General parameters

4.5.1 General

Designation	Values	Description
Device type	***Amun 716 S***	Standard device.
	Set basic KNX Multi (Order No. 9070900)	Set basic KNX Multi for use with thePrema 360 P.
Reduce brightness of the LEDs [25]		The LEDS should:
	never	Shine every time at maximum brightness.
	always	Always shine at the specified brightness
	only in night mode	Shine at the specified brightness if the RTC is set to night mode.
	in the dark	Shine at the specified brightness when it is dark in the room.
	via bus	Be able to be reduced or dimmed via bus telegrams.
Object type [26]	***via switch object***	Brightness reducible via switch telegram.
	via percentage value	The brightness of the LEDs can be set as desired via dimming telegrams.
Value for reduced brightness [27]	*0-100%* Std. = ***30%***	LED brightness for the setting *always only in night mode*, or *via switch object*.
Function of the external inputs I3+I4	***Binary input***	I3 and I4 are normal binary inputs, as I1 and I2.
	Temperature sensor input	I3 and I4 or used for temperature measurement, as well as for the internal RTC and for other bus sharing units. For this purpose, always one remote sensor is connected.

ⓘ The external inputs I3 and I4 can be used as analogue inputs for temperature measurement via remote sensor.

[25] Only Amun 716 S
[26] Only Amun 716 S
[27] Only Amun 716 S

4.5.2 Measurement values

Designation	Values	Description
CO_2		
Send CO_2 content on change of	*not due to a change* *100 ppm* ***200 ppm*** *300 ppm* *500 ppm*	only send cyclically (if enabled) Send if the value has changed since the last transmission by the selected amount
Enable fresh air calibration	***no*** *yes*	If the device is properly used, a calibration procedure is not necessary. See in the Appendix: *Fresh air calibration* ⓘ With the setting *yes*, the calibration is not started, but admitted for the duration of 2 h after download.
Reference value for fresh air calibration	*350-600 ppm*	Standard value: **400 ppm** See in the Appendix: *Fresh air calibration*
Send CO_2 content cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* … *every 45 min* *every 60 min*	How often should it be resent?
HUMIDITY		
Send humidity value on change of	*not due to a change* *2%* ***3%*** *5%* *10%*	only send cyclically (if enabled) Send if the value has changed by the selected amount since the last transmission.
Send humidity value cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* … *every 45 min* *every 60 min*	How often should it be resent?
TEMPERATURE		

Designation	Values	Description
Transmit temperature on change of (internal sensor)		Only valid for temperature measurement at the internal sensor.
	not due to a change *0.2 K* *0.3 K* *0.5 K* *0.7 K* *1 K* *1.5 K* *2 K*	Only send cyclically (if enabled) Send if the value has changed by the selected amount since the last transmission.
Send temperature cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?
Temperature calibration	*-64..+64* (x 0.1 K)	Correction value for temperature measurement if sent temperature deviates from the actual ambient temperature. **Example:** Temperature = 20°C sent temperature = 21°C Correction value = 10 (d.h. 10 x 0.1°C)
AIR PRESSURE		
Send air pressure on change of	*not due to a change* *10 hPa (mbar)* *20 hPa (mbar)* ***30 hPa (mbar)*** *40 hPa (mbar)* *50 hPa (mbar)*	only send cyclically (if enabled) Send if the value has changed by the selected amount since the last transmission.
Send air pressure cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?
COMFORT		

Designation	Values	Description
Send degree of comfort cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent? (See in the Appendix: *Comfort*)

4.6 CO_2 sensor functional block

4.6.1 CO2 thresholds

Designation	Values	Description
Number of CO_2 thresholds	*1, 2, 3*	Define required thresholds.
Thresholds can be overwritten via object	***no*** *yes*	The thresholds can only be set in the ETS. The thresholds can be changed by bus telegrams any time.
Overwrite threshold after download	***no*** *yes*	After download, the previously saved thresholds are preserved. A download deletes and overwrites all saved thresholds.
Parameter mode for thresholds	***Standard*** *Expert mode*	Only the thresholds are entered. The hysteresis will be calculated automatically. Thresholds and hysteresis can be entered.
Standard mode		
CO_2 threshold 1	*600-799 ppm* Std. = ***700 ppm***	Input of first threshold.
CO_2 threshold 2	*800-1399 ppm* Std. =***1300 ppm***	Input of second threshold.
CO_2 threshold 3	*1400-2500 ppm* Std. =***1800 ppm***	Input of third threshold.
Expert mode		
Hysteresis 1	***100 ppm*** *200 ppm* *300 ppm* *500 ppm*	The hysteresis prevents frequent switching after small changes in readings.
CO_2 threshold 1	*600-2500 ppm* Std. = ***600 ppm***	Input of first threshold.
Hysteresis 2	***100 ppm*** *200 ppm* *300 ppm* *500 ppm*	The hysteresis prevents frequent switching after small changes in readings.
CO_2 threshold 2	*600-2500 ppm* Std. =***1000 ppm***	Input of second threshold.
Hysteresis 3	*100 ppm* ***200 ppm*** *300 ppm* *500 ppm*	The hysteresis prevents frequent switching after small changes in readings.
CO_2 threshold 3	*600-2500 ppm* Std. =***1500 ppm***	Input of third threshold.

For the expert mode: The distance of two thresholds must be at least as big as the hysteresis in between.
If the LED at the device flashes red [28], the thresholds are not correctly defined

For the thresholds at CO_2 and humidity, the hysteresis is negative on one side, that is.
Threshold exceeded = measurement value ≥ threshold
fallen below threshold = measurement value ≤ threshold − hysteresis

[28] Only Amun 716 S

4.6.2 Thresholds 1, 2, 3 CO2

The parameters are identical for all 3 thresholds.
Threshold 1 is given as an example.

Designation	Values	Description	
Telegram type for threshold 1 CO_2	***Switch command*** *Priority* *Percentage value* *Value*	Select telegram type for this threshold.	
When exceeding the threshold	*no telegram,* ***send following telegram once,*** *send cyclically*	Response if the measured value lies above the set threshold.	
Telegram	With object type = switching (1 bit)		
	ON	Send switch-on command	
	OFF	Send switch-off command	
	With object type = priority (2 bit)		
		Function	Value
	no priority	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = *value 0-255*		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = *percentage value (1 byte)*		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	
When falling below threshold	*no telegram,* **send following telegram once,** *send cyclically*	Response if the measured value lies above the set threshold.	
Telegram	With object type = switching (1 bit)		
	ON	Send switch-on command	
	OFF	Send switch-off command	
	With object type = priority (2 bit)		
		Function	Value
	no priority	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = *value 0-255*		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = *percentage value (1 byte)*		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	

Designation	Values	Description
Response when setting the block	***ignore***	The threshold cannot be blocked.
	do not send	The threshold will not send as long as the block object is set.
	as with underrun threshold	Same reaction as set in the When threshold is not exceeded parameter (see above).
	as with exceeded threshold	Same reaction as set in the With exceeding the threshold parameter (see above).
Response when cancelling the block	*do not send*	Not automatically resent when the block is cancelled
	update	The current threshold status is sent immediately after cancelling the block
Block telegram	***Block with 1 (standard)***	0 = enable 1 = block
	Block with 0	0 = block 1 = enable
Send cyclically	*every min,* *every 2 min* *every 3 min* ... ***every 30 min*** *every 45 min* *every 60 min*	How often should it be resent?

4.6.3 Ventilating CO2

Designation	Values	Description
Fan control via	***fixed values***	Up to 3 fixed fan speeds
	PI controller	Infinite PI fan control via CO_2 setpoint.
Fixed values		
Send given value:	***Input as percentage***	Input format for fan control. This setting is not relevant for the actuator.
	Input as a number from 0 to 255	
If CO_2 below threshold 1	***0**-100% or 0-255*	Setting of desired Fan speeds depending on CO_2 content.
If CO_2 between threshold 1 and 2	*0-100% or 0-255* Std.: ***35%***	
If CO_2 between threshold 2 and 3	*0-100% or 0-255* Std.: ***70%***	
If CO_2 greater than threshold 3	*0-**100%** or 0-255*	
PI controller		
Setpoint overwritable via object	***no***	The setpoint can only be set in the ETS.
	yes	The setpoint can be changed by bus telegrams any time. **Permissible values:** 400 – 1000 ppm.
Overwrite setpoint after download	***no***	After download, the previously saved value will be preserved.
	yes	A download deletes and overwrites the saved setpoint.
Setpoint	*400-1000 ppm* Std.: ***500 ppm***	CO_2 setpoint for fan control
Control parameters	***Standard***	Standard application: P and I share preset. P = 150 ppm I = 15 min
	User-defined	Professional use: Configure P/PI controller yourself
I share	*Pure P controller*	Proportional control, no integrated time.
	5 min	The integrated time determines the response time of the control. It establishes the increase by which the actuating value from the controller is raised in addition to that from the P-term. The I share remains active for as long as there is a control deviation. The I share is added to the P share.
	10 min	
	15 min	
	20 min	
	25 min	
	30 min	

Designation	Values	Description
P share	*100 ppm* ***150 ppm*** *200 ppm* *250 ppm* *300 ppm* *350 ppm* *400 ppm*	Professional setting for adapting control response to the room. Small values cause large changes in actuating values, larger values cause finer actuating value adjustment.
Minimum actuating value	***0%,*** *10%, 20%* *30%, 40%, 50%*	Lowest permissible actuating value. Can, for instance, be used to maintain a permanent air flow, even if the controller says 0 %.
Response when falling below the minimum actuating value	***Output 0%*** *Output minimum actuating value*	If the actuating value is too low, always switch off the fan. The fan should not be at a complete standstill, but continue to run with the minimum actuating value, and thus ensure a permanent air exchange.
Maximum actuating value	*50%, 60%, 70%* *80%, 90%,* ***100%***	Highest permissible actuating value.
Send actuating value on change	*No, only send cyclically* *by 2%,* ***by 5%*** *by 10%, by 20%*	After what percentage change in the actuating value is the new value to be transmitted?
Response when setting the block	***Ignore block*** *do not send* *Send following value*	No block function, no further block parameters. Response to block telegram.
Value if ventilation blocked	***0****-100%*	Fan speed in block mode.
Response when unlocking	*do not send* ***update***	No response. send current control setpoint.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable
Send fan value cyclically/send actuating value cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* *…* *every 45 min* *every 60 min*	How often should it be resent?

4.6.4 CO2 scenes

Designation	Values	Description
Send specified scene		
If CO_2 below threshold 1	***Scene 1*** ... *Scene 64*	Setting of scene numbers to be sent depending on CO_2 content.
If CO_2 between threshold 1 and 2	*Scene 1* ***Scene 2*** ... *Scene 64*	
If CO_2 between threshold 2 and 3	*Scene 1* ... ***Scene 3*** ... *Scene 64*	
If CO_2 greater than threshold 3	*Scene 1* ... ***Scene 4*** ... *Scene 64*	
Response when setting the block	***Ignore block*** *do not send* *Send following scene*	No block function, no further block parameters. Response to block telegram.
If scenes CO_2 blocked	***Scene 1*** ... *Scene 64*	Scene to be sent when the block object is set.
Response when cancelling the block	*do not send* ***update***	no response. send current control setpoint.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable
Send scene number cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?

4.7 Humidity sensor functional block

4.7.1 Humidity thresholds

Designation	Values	Description
Number of humidity thresholds	*1* *2* ***3***	Define required thresholds.
Thresholds can be overwritten via object	***no*** *yes*	The thresholds can only be set in the ETS. The thresholds can be changed by bus telegrams any time
Overwrite threshold after download	***no*** *yes*	After download, the previously saved thresholds are preserved. A download deletes and overwrites all saved thresholds.
Monitor dew point	***no*** *yes*	No monitoring Show the ***Monitor dew point*** parameter page.
Parameter mode for thresholds	***Standard*** *Expert mode*	Only the thresholds are entered. The hysteresis will be calculated automatically. Thresholds and hysteresis can be entered.
Standard mode		
Humidity threshold 1	*10-39%* **Std.: *35%***	Input of first threshold
Humidity threshold 2	*40-59%* **Std.: *50%***	Input of second threshold
Humidity threshold 3	*60-90%* **Std.: *65%***	Input of third threshold
Expert mode		
Hysteresis 1	*1%, 2%,* ***3%*** *5%, 7%, 10%*	The hysteresis prevents frequent switching after small changes in readings.
Humidity threshold 1	*10-100%* **Std.: *45%***	Input of first threshold
Hysteresis 2	*1%, 2%,* ***3%*** *5%, 7%, 10%*	The hysteresis prevents frequent switching after small changes in readings.
Humidity threshold 2	*10-100%* **Std.: *55%***	Input of second threshold
Hysteresis 3	*1%, 2%,* ***3%*** *5%, 7%, 10%*	The hysteresis prevents frequent switching after small changes in readings.
Humidity threshold 3	*10-100%* **Std.: *70%***	Input of third threshold

For the thresholds at CO_2 and humidity, the hysteresis is negative on one side, that is.
Threshold exceeded = measurement value ≥ threshold
fallen below threshold = measurement value ≤ threshold − hysteresis

4.7.2 Humidity thresholds 1, 2, 3

The parameters are identical for all 3 thresholds.
Threshold 1 is given as an example.

Designation	Values	Description	
Telegram type for humidity threshold 1	***Switch command*** *Priority* *Percentage value* *Value*	Select telegram type for this threshold.	
When exceeding the threshold	*no telegram*, ***send following telegram once,*** *send cyclically*	Response if the measured value lies above the set threshold.	
Telegram	With object type = switching (1 bit)		
	ON	Send switch-on command	
	OFF	Send switch-off command	
	With object type = priority (2 bit)		
		Function	Value
	no priority	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = *value 0-255*		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = *percentage value (1 byte)*		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	
When falling below threshold	*no telegram*, ***send following telegram once,*** *send cyclically*	Response if the measured value lies above the set threshold.	
Telegram	With object type = switching (1 bit)		
	ON	Send switch-on command	
	OFF	Send switch-off command	
	With object type = priority (2 bit)		
		Function	Value
	no priority	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = *value 0-255*		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = *percentage value (1 byte)*		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	

Designation	Values	Description
Response when setting the block	***ignore***	The threshold cannot be blocked.
	do not send	The threshold will not send as long as the block object is set.
	as with underrun threshold	Same reaction as set in the When threshold is not exceeded parameter (see above).
	as with exceeded threshold	Same reaction as set in the With exceeding the threshold parameter (see above).
Response when cancelling the block	*do not send*	Not automatically resent when the block is cancelled
	update	The current threshold status is sent immediately after cancelling the block
Block telegram	***Block with 1 (standard)***	0 = enable 1 = block
	Block with 0	0 = block 1 = enable
Send cyclically	*every min,* *every 2 min* *every 3 min* … ***every 30 min*** *every 45 min* *every 60 min*	How often should it be resent?

4.7.3 Ventilating humidity

Designation	Values	Description
Send given value:	***Input as percentage*** *Input as a number from 0 to 255*	Input format for fan control. This setting is not relevant for the actuator.
If humidity is below threshold 1	***0****-100% or 0-255*	Setting of desired fan speeds depending on humidity.
If humidity is between thresholds 1 and 2	*0-100% or 0-255* Std.: ***35%***	
If humidity is between thresholds 2 and 3	*0-100% or 0-255* Std.: ***70%***	
If humidity is greater than threshold 3	*0-****100%*** *or 0-255*	
Response when setting the block	***Ignore block*** *do not send* *Send following value*	No block function, no further block parameters. Response to block telegram.
Value if ventilation blocked	***0****-100%*	Fan speed in block mode.
Response when unlocking	*do not send* ***update***	no response. send current control setpoint.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable
Send fan value cyclically [29] *send actuating value cyclically* [30]	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* *…* *every 45 min* *every 60 min*	How often should it be resent?

[29] With fan control via: fixed values
[30] With fan control via: PI controller

4.7.4 Humidity scenes

Designation	Values	Description
Send specified scene		
If humidity is below threshold 1	***Scene 1*** … Scene 64	Setting of scene numbers to be sent depending on humidity.
If humidity is between thresholds 1 and 2	*Scene 1* ***Scene 2*** … *Scene 64*	
If humidity is between thresholds 2 and 3	*Scene 1* … ***Scene 3*** … *Scene 64*	
If humidity is greater than threshold 3	*Scene 1* … ***Scene 4*** … *Scene 64*	
Response when setting the block	***Ignore block*** *do not send* *Send following scene*	No block function, no further block parameters. Response to block telegram.
If humidity scenes blocked	***Scene 1*** … *Scene 64*	Scene to be sent when the block object is set.
Response when unlocking	*do not send* ***update***	no response. Send the current control setpoint.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable
Send scene number cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* … *every 45 min* *every 60 min*	How often should it be resent?

4.7.5 Dew point monitoring

In order to avoid the formation of condensate in cooling mode, a dew point alarm will be sent and cooling will be stopped, as soon as the humidity has reached a critical value.
For this purpose, the *Send dew point alarm* object will be linked with the RTC object *Receive dew point alarm*.

Designation	Values	Description
Threshold for dew point alarm	*60-90%*	From which rel. humidity should the dew point alarm be triggered and sent?
Send dew point alarm cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?

4.8 Room temperature controller RTC functional block

4.8.1 Setting

Designation	Values	Description
Control	***Heating control only***	Heating mode only
	Heating and cooling	Additionally, a cooling system is to be controlled.
Rotary control function [31]	***Base setpoint***	The base setpoint is exclusively set at the rotary control.
	Manual offset	The setpoint can be adjusted via the rotary control. The base setpoint is received via the *base setpoint* object.
	Blocked [32]	The rotary control has no function. The base setpoint is received via the *base setpoint* object.
Manual offset works	***in comfort, standby and night mode,*** *in comfort and standby, only for comfort*	The set point offset: Is only considered in the selected mode and is ineffective in all operation modes.
Manual offset at the end of night mode	***Do not change*** *Reset to 0 K*	Only available if the *Rotary control function* is set to *base setpoint* or *blocked*.
Use floor temperature limitation (sensor at I4)	***No***	no floor temperature limitation.

[31] Only Amun 716 S
[32] Set basic KNX Multi

Designation	Values	Description
	yes	The floor temperature is measured by a sensor at external input I4. On the ***Heating setpoints*** parameter page the *Maximum floor temperature* parameter is shown. **Functionality:** If the *Maximum floor temperature* is reached, the heating actuating value is reduced to 0%. The hysteresis is 5 K. Prerequisite: The *Function of the external inputs I3+I4* parameter on the ***General*** parameter page has to be set to *Temperature sensor input*. See also Chapter: *External inputs I1-I4 functional block* → *Temperature sensor function (only I3 and I4)*
Set point correction at high outside temperature	***None*** *Receive only* ***Calculate internally and send***	Function is deactivated The correction value is received by the bus, and the own setpoint is adjusted to the increase in outside temperature. The device calculates the correction value, sends it to other controllers and adjusts the own setpoint to the increase in outside temperature. See in the Appendix: *Set point correction*
Button function [33]	*Blocked* ***Presence buttons*** *Select operating modes*	No function. The button starts the comfort operating mode. The button is used for selecting the operating mode.

[33] Only Amun 716 S

4.8.2 Actual value

On this parameter page the source is selected which is used as the actual value for control. This can be the temperature sensor integrated in the device, an external sensor, or a combination of up to 3 sensors.

ⓘ The control actual value might, depending on the selection of the *source for the actual value*, deviate from the internally measured temperature (object *temperature value*).

Designation	Values	Description
Source for actual value [34]	***Internal sensor***	Control actual value. The device measures and controls the room temperature via the internal sensor. (Control actual value = internally measured temperature).
	External actual value object	The room temperature is solely acquired via the bus.
	Average value of internal + ext. actual value object	The device calculates the average value of the room temperature received from the bus and the internal measurement.
	Sensor at I3	External sensor at I3.
	Average value of internal + I3	Average value of the internal value and the value measured at I3.
	Average value of I3 + ext. actual value obj.	Use average value of I3 and bus.
	Average value of internal + I3 + ext. actual value obj.	Use average value from 3 sources: I3 + internal + bus.
Send control actual value in the event of change of	***not due to a change***	only cyclical sending possible.
	0.2 K, 0.3 K, 0.5 K, 0.7 K 1 K, 1.5 K, 2 K	Minimum change for resending.
Send control actual value cyclically	***no***	Only send in the event of a change.
	yes	Send in the event of a change and cyclically.

[34] *Source for actual value*: The options with I3 are only available if the external inputs for temperature measurement are set, i.e. *Function of external inputs I3+I4 = temperature sensor input* (See General parameter page).

Designation	Values	Description
Monitor actual value	***no*** *yes*	No monitoring. All selected actual value sources are monitored. In case of an error, the object sends *actual value failure* error telegrams. ⓘ As long as at least one valid actual value remains available, this will be continued to be used for control. This is the case if the average value is determined out of 2 or 3 sources.
Monitoring time for external actual value	*2 min, 3 min, 5 min, 10 min, 15 min, 20 min, 30 min, 45 min, 60 min*	Only for the *External actual value* object. If no value is received within the configured time and the object is the only selected source, the emergency program will be activated. ⓘ As long as at least one valid actual value remains available, this will be continued to be used for control, and the emergency program remains inactive. This is the case if the average value is determined out of 2 or 3 sources.

Designation	Values	Description
Emergency program in case of actual value failure	*with PI controller: 0%, with 2-point controller: Off* ***with PI controller: 10%, with 2-point controller: On*** *with PI controller: 20%, with 2-point controller: On* *with PI controller: 30%, with 2-point controller: On* *with PI controller: 50%, with 2-point controller: On*	The emergency program will only be executed if the selected Source for actual value provides no valid value anymore. The heating/cooling will then be controlled with a fixed actuating value. This might be the case if only one *Source for actual value* is selected, e.g. only *I3*. In case of actual value failure, the value of the actuating value for the emergency program will, depending on the operating mode (heating/cooling), be output to the corresponding object. ⓘ As long as at least one valid actual value remains available, this will be continued to be used for control, and the emergency program remains inactive. This is the case if the average value is determined out of 2 or 3 sources. **Example:** *Average value of internal + I3*. If the sensor at I3 fails, the RTC controls with the remaining, i.e. with the internal sensor, in this case.
Actual value failure telegram	*always cyclically* ***only send cyclically in case of an error***	The object sends the current status always cyclically and in the event of a change: Error = 1, no error = 0 Only sends in case of an error, cyclically and in the event of a change: error = 1.
Send cyclically	*every min* *every 2 min* *every 3 min* … ***every 30 min*** … *every 60 min*	How often should it be resent?

4.8.3 Operating Mode

Designation	Values	Description
Operating mode after reset	*Frost protection* *Temperature reduction at night* ***Standby*** *Comfort*	Operating mode after start-up or reprogramming
Objects for determining the operating mode	***New: Operating mode, presence, window status*** *Old: comfort, night, frost*	The operating mode is changed depending on the window and presence contacts. Traditional setting without window and presence status. ⓘ As long as the frost protection object is = 1, no other operating mode can be selected.
Type of presence sensor	 *Presence detector*	Only for *objects for determining the operating mode = new.*. The presence sensor activates comfort operating mode. Comfort operating mode as long as the presence object is set [35].

[35] Exception: If a window is opened (window object = 1), the room thermostat changes to frost protection mode.

Designation	Values	Description
	Presence button	If a new operating mode is received on the operating mode preset object with the presence object set, it will be accepted and the presence object will be reset. Reception of the same operating mode prior to the presence status (e.g. via cycl. sending) is ignored. If the presence object is set during night / frost operation, it is reset after the configured comfort extension finishes [36] If the presence object is set during standby mode, the comfort operating mode is accepted without time restriction.
When increasing the temperature at the rotary control [37]	***Do not set presence object*** *Set presence object*	Only if *type of presence sensor = presence button*. Only increase the temperature Presence object is set, the controller changes to comfort mode.
Time for comfort extension	*30 min* *1 h* *1.5 h* ***2 h*** *2.5 h* *3 h* *3.5 h*	This determines how long the controller should remain in comfort mode after the presence button is pressed.
Cyclical sending of current operating mode	***do not send cyclically*** *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?

[36] Exception: If a window is opened (window object = 1), the room thermostat changes to frost protection mode.
[37] Only Amun 716 S

4.8.4 Heating control

Designation	Values	Description
Type of control	***Continuous***	Infinite control (0.. 100 %).
	2-point	Switching control (On/Off). See in the Appendix: *Continuous and switching control*.
Number of heating stages	***Only one heating stage*** *Main stage and additional stage*	Choice of 1- or 2-stage heating
Hysteresis of 2-point controller	*0.3 K* *0.5 K* *0.7K* ***1 K*** *1.5 K*	Interval between the tripping point (setpoint) and the turn back on point (setpoint – hysteresis). The hysteresis prevents a permanent switching on/off.
Recirculation of hysteresis after switching point	***None*** *0.1 K/min* *0.2 K/min* *0.3 K/min*	The recirculation causes a gradual decrease in the hysteresis over time, and the control accuracy is increased. The hysteresis is equivalent to the programmed value for each switch-off and is gradually reduced by the recirculation process. The hysteresis can reduce to 0 K over prolonged periods of switch-off. When switching on the next time, it will be reset to the configured value.
Setting the control parameters	***Via installation type***	Standard application. The control parameters are preset.
	User-defined	Professional use: Configure P/PI controller yourself.
Installation type	***Radiator heating system***	PI controller with: Integrated time = 90 minutes Bandwidth = 2.5 K
	Underfloor heating	Integrated time = 30 h Bandwidth = 4 K
Proportional band of heating controller	*1 K, 1.5 K, 2 K,* **2.5 K**, *3 K, 3.5 K, 4 K, 4.5 K, 5 K, 5.5 K, 6 K, 6.5 K, 7 K, 7.5 K, 8 K, 8.5 K*	Professional setting for adapting control response to the room. Small values cause large changes in actuating values, larger values cause a finer actuating value adjustment. See in the Appendix: *Temperature control*

Designation	Values	Description
Integrated time of heating controller	*pure P controller* *30 min, 60 min* **90 min**, *120 min* *150 min, 180 min* *210 min* *4 h, 5 h, 10 h* *15 h, 20 h, 25 h* *30 h, 35 h*	Professional setting: See in the Appendix: *Response of the PI controller* This time can be adapted to suit particular circumstances. If the heating system is over-dimensioned and therefore too fast, shorter values should be used. On the other side, longer integration times are beneficial for a slightly undersized heating (slow).
Send heating actuating value cyclically	*At change by 1%* *At change by 2%* *At change by 3%* ***At change by 5%*** *At change by 7%* *At change by 10%* *At change by 15%*	After what percentage change in the actuating value is the new value to be transmitted. Small values increase control accuracy but also the bus load.
Send heating actuating value cyclically	***do not send cyclically*** *every 2 min* *every 3 min* … *every 45 min* *every 60 min*	How often should it be resent?

4.8.5 Heating setpoints

Designation	Values	Description
Base setpoint after loading the application	*18 °C, 19 °C, 20 °C* ***21 °C****, 22 °C, 23 °C 24 °C, 25 °C*	Output setpoint for temperature control.
Minimum valid base setpoint	*5-20°C in 1 degree increments* Std.: ***10 °C***	If the object receives a base setpoint which is lower than the minimum valid base setpoint, the base setpoint will be increased to the value set here.
Maximum valid base setpoint	*17..***32 °C** *in 1 degree increments*	If the object receives a base setpoint which is higher than the maximum valid base setpoint, the base setpoint will be set to the value set here.
Maximum valid set point offset	*+/- 1 K +/- 2 K* ***+/- 3 K*** *+/- 4 K +/- 5 K*	Limits the possible setting range for the setpoint offset function. Is valid for the *Man. set point offset* as well as for the rotary control.
Reduction in standby mode (when heating)	*0 K, 0.5 K, 1 K, 1.5 K,* ***2 K,*** *2.5 K, 3 K, 3.5 K, 4 K, 4.5 K, 5 K*	**Example:** With a base setpoint of 21 °C in heating mode and a reduction of 2K, the device controls with a setpoint of 21 − 2 = 19 °C.
Reduction in night mode (during heating)	*3 K, 4 K,* ***5 K*** *6 K, 7 K, 8 K*	By what value should the temperature be reduced in night mode?
Setpoint for frost protection mode (during heating)	*3-10 °C* Std.: ***6 °C***	Preset temperature for frost protection mode in heating mode (Heat protection applies in cooling mode).
Current setpoint in comfort mode	***Actual value (heating <> cooling)***	Feedback of current setpoint value via the bus: The setpoint actually being used for control is always to be sent (= current setpoint). **Example** with Base setpoint 21 °C and dead zone 2 K: During heating, 21 °C is transmitted and during cooling, base setpoint + dead zone is transmitted (21 °C + 2 K = 23 °C)

Designation	Values	Description
	Average value betw. heating and cooling	Same value in comfort mode during both heating and cooling mode, i.e.: base setpoint + half dead zone will be sent, so users of the room will not be irritated. **Example** with Base setpoint 21°C and dead zone of 2 K: Average value = 21 °C+1 K = 22 °C, but 21 °C or 23 °C are used for control
Maximum floor temperature [38]	*24 °C, 26 °C, 28 °C* ***30 °C,*** *32 °C, 34 °C* *36 °C, 38 °C, 40 °C*	Maximum permissible floor temperature.
Cyclical sending of current setpoint	***do not send cyclically*** *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?

[38] *Maximum floor temperature*: This parameter is only available if the external inputs for temperature measurement are set, i.e. *Function of external inputs I3+I4 = temperature sensor input* (see ***General*** parameter page) and the parameter *Use floor temperature limitation (sensor at I4)* is set to *yes*.

4.8.6 Additional stage heating

Temperature control is done via a proportional controller.

Designation	Values	Description
Output of the actuating value		Control is done via a proportional controller.
	Percent	Continuous actuating value 0-100 %
	PWM	Pulse-width modulated switching actuating value.
Difference between main stage and additional stage	*0 K, 0.5 K, 1 K* *1.5 K,* **2 K**, *2.5 K* *3 K, 3.5 K, 4 K*	Defines the negative distance between the current setpoint and the setpoint of the additional stage. **Example** with basic setpoint of 21 °C and difference of 1 K: The main stage controls with the base setpoint and the addition stage controls with base setpoint value – 1K = 20°C
Proportional band	*1 K, 1.5 K, 2 K, 2.5 K* *3 K, 3.5 K,* **4 K**, *4.5 K* *5 K, 5.5 K, 6 K, 6.5 K* *7 K, 7.5 K, 8 K, 8.5 K*	With continuous additional stage, Professional setting for adapting control response to the room. Large values cause finer changes to the control variables with the same control deviation and more precise control than smaller values.
PWM period	*3-30 min* Std.: ***5 min***	An actuation cycle consists of a switching-on and a switching-off process and forms a PWM period. **Example:** Actuating value = 20 %, PWM time = 10 min: In an actuating cycle of 10 min, 2 min switched on and 8 min switched off (i.e. 20 % on/80 % off).
Transmission of actuating value	*At change by 1%* *At change by 2%* *At change by 3%* ***At change by 5%*** *At change by 7%* *At change by 10%* *At change by 15%*	After what percentage change in the actuating value is the new value to be transmitted. Small values increase control accuracy but also the bus load.

Designation	Values	Description
Send cyclically	***do not send cyclically*** *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?

4.8.7 Cooling control

Designation	Values	Description
Type of control	***Continuous*** *2-point*	Infinite control (0.. 100 %). Switching control (On/Off). See in the Appendix: *Continuous and switching control*.
Number of cooling stages	***Only one cooling stage*** *Main stage and additional stage*	Choice of 1- or 2-stage cooling
Hysteresis of 2-point controller	*0.3 K, 0.5 K, 0.7 K* **1 K**, *1.5 K*	Interval between the tripping point (setpoint) and the turn back on point (setpoint – hysteresis). The hysteresis prevents a permanent switching on/off.
Recirculation of hysteresis after switching point	***None*** *0.1 K/min* *0.2 K/min* *0.3 K/min*	The recirculation causes a gradual decrease in the hysteresis over time, and the control accuracy is increased. The hysteresis is equivalent to the programmed value for each switch-off and is gradually reduced by the recirculation process. The hysteresis can reduce to 0 K over prolonged periods of switch-off. When switching on the next time, it will be reset to the configured value.
Setting the control parameters	***Via installation type*** *User-defined*	Standard application. The control parameters are preset. Professional use: Configure P/PI controller yourself.
Installation type	***Cooling surface*** *Fan coil unit*	PI controller with: Integrated time = 240 minutes Bandwidth = 5 K Integrated time = 180 min. Bandwidth = 4 K
Proportional band of the cooling control	*1 K, 1.5 K, 2 K, 2.5 K* *3 K, 3.5 K, 4 K, 4.5 K* **5 K**, *5.5 K, 6 K, 6.5 K* *7 K, 7.5 K, 8 K, 8.5 K*	Professional setting for adapting control response to the room. Small values cause large changes in actuating values, larger values cause a finer actuating value adjustment. See in the Appendix: *Temperature control*

Designation	Values	Description
Integrated time of the cooling control	*30 min, 60 min,* **90 min**, *120 min* *150 min, 180 min* *210 min* *4 h, 5 h, 10 h* *15 h, 20 h, 25 h* *30 h, 35 h*	Professional setting: See in the Appendix: *Response of the PI controller* This time can be adapted to suit particular circumstances. If the cooling system is over-dimensioned and therefore too fast, shorter values should be used. On the other side, longer integration times are beneficial for a slightly undersized cooling (slow).
Send cooling actuating value	*at change by 1%* *at change by 2%* *at change by 3%* ***at change by 5%*** *at change by 7%* *at change by 10%* *at change by 15%*	After what percentage change in the actuating value is the new value to be transmitted. Small values increase control accuracy but also the bus load.
Switching between heating and cooling	***Automatic*** *Via object*	The controller automatically switches to cooling mode when the actual temperature is above the setpoint. The cooling mode can only be activated on the bus via object *Change over between heating and cooling*. Cooling mode remains off for as long as this object is not set.
Format object heating/cooling	***DPT1.100 (Heating=1/Cooling=0)*** *Inverted (Heating=0/Cooling=1)*	Standard format. Compatible with RAM 713 S, VARIA etc.
Output of the cooling actuating value	***On separate obj. (for 4-pipe systems)*** *In common with actuating val. heating (for 2-pipe systems)*	For 4-pipe systems: The actuating values are sent to 2 separate objects:a Obj. *heating actuating value* Obj. *cooling actuating value*. For 2-pipe systems: The actuating value is always sent to the same object (obj. *actuating value heating/cooling*), independent of whether heating or cooling mode is active.
Send cooling actuating value cyclically	***do not send cyclically*** *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?

4.8.8 Cooling setpoints

Designation	Values	Description
Dead zone between heating and cooling	*0 K*[39]*, 0.5 K*[40]*, 1 K, 1.5 K,* **2 K**, *2.5 K, 3 K, 3.5 K* *4 K, 4.5 K, 5 K, 5.5 K, 6 K* *+ hysteresis heating*[41] *+ hysteresis cooling*[42]	Specifies the buffer zone between setpoints for heating and cooling mode. The dead zone is expanded through hysteresis in switching (2 point) control. See in the Appendix: *Dead zone*
Increase in standby mode (during cooling)	*0 K, 0.5 K, 1 K* *1.5 K,* **2 K**, *2.5 K* *3 K, 3.5 K, 4 K* *4.5 K, 5 K*	The standby temperature is increased in cooling mode.
Increase in night mode (during cooling)	*3 K, 4 K,* **5 K** *6 K, 7 K, 8 K*	The in cooling mode, the temperature is increased in night mode.
Setpoint for heat protection mode (during cooling)	***0 = 42 °C, i.e. no real heat protection*** *29 °C, 30 °C, 31 °C* *32 °C, 33 °C, 34 °C* *35 °C*	Heat protection represents the maximum permitted temperature for the controlled room. It performs the same function during cooling as the frost protection mode during heating, e.g. saves energy while prohibiting non-permitted temperatures.

[39] 0 K and 0.5 K: Only in 2-pipe system.
[40] 0 K and 0.5 K: Only in 2-pipe system.
[41] Only with type of control heating = 2-point.
[42] Only with type of control cooling = 2-point.

4.8.9 Set point adjustment

Designation	Values	Description
Set point correction from	**25 °C**, *26 °C, 27 °C, 28 °C* *29 °C, 30 °C, 31 °C, 32 °C* *33 °C, 34 °C, 35 °C, 36 °C* *37 °C, 38 °C , 39 °C, 40 °C*	Activation threshold for set point correction.
Adjustment	*1 K per 1 K outdoor temperature* *1 K per 2 K outdoor temperature* **1 K per 3 K outdoor temperature** *1 K per 4 K outdoor temperature* *1 K per 5 K outdoor temperature* *1 K per 6 K outdoor temperature* *1 K per 7 K outdoor temperature*	Strength of the set point correction: At which change of the outdoor temperature should the setpoint be adjusted by 1 K?
Set point adjustment format	***relative***	The *Outdoor temperature compensation* object sends a temperature difference in K, in dependence on the outdoor temperature. This value can be used as a set point offset for additional room thermostats.
	absolute	The *Outdoor temperature compensation* object sends a setpoint in °C (base setpoint without correction). This is increased in relation to the outdoor temperature and serves as setpoint for additional temperature controllers.
Start setpoint	*15 °C-30 °C* Std.: ***21 °C***	(Only with format = absolute). This is the base setpoint for the external controller. If correction is required, it is added to this and the result is sent as a newer, adjusted setpoint
Maximum adjustment	*Unlimited*[43]	The setpoint continues to increase as long as the outside temperature increases.

[43] *In case of set point correction at high temperatures = calculate internally and send.*

Designation	Values	Description
	Until heat protection temp. reached [44] *+3 K* *+5 K* *+7 K*	The setpoint is only increased up to the configured heat protection temperature. The setpoint increase ends as soon as the adjustment has achieved the set value.
Send set point adjustment	***do not send cyclically*** *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?

[44] *In case of set point correction at high temperatures = only receive.*

4.8.10 Additional stage cooling

Control is done via a proportional controller.

Designation	Values	Description
Type of actuating value		Control is done via a proportional controller.
	Percent	Continuous actuating value 0-100 %
	PWM	Pulse-width modulated switching actuating value.
Difference between main stage and additional stage	*0 K, 0.5 K, 1 K* *1.5 K,* **2 K**, *2.5 K* *3 K, 3.5 K, 4 K*	Defines the negative distance between the current setpoint and the setpoint of the additional stage. **Example** with basic setpoint of 21 °C and difference of 1 K: The main stage controls with the base setpoint and the addition stage controls with base setpoint value − 1K = 20°C
Proportional band	*1 K, 1.5 K, 2 K, 2.5 K* *3 K, 3.5 K,* **4 K**, *4.5 K* *5 K, 5.5 K, 6 K, 6.5 K* *7 K, 7.5 K, 8 K, 8.5 K*	With continuous additional stage, Professional setting for adapting control response to the room. Large values cause finer changes to the control variables with the same control deviation and more precise control than smaller values.
PWM period	*3-30 min* Std.: ***5 min***	An actuation cycle consists of a switching-on and a switching-off process and forms a PWM period. **Example:** Actuating value = 20 %, PWM time = 10 min: In an actuating cycle of 10 min, 2 min switched on and 8 min switched off (i.e. 20 % on/80 % off).
Transmission of actuating value	*At change by 1%* *At change by 2%* *At change by 3%* ***At change by 5%*** *At change by 7%* *At change by 10%* *At change by 15%*	After what percentage change in the actuating value is the new value to be transmitted. Small values increase control accuracy but also the bus load.
Send cyclically	***do not send cyclically*** *every 2 min* *every 3 min* … *every 45 min* *every 60 min*	How often should it be resent?

4.9 External inputs I1-I4 functional block

4.9.1 Switch function

Designation	Values	Description
Activate channel	***no*** *yes*	Use input?
Channel function	***Switch..*** *Push button..* *Dimming..* *Blinds..*	Sends, depending on whether the input is 0 or 1.
Debounce time	*30 ms,* ***50 ms,*** *80 ms* *100 ms, 200 ms,* *1 s , 5 s, 10 s*	In order to avoid a disruptive switching due to debouncing of the contact connected to the input, the new status of the input is only accepted after a delay time. Larger values (≥ 1s) can be used as a switch-on delay
Send cyclically	*every min* *every 2 min* *every 3 min* ... ***every 30 min*** *every 45 min* *every 60 min*	Common cycle time for all 3 initial objects of the channel.
Number of telegrams	***one telegram*** *two telegrams* *three telegrams*	Each channel has 3 initial objects and can thus send up to 3 different telegrams.
Activate block function	***no*** *yes*	No block function. Show block function parameter page.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable

4.9.1.1 Switch object parameter pages 1, 2, 3

Each of the 3 objects can be configured individually on its own parameter page.

Designation	Values	Description	
Object type	***Switching (1 bit)*** *Priority (2 bit)* *Value 0-255* *Percentage value (1 byte)* *2 byte floating-point number DPT 9.x* *4 byte floating-point number DPT 14.x*	Telegram type for this object.	
Send if input = 1	*no* ***yes***	Send if voltage is present at the input?	
Telegram	*With object type = switching 1 bit*		
	ON *OFF* *BY*	Send switch-on command Send switch-off command Invert current state (ON-OFF-ON etc.)	
	With object type = priority 2 bit		
		Function	Value
	inactive	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = value 0-255		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = percentage value 1 byte		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	
	With object type = 2 byte floating-point number		
	-670760...670760 Std.: ***0***	Any value between -670760 and 670760 can be sent.	
	With object type = 4 byte floating-point number		
	-1E+38.. 1E+38 Std.: ***0***	Any value between -1E+38 and 1E+38 can be sent. Input format: The ETS only allows the input as a decimal without power. **Example:** 15234825.123456	
Send if input = 0	*no* ***yes***	Send if voltage is present at the input?	
Telegram	See above: Same object type as *Send if input = 1*		

Designation	Values	Description
Send cyclically	***no*** *yes, always* *only if input = 1* *only if input = 0*	When should be sent cyclically? The cycle time is set on the main parameter page of the channel.
Response after restoration of the bus supply	***none*** *update (immediately)* *update (after 5 s)* *update (after 10 s)* *update (after 15 s)*	Do not send. Send update telegram immediately or with delay.
Response when setting the block	***Ignore block*** *no response* *as with input = 1* *as with input = 0*	The block function is ineffective with this telegram. Do not respond when setting the block. Respond as with rising edge. Respond as with falling edge.
Response when cancelling the block	***no response*** *update*	Do not respond when the block is cancelled. Send update telegram.

ⓘ If a channel is blocked, no telegrams will be sent cyclically.

4.9.2 Switch function I1, I2, I3, I4

Designation	Values	Description
Activate channel	***no*** *yes*	Use input?
Channel function	*Switch..* ***Push button..*** *Dimming..* *Blinds..*	A push button is connected to the input.
Debounce time	*30 ms,* ***50 ms,*** *80 ms* *100 ms, 200 ms,* *1 s , 5 s, 10 s*	In order to avoid a disruptive switching due to debouncing of the contact connected to the input, the new status of the input is only accepted after a delay time. Larger values (≥ 1s) can be used as a switch-on delay.
Connected push button	***NO contact*** *Opening contact*	Set the Type of connected contact.
Long button push starting at	***300 ms****, 400 ms* *500 ms, 600 ms* *700 ms, 800 ms* *900 ms, 1 s*	Serves to clearly differentiate between long and short button push. If the push button is pressed for at least as long as the set time, then a long button push will be registered.
Time for double-click	***300 ms****, 400 ms* *500 ms, 600 ms* *700 ms, 800 ms* *900 ms, 1 s*	Serves to differentiate between a double-click and 2 single clicks. Time period in which the second click must begin, in order to recognise a double-click.
Send cyclically	*every min* *every 2 min* *every 3 min* *…* ***every 30 min*** *every 45 min* *every 60 min*	Common cycle time for all 3 initial objects of the channel.
Number of telegrams	***one telegram*** *two telegrams* *three telegrams*	Each channel has 3 initial objects and can thus send up to 3 different telegrams.
Activate block function	***no*** *yes*	No block function. Show block function parameter page.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable

4.9.2.1 Parameter pages button object 1, 2, 3

Each of the 3 objects can be configured individually on its own parameter page.

Designation	Values	Description	
Object type	***Switching (1 bit)*** *Priority (2 bit)* *Value 0-255* *Percentage value (1 byte)* *2 byte floating-point number DPT 9.x* *4 byte floating-point number DPT 14.x*	Telegram type for this object.	
Send after short operation	***do not send*** *Send telegram*	Respond to short button push?	
Telegram	*With object type = switching 1 bit*		
	ON *OFF* *BY*	Send switch-on command Send switch-off command Invert current state (ON-OFF-ON etc.)	
	With object type = priority 2 bit		
		Function	Value
	inactive	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = value 0-255		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = percentage value 1 byte		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	
	With object type = 2 byte floating-point number		
	-670760...670760 Std.: ***0***	Any value between -670760 and 670760 can be sent.	
	With object type = 4 byte floating-point number		
	-1E+38.. 1E+38 Std.: ***0***	Any value between -1E+38 and 1E+38 can be sent. Input format: The ETS only allows the input as a decimal without power. **Example:** 15234825.123456	
Send after long operation	***do not send*** *Send telegram*	Respond to long button push?	
Telegram	See above: Same object type as with short operation.		
Send after double-click	***do not send*** *Send telegram*	Respond to double-click?	

Designation	Values	Description
Telegram	See above: Same object type as with short operation.	
Send cyclically	***no*** *yes*	The cycle time is set on the main parameter page of the channel.
Response after restoration of the bus supply	***none*** *As with short (immediately)* *As with short (after 5 s)* *As with short (after 10 s)* *As with short (after 15 s)* *As with long (immediately)* *As with long (after 5 s)* *As with long (after 10 s)* *As with long (after 15 s)* *As with double-click (immediately)* *As with double-click (after 5 s)* *As with double-click (after 10 s)* *As with double-click (after 15 s)*	Do not send. Send update telegram immediately or with delay. The value to be sent depends on the value configured for long, short button push, or double-click.
Response when setting the block	***Ignore block*** *no response* *as with short* *as with long* *as with double-click*	The block function is ineffective with this telegram. Do not respond when setting the block. Respond as with a short button push. Respond as with a long button push. Respond as with a double-click.
Response when cancelling the block	***no response*** *as with short* *as with long* *as with double-click*	Do not respond when the block is cancelled. Respond as with a short button push. Respond as with a long button push. Respond as with a double-click.

Note: If a channel is blocked, no telegrams will be sent cyclically.

4.9.3 Dimming function I1, I2, I3, I4

Designation	Values	Description
Activate channel	***no*** *yes*	Use input?
Channel function	*Switch..* *Push button..* ***Dimming..*** *Blinds..*	The input controls a dimming actuator,
Debounce time	*30 ms,* ***50 ms,*** *80 ms* *100 ms, 200 ms,* *1 s , 5 s, 10 s*	In order to avoid a disruptive switching due to debouncing of the contact connected to the input, the new status of the input is only accepted after a delay time. Larger values (≥ 1s) can be used as a switch-on delay
Long button push starting at	***300 ms****, 400 ms* *500 ms, 600 ms* *700 ms, 800 ms* *900 ms, 1 s*	Serves to clearly differentiate between long and short button push. If the push button is pressed for at least as long as the set time, then a long button push will be registered.
Double-click additional function	***no*** *yes*	No double-click function The double-click parameter page is shown.
Time for double-click	***300 ms****, 400 ms* *500 ms, 600 ms* *700 ms, 800 ms* *900 ms, 1 s*	Serves to differentiate between a double-click and 2 single clicks. Time period in which the second click must begin, in order to recognise a double-click.
Activate block function	***no*** *yes*	No block function. Show block function parameter page.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable

4.9.3.1 Dimming parameter page

Designation	Values	Description
Response to "long" / "short"		The input distinguishes between a long and a short button push, and can thus carry out 2 functions.
	One button operation	The dimmer is operated with a single push button. Short button push = ON/OFF Long button push = brighter/darker release = stop With the other variants, the dimmer is operated using 2 buttons (rocker).
	brighter/ON	Short button push = ON Long button push = brighter Release = stop
	brighter/BY	Short button push = ON/OFF Long button push = brighter Release = stop
	darker/OFF	Short button push = OFF Long button push = darker Release = stop
	darker/BY	Short button push = ON/OFF Long button push = darker Release = stop
Increment for dimming		With a long button push, the dimming value is:
	100%	Increased (or decreased) until the button is released.
	50% *25%* *12.5%* *6%* *3%* *1.5%*	Increased by the selected value (or reduced)
Response in case of bus and mains restoration	***none***	Do not react.
	ON	Switch on dimmer
	OFF	Switch off dimmer

Designation	Values	Description
	after 5 s ON *after 10 s ON* *after 15 s ON* *after 5 s OFF* *after 10 s OFF* *after 15 s OFF*	Switch on dimmer with delay Switch off dimmer with delay
Response when setting the block	***Ignore block***	The block function is ineffective with this telegram.
	no response	Do not respond when setting the block.
	ON	Switch on dimmer
	OFF	Switch off dimmer
Response when cancelling the block	***no response***	Do not respond when the block is cancelled.
	ON	Switch on dimmer
	OFF	Switch off dimmer

4.9.3.2 Double-click parameter page

Designation	Values	Description	
Object type	***Switching (1 bit)*** *Priority (2 bit)* *Value 0-255* *Percentage value (1 byte)* *2 byte floating-point number* *DPT 9.x* *4 byte floating-point number* *DPT 14.x*	Telegram type for this object.	
Telegram	*With object type = switching 1 bit*		
	ON *OFF* *BY*	Send switch-on command Send switch-off command Invert current state (ON-OFF-ON etc.)	
	With object type = priority 2 bit		
		Function	Value
	inactive	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = value 0-255		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = percentage value *1 byte*		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	
	With object type = 2 byte floating-point number		
	-670760...670760 Std.: ***0***	Any value between -670760 and 670760 can be sent.	
	With object type = 4 byte floating-point number		
	-1E+38.. 1E+38 Std.: ***0***	Any value between -1E+38 and 1E+38 can be sent. Input format: The ETS only allows the input as a decimal without power. **Example:** 15234825.123456	
Send cyclically	***do not send cyclically*** *every min* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be sent again?	
Response after restoration of the bus	***none***	Do not send.	

Designation	Values	Description
supply	*As with double-click (immediately)* *As with double-click (after 5 s)* *As with double-click (after 10 s)* *As with double-click (after 15 s)*	Send update telegram immediately or with delay. The value to be sent depends on the value configured for double-click.
Response when setting the block	***Ignore block*** *no response* *as with double-click*	The block function is ineffective with this telegram. Do not respond when setting the block. Respond as with a double-click.
Response when cancelling the block	***no response*** *as with double-click*	Do not respond when the block is cancelled. Respond as with a double-click.

4.9.4 Blinds function I1, I2, I3, I4

Designation	Values	Description
Activate channel	***no*** *yes*	Use input?
Channel function	*Switch..* *Push button..* *Dimming..* ***Blinds..***	The input controls a blinds actuator.
Debounce time	*30 ms,* ***50 ms,*** *80 ms* *100 ms, 200 ms,* *1 s , 5 s, 10 s*	In order to avoid a disruptive switching due to debouncing of the contact connected to the input, the new status of the input is only accepted after a delay time. Larger values (≥ 1s) can be used as a switch-on delay
Long button push starting at	***300 ms****, 400 ms* *500 ms, 600 ms* *700 ms, 800 ms* *900 ms, 1 s*	Serves to clearly differentiate between long and short button push. If the push button is pressed for at least as long as the set time, then a long button push will be registered.
Double-click additional function	*no* ***yes***	No double-click function The double-click parameter page is shown.
Time for double-click	***300 ms****, 400 ms* *500 ms, 600 ms* *700 ms, 800 ms* *900 ms, 1 s*	Serves to differentiate between a double-click and 2 single clicks. Time period in which the second click must begin, in order to recognise a double-click.
Activate block function	***no*** *yes*	No block function. Show block function parameter page.
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = enable 1 = block 0 = block 1 = enable

4.9.4.1 Blinds parameter page

Designation	Values	Description
Operation		The input distinguishes between a long and a short button push, and can thus carry out 2 functions.
	One button operation	The blinds are operated with a single push button. Short button push = Step. Long button push = Move.
	DOWN	Short button push = Step. Long button push = lowering.
	UP	Short button push = Step. Long button push = raising.
Movement is stopped by	*releasing the button* ***Short operation***	How is the stop command to be triggered?
Response in case of bus and mains restoration	***none***	Do not react.
	UP	Raise the blind
	DOWN	Lower blinds
	after 5 s UP *after 10 s UP* *after 15 s UP*	Raise blinds with delay
	after 5 s DOWN *after 10 s DOWN* *after 15 s DOWN*	Lower blinds with delay
Response when setting the block	***Ignore block***	The block function is ineffective with this telegram.
	no response	Do not respond when setting the block.
	UP	Raise the blind
	DOWN	Lower blinds
Response when cancelling the block	***no response***	Do not respond when the block is cancelled.
	ON	Raise the blind
	OFF	Lower blinds

4.9.4.2 Double-click parameter page

Designation	Values	Description	
Object type	***Switching (1 bit)*** *Priority (2 bit)* *Value 0-255* *Percentage value (1 byte)* *2 byte floating-point number* *DPT 9.x* *4 byte floating-point number* *DPT 14.x*	Telegram type for this object.	
Telegram	*With object type = switching 1 bit*		
	ON *OFF* *BY*	Send switch-on command Send switch-off command Invert current state (ON-OFF-ON etc.)	
	With object type = priority 2 bit		
		Function	Value
	inactive	Priority not active (no control)	0 (00_{bin})
	ON	Priority ON Priority ON (control: enable, on)	3 (11_{bin})
	OFF	Priority OFF (control: disable, off)	2 (10_{bin})
	With object type = value 0-255		
	*0-**255***	Any value between 0 and 255 can be sent.	
	With object type = percentage value *1 byte*		
	*0-**100%***	Any percentage value between 0 and 100 % can be sent.	
	With object type = 2 byte floating-point number		
	-670760...670760 Std.: ***0***	Any value between -670760 and 670760 can be sent.	
	With object type = 4 byte floating-point number		
	-1E+38.. 1E+38 Std.: ***0***	Any value between -1E+38 and 1E+38 can be sent. Input format: The ETS only allows the input as a decimal without power. **Example:** 15234825.123456	
Send cyclically	***do not send cyclically*** *every min* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be sent again?	
Response after restoration of the bus	***none***	Do not send.	

Designation	Values	Description
supply	*As with double-click (immediately)* *As with double-click (after 5 s)* *As with double-click (after 10 s)* *As with double-click (after 15 s)*	Send update telegram immediately or with delay. The value to be sent depends on the value configured for double-click.
Response when setting the block	***Ignore block*** *no response* *as with double-click*	The block function is ineffective with this telegram. Do not respond when setting the block. Respond as with a double-click.
Response when cancelling the block	***no response*** *as with double-click*	Do not respond when the block is cancelled. Respond as with a double-click.

4.9.5 Temperature sensor function (only I3 and I4)

The external inputs I3 and I4 can be used as analogue inputs for temperature measurement via remote sensor.

This function is activated on the ***General*** parameter page with the parameter *function of the external inputs I3 + I4*.

The temperature measured at I3 can be used internally as an actual value for the RTC (see *Source for actual value* parameter).

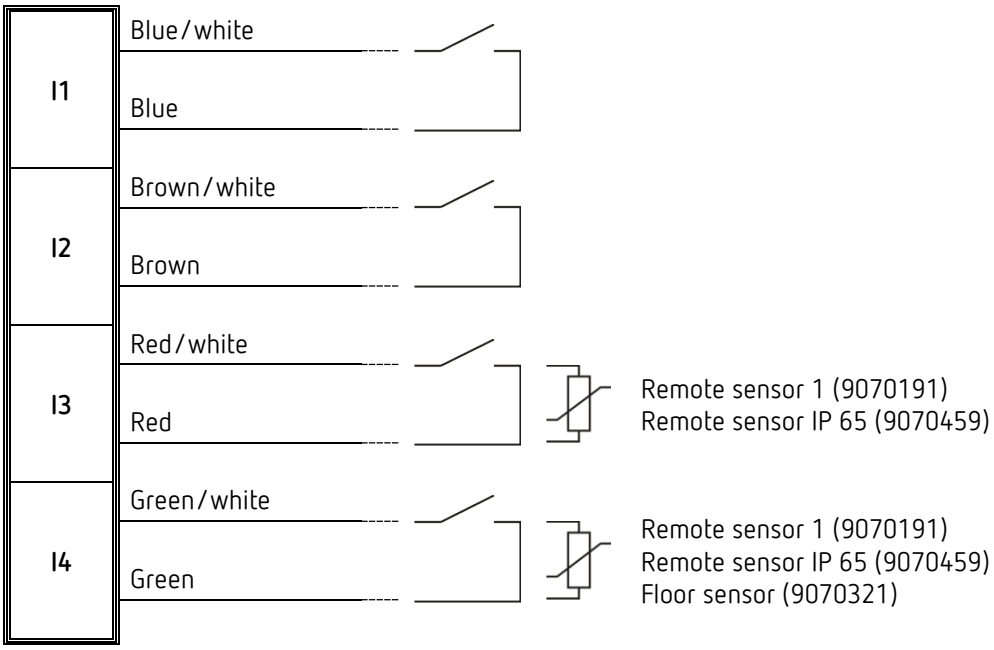
The temperature measured at I4 can be used internally as a floor temperature for the RTC. See parameter *Use floor temperature limitation (sensor at I4)* on the ***Settings*** parameter page).

Notwithstanding the above, both measurement values can also be sent to the bus.

Designation	Values	Description
Activate channel	***no*** *yes*	Use input?
Sensor type	***Remote sensor 1 (9070191)*** *Remote sensor IP 65 (9070459)* *Floor sensor (9070321)*	External temperature sensor 1 Item no. 9070191, for surface-mounted installation. External temperature sensor RAMSES IP65 Item no. 9070459, for surface-mounted installation. Only at input I4: Temperature sensor for laying in floor, IP65 protection rating.
Temperature calibration	*-64..+64* (x 0.1 K)	Correction value for temperature measurement if sent temperature deviates from the actual ambient temperature. **Example:** Temperature = 20°C sent temperature = 21°C Correction value = 10 (d.h. 10 x 0.1°C)
Transmit temperature in the event of change of	*not due to a change* *0.2 K* *0.3 K* ***0.5 K*** *0.7 K* *1 K* *1.5 K* *2 K*	Only send cyclically (if enabled) Send if the value has changed by the selected amount since the last transmission.

Designation	Values	Description
Send temperature cyclically	***do not send cyclically*** *every min,* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should the current measured value be resent?

4.9.6 Connection of the external inputs



Contact voltage: 5V SELV
Contact current: 0.5 mA (mean value), 5 mA (peak value)

Only connect floating contacts or Theben temperature sensors.

CAUTION! Observe extra-low voltage/distances!

4.10 Comparator functional block

This function can be used to compare different values.
The resulting final value is sent out via object.
For each of the three inputs, a control value (CO_2, humidity and RTC) or an input object (DPT5.1) can be configured.

This function is available from application programme version 3.1.
Only for devices manufactured as of 2113, firmware version 3.0.1 (18 01).

Designation	Values	Description
Output sends	***Minimum value***	Send only the lowest value of all inputs.
	Mean value	Determine and send the mean value of all inputs.
	Maximum value	Send only the highest value of all inputs.
Input 1		First comparison value
	inactive	Do not use.
	Input object (0-100%)	Receive external value via object.
	Actuating value – RTC[45] *Actuating value – ventilation CO2* *Actuating value – ventilation of humidity*	Use current value of the selected actuating value (internally connected).
Input 2		Second comparison value.
	inactive	Do not use.
	Input object (0-100%)	Receive external value via object.
	Actuating value – RTC[46] **Actuating value – ventilation CO_2** *Actuating value – ventilation of humidity*	Use current value of the selected actuating value (internally connected).
Input 3		Third comparison value.

[45] Only when the RTC is active. See *Activate room temperature controller (RTC),* on parameter page ***General.***
[46] Only when the RTC is active. See *Activate room temperature controller (RTC),* on parameter page ***General.***

Designation	Values	Description
	inactive *Input object (0-100%)* *Actuating value – RTC* [47] *Actuating value – ventilation CO2* ***Actuating value – ventilation of humidity***	Do not use. Receive external value via object. Use current value of the selected actuating value (internally connected).
Send in cycles	***Do not send in cycles*** *every min* *every 2 min* *every 3 min* ... *every 45 min* *every 60 min*	How often should it be resent?
Send actuating value on change	***not due to a change*** *1%* *5%* *10%* 15% 20%	only send cyclically (if enabled) Send if the value has changed by the selected amount since the last transmission.
Response when setting the block	***Ignore block*** *do not send* *Send following value*	No block function, no further block parameters. Response to block telegram.
Value [48]	*0-100%*	Value to be sent in lock mode.
Response when cancelling the block	*do not send* ***update***	No response. send the current actuating value (comparison value).
Block telegram	***Block with 1 (standard)*** *Block with 0*	0 = cancel block 1 = block 0 = block 1 = cancel block

[47] Only when the RTC is active. See *Activate room temperature controller (RTC),* on parameter page ***General.***
[48] Only available if *Response when setting the block = Send following value*.

5 Typical applications

These typical applications are designed to aid planning and are not to be considered an exhaustive list.
It can be extended and updated as required.

5.1 Control of air quality plus 3 stage manual fan control.

A fan is to provide fresh air if the CO_2 content exceeds the set thresholds.

Additional manual control:
There is a choice of 3 manual fan stages (forced operation).
For manual operation, a 4-way button is connected to the external inputs I1-I4.

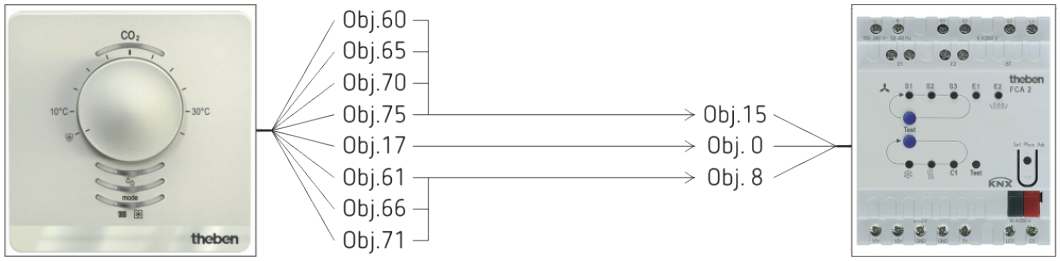
Button 1	Start forced stage 1
Button 2	Start forced stage 2
Button 3	Start forced stage 3
Button 4	Restore automatic operation

After reset or restoration of the bus supply, the fan operates in automatic mode, i.e. depending on CO_2 content.
If one of the buttons 1...3 is pressed, AMUN 716 S sends the forced command (1) and the desired fan stage to the FCA 2 actuator.
Forced operation is ended with button 4, and automatic operation will be restored.

5.1.1 Devices

- Amun 716 S (Order No. 7169230)
- FCA 2 (4920210)

5.1.2 Overview



5.1.3 Objects and links

No.	Amun 716 S Object name	No.	FCA 1 Object name	Comment
17	*Ventilation CO_2 – actuating value 0-100%*	0	*Actuating value for fan*	Fan control depending on CO_2 content
60	*Channel I1.1 – switching*	15	*Fan Forced/Auto*	Trigger forced operation with an ON telegram
61	*Channel I1.2 – send value*	8	*Fan stage in forced operation*	Specify manual stage 1
65	*Channel I2.1 – switching*	15	*Fan Forced/Auto*	Trigger forced operation with an ON telegram
66	*Channel I2.2 – send value*	8	*Fan stage in forced operation*	Specify manual stage 2
70	*Channel I3.1 – switching*	15	*Fan Forced/Auto*	Trigger forced operation with an ON telegram
71	*Channel I3.2 – send value*	8	*Fan stage in forced operation*	Specify manual stage 3
75	*Channel I4.1 – switching*	15	*Fan Forced/Auto*	Finish forced operation with an OFF telegram. Automatic operation will be restored.

5.1.4 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

Amun 716:

Parameter page	Parameter	Setting
CO_2 thresholds	*Number of CO_2 thresholds*	*3*
Ventilation of CO_2	*Fan control via*	*fixed values*
	If CO_2 below threshold 1	*0%*
	If CO_2 between threshold 1 and 2	*30%*
	If CO_2 between threshold 2 and 3	*70%*
	If CO_2 greater than threshold 3	*100%*
Channel I1	*Activate channel*	*ON*
	Channel function	*Push button*
	Number of telegrams	*Two telegrams*
Button object 1	*Object type*	*Switching (1 bit)*
	Send after short operation	*Send telegram*
	Telegram	*ON*
	Send after long operation	*do not send*
	Send after double-click	*do not send*
	Send cyclically	*No*
	Response after restoration of the bus supply	*None*
Button object 2	*Object type*	*Value 0-255*
	Send after short operation	*Send telegram*
	Telegram	*1*
	Send after long operation	*do not send*
	Send after double-click	*do not send*
	Send cyclically	*No*
	Response after restoration of the bus supply	*None*
Channel I2	All parameters:	as channel I1
Button object 1	All parameters:	as channel I1
Button object 2	*Telegram*	*2*
	All other parameters:	as channel I1
Channel I3	All parameters:	as channel I1
Button object 1	All parameters:	as channel I1
Button object 2	*Telegram*	*3*
	All other parameters:	as channel I1
Channel I4	*Activate channel*	*ON*
	Channel function	*Push button*
	Number of telegrams	*One telegram*
Button object 1	*Object type*	*Switching (1 bit)*
	Send after short operation	*Send telegram*
	Telegram	*OFF*
	Send after long operation	*do not send*
	Send after double-click	*do not send*
	Send cyclically	*No*
	Response after restoration of the bus supply	*None*

FCA 2:

Parameter page	Parameter	Setting
General	*Supported function*	*Ventilation*
	Switch fan between auto and forced	*via object auto/forced, forced = 1*
Fan	*Fan controller*	*Standard (1-3 stages)*
	Number of fan stages	*3 stages*
	Switch-on threshold for fan stage 1	*20%*
	Switch-on threshold for fan stage 2	*60%*
	Switch-on threshold for fan stage 3	*90%*
	Format compulsory control and limitation	*Fan stages (0 - 3)*

5.2 Controlling ventilation flaps depending on room air quality (CO2) or heating/cooling demand.

In modern buildings, the energy demand for room air conditioning is reduced due to optimum building insulation and highly insulating windows. At the same time, controlled room ventilation is essential in these buildings to ensure the hygienic and physical minimum air renewal. In combination, it is possible to provide room air conditioning also via the controlled room ventilation, making additional systems unnecessary.
The comparator integrated in the AMUN 716 S KNX offers optimum capabilities, to control e.g. ventilation flaps depending on room air quality (CO2) or heating/cooling demand.

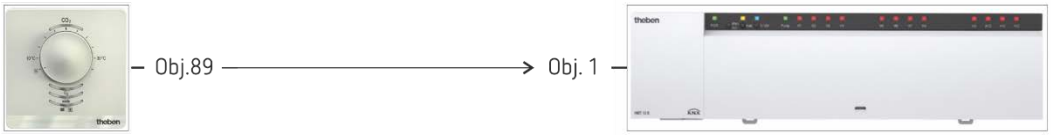
In a room, the ventilation flaps are to be controlled depending on CO2 and heating/cooling demand, i.e. the ventilation flaps open when fresh air is needed, or when heating or cooling energy is needed. The AMUN 716 S KNX collects and evaluates the physical quantities and sends the actuating value for the control of the ventilation flaps to a channel of the HMT 12 S KNX heating actuator. The actuator controls up to 12 ventilation flaps/rooms individually with a 0-10 V output signal.

ⓘ This function is available from application programme version 3.1. Only for devices from date of manufacture 2113, firmware version 3.0.1 (18 01).

5.2.1 Devices

- AMUN 716 S KNX (7169230)
- HMT 12 S KNX (4900374)

5.2.2 Overview



5.2.3 Objects and links

No.	AMUN 716 S Object name	No.	HMT 12 S Object name	Comment
89	Comparator - output	1	Channel H1 – continuous actuating value	Actuating value for ventilation flap control

5.2.4 Important parameter settings

Standards or customer-defined parameter settings apply to unlisted parameters.

AMUN 716 S:

Parameter page	Parameters	Setting
General	*Activate room temperature controller (RTC)*	*yes*
Ventilating CO_2	*Fan control via*	*PI controller*
	Setpoint	*800 ppm*
	Minimum actuating value	*20%*
	Response when falling below the minimum actuating value	Customer-specific Setting: *Output 0%* [49] or *Output minimum actuating value* [50]
Comparator	*Output sends*	*Maximum value*
	Input 1	*Actuating value – RTC*
	Input 2	*Actuating value – ventilation CO2*
	Input 3	*Inactive*

HMT 12 S:

Parameter page	Parameters	Setting
General	*Activate room temperature controller (RTC)*	*yes*
Channel H1		
Configuration options	*Type of valve control*	*0-10 V*

[49] If the actuating value is too low, always switch off the fan.
[50] The fan should not be at a complete standstill, but continue to run with the minimum actuating value, and thus ensure a permanent air exchange.

5.3 CO_2-dependent ventilation with PI control

A fan with infinitely variable speed control is to provide an air quality as constant as possible. The fan speed is precisely controlled by using a PI controller and a dimming actuator.

5.3.1 Devices

- Amun 716 S (Order No. 7169230)
- DM 2 T (Order No. 4940270)

5.3.2 Overview

5.3.3 Objects and links

No.	AMUN 716 S Object name	No.	DM 2 T Object name	Comment
17	Ventilation CO_2 – actuating value 0-100%	2	Channel C1 – dimming value	Actuating value for fan speed.

5.3.4 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

AMUN 716 S:

Parameter page	Parameter	Setting
Ventilating CO_2	*Fan control via*	*PI controller*
	Setpoint	*800 ppm*
	Minimum actuating value	*20%*
	Response when falling below the minimum actuating value	Customer-specific setting: *Output 0%* [51] or *Output minimum actuating value* [52]

DM 2 T:

Parameter page	Parameter	Setting
Dimming response	*Load selection*	*Fan (soft switching deactivated)*
	Start-up time	Customer-specific, depending on size of the fan.
	Minimum dimming value	*20%*
	Dimming time 1 from 0 to 100 %	*60 s*
	When receiving an absolute value (8 bit)	*Soft on with dimming time 1*

[51] If the actuating value is too low, always switch off the fan.
[52] The fan should not be at a complete standstill, but continue to run with the minimum actuating value, and thus ensure a permanent air exchange.

5.4 Base function: Humility-dependent ventilation, single-stage.

At a relative humidity above 75 %, the fan has to switch on.

5.4.1 Devices

- Amun 716 S (Order No. 7169230)
- RM 4 U (Order No. 4940223)

5.4.2 Overview

5.4.3 Objects and links

No.	AMUN 716 S Object name	No.	RM 4 U Object name	Comment
8	Humidity threshold 1 - switching	0	Channel C1 - switch object	Switch On/Off command.

5.4.4 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

AMUN 716:

Parameter page	Parameter	Setting
Humidity thresholds	*Number of humidity thresholds*	*1*
	Humidity threshold 1	*75%*

RM 4 U:

Parameter page	Parameter	Setting
Channel C1: Configuration options	*Channel function*	*switch On/Off*

5.5 Location school: Heating with presence detector and frost protection via window contact.

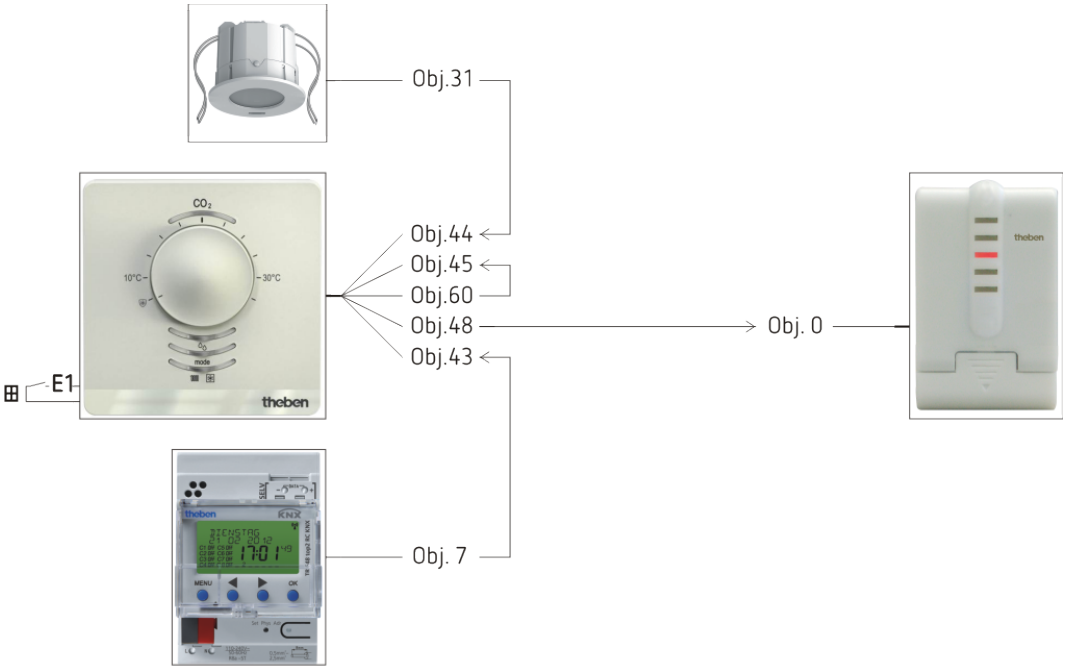
The room temperature controller (RTC) controls one or more actuators.
Once someone enters the room the controller has to change to comfort mode, otherwise it operates in standby mode during the day and in night mode at night.
If a window is opened, the controller has to automatically change to frost protection mode.
A presence detector is used for presence recognition.
The presence telegram is only sent after a switch-on delay so that the heating is not activated if the room is only occupied for a short time.

All windows are fitted with window contacts. These are connected with input E1 on the device. As an alternative, the external interface of the Cheops drive actuator can also be used for this.
The window status is sent via a common group address to the window position input object.
The device will recognise when a window is opened and automatically switch to frost protection mode.
When the window is closed, the previously set operating mode will be restored.

5.5.1 Devices

- Amun 716 S (Order No. 7169230)
- PlanoSpot 360 KNX (Order No. 2039100)
- TR 648 top2 RC KNX (Order No. 6489212)
- Cheops drive (Order No. 7319200)

5.5.2 Overview



5.5.3 Objects and links

No.	PlanoSpot 360 KNX Object name	No.	Amun 716 S Object name	Comment
31	Presence channel C4.1	44	Presence	Presence telegram. Triggers comfort mode.

No.	TR 648 top2 Object name	No.	Amun 716 S Object name	Comment
7	C1.1 switching channel – HVAC operating mode	43	Operating mode preset	Switches the controller between standby and night.

No.	Amun 716 S Object name	No.	Cheops drive Object name	Comment
48	Heating actuating value	0	Actuating value	Actuating value for actuator.

No.	Amun 716 S Object name	No.	Amun 716 S Object name	Comment
60	Channel I1.1 switching	45	Window status	The windows status is detected at input E1 (window contact) and sent to the controller (window status) via a group address. When opening the window, the controller changes into frost protection mode.

5.5.4 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

Amun 716 S:

Parameter page	Parameter	Setting
RTC setting	*Control*	*Heating control only*
	Rotary control function	*Blocked*
	Button function	*Blocked*
Operating Mode	*Objects for determining the operating mode*	*New: Operating mode, presence, window status*
	Type of presence sensor type (presence obj.)	*Presence detector*
Channel I1	*Activate channel*	*ON*
	Channel function	*Switch*
	Number of telegrams	*One telegram*
Switch object 1	*Object type*	*Switching (1 bit)*
	Send if input = 1	*yes*
	Telegram	*ON*
	Send if input = 0	*yes*
	Telegram	*OFF*
	Send cyclically	*yes*
	Response after restoration of the bus supply	*update (immediately)*

PlanoSpot 360 KNX:

Parameter page	Parameter	Setting
General	*Channel C4 – presence*	*active*
Channel C4 – presence	*Presence switch-on delay*	*5 min*
	Presence time delay	*10 min*

TR 648 top2 RC:

Parameter page	Parameter	Setting
General	*Activate time switch channel C1*	*yes*
Switching channel C1	*Telegram type C1.1*	*HVAC operating mode*
	As with clock -> ON	*send following telegram once*
	Telegram	*Standby*
	With clock -> OFF	*send following telegram once*
	Telegram	*Temperature reduction at night*

Cheops drive:

The standard values can be used here.

5.6 Location single-family house:

5.6.1 Heating with presence detector and frost protection via window contact.

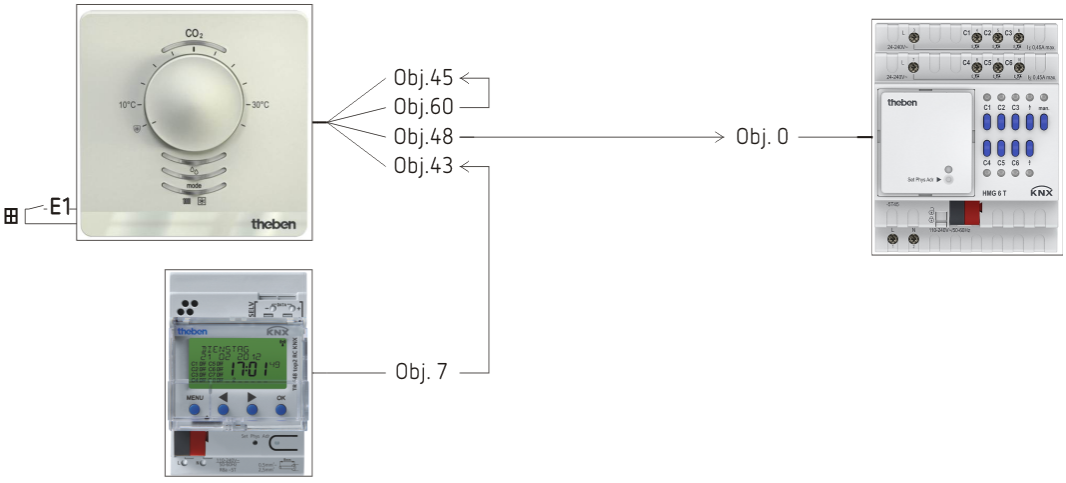
The room temperature controller (RTC) controls one or more actuators.
Comfort mode is triggered by other button at the device, otherwise the controller operates in standby mode during the day and in night mode at night.
If a window is opened, the controller has to automatically change to frost protection mode.

All windows are fitted with window contacts. These are connected with input E1 on the device.
The window status is sent via a common group address to the window position input object.
The device will recognise when a window is opened and automatically switch to frost protection mode. When the window is closed the previously set operation mode will be restored.

5.6.2 Devices

- Amun 716 S (Order No. 7169230)
- TR 648 top2 RC KNX (Order No. 6489212)
- HM 6 T (4940240)

5.6.3 Overview



5.6.4 Objects and links

No.	TR 648 top2 Object name	No.	Amun 716 S Object name	Comment
7	C1.1 switching channel – HVAC operating mode	43	Operating mode preset	Switches the controller between standby and night.

No.	Amun 716 S Object name	No.	HM 6 T Object name	Comment
48	Heating actuating value	0	Continuous actuating value	Actuating value for the heating actuator.

No.	Amun 716 S Object name	No.	Amun 716 S Object name	Comment
60	Channel I1.1 switching	45	Window status	The windows status is detected at input E1 (window contact) and sent to the controller (window status) via a group address. When opening the window, the controller changes into frost protection mode.

5.6.5 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

Amun 716 S:

Parameter page	Parameter	Setting
RTC setting	*Control*	*Heating control only*
	Rotary control function	*Manual offset*
	Button function	*Presence button*
Operating Mode	*Objects for determining the operating mode*	*New: Operating mode, presence, window status*
Channel I1	*Activate channel*	*ON*
	Channel function	*Switch*
	Number of telegrams	*One telegram*
Switch object 1	*Object type*	*Switching (1 bit)*
	Send if input = 1	*yes*
	Telegram	*ON*
	Send if input = 0	*yes*
	Telegram	*OFF*
	Send cyclically	*yes*
	Response after restoration of the bus supply	*update (immediately)*

TR 648 top2 RC:

Parameter page	Parameter	Setting
General	*Activate time switch channel C1*	*yes*
Switching channel C1	*Telegram type C1.1*	*HVAC operating mode*
	As with clock -> ON	*send following telegram once*
	Telegram	*Standby*
	With clock -> OFF	*send following telegram once*
	Telegram	*Temperature reduction at night*

HM 6 T:

Parameter page	Parameter	Setting
Channel H1: Configuration options	*Channel function*	*Heating actuator*
	Type of actuating value	*continuous..*

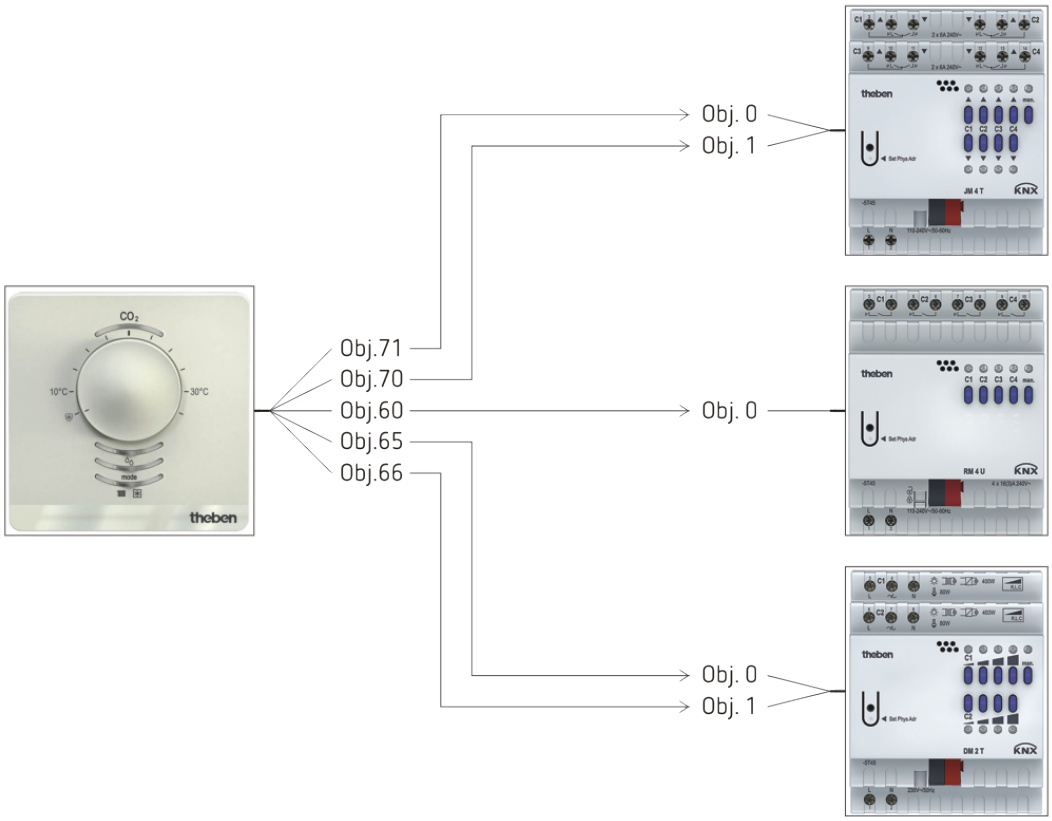
5.7 Switching, dimming light and controlling blinds

Via the external inputs, it is possible to simply control various actuators, such as switching, blinds, and dimming actuators, with conventional buttons.

5.7.1 Devices

- Amun 716 S (Order No. 7169230)
- RM 4 U (Order No. 4940223)
- DM 2 T (Order No. 4940270)
- JM 4 T (Order No. 4940250)

5.7.2 Overview



5.7.3 Objects and links

No.	Amun 716 S Object name	No.	RM 4 U Object name	Comment
60	Channel I1.1 – switching	0	Channel C1 – switch object	Switch command for the light.

No.	Amun 716 S Object name	No.	DM 2 T Object name	Comment
65	Channel I2 – switching	0	Switching ON/OFF	Switch command for the light.
66	Channel I2 – brighter/darker	1	Brighter/darker	4 bit dimming command

No.	Amun 716 S Object name	No.	JM 4 T Object name	Comment
70	Channel 3 – step/stop	1	Step/stop	Switch command for the light.
71	Channel I3 – up/down	0	Up/Down	1 bit operating command

5.7.4 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

Amun 716 S:

Parameter page	Parameter	Setting
Channel I1	*Activate channel*	*ON*
	Channel function	*Push button*
	Number of telegrams	*One telegram*
Button object 1	*Object type*	*Switching (1 bit)*
	Send after short operation	*Send telegram*
	Send after long operation	*do not send*
	Send after double-click	*do not send*
Channel I2	*Activate channel*	*ON*
	Channel function	*Dimming*
	Double-click additional function	*no*
Dimming	*Reaction to long/short*	*One button operation*
Channel I3	*Activate channel*	*ON*
	Channel function	*Blinds*
	Double-click additional function	*no*
Blinds	*Operation*	*One button operation*

RM 4 U:

Parameter page	Parameter	Setting
Channel C1: Configuration options	*Channel function*	*switch On/Off*

DM 2 T:

Parameter page	Parameter	Setting
Dimming response	*Load selection*	To be set system-specific.

JM 4 T

Parameter page	Parameter	Setting
Channel C1: Configuration options	*Type of motor*	To be set system-specific.
	Type of hanging	*Blinds..*
Drive settings	*Complete runtime down (s)*	To be set system-specific.
	Complete slat turning	To be set system-specific.

5.8 Two-stage heating for floor and radiators

A room is heated via the floor and additionally via radiators.
Both heating sources have very different requirements and are therefore controlled via 2 separate heating stages.
The first heating stage controls and limits the floor temperature (slow, inert heating).
The second heating stage controls one or several radiators (fast heating).

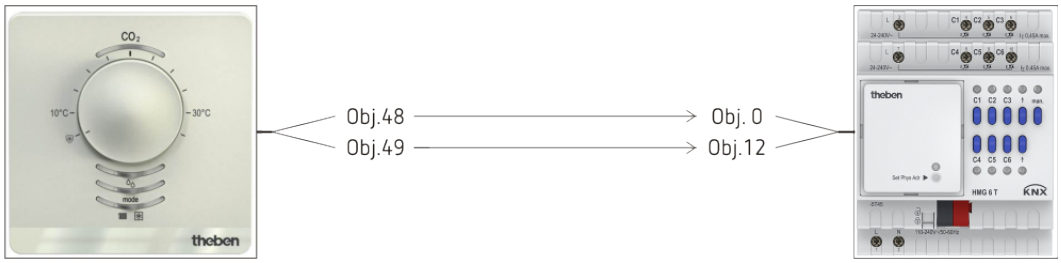
The floor temperature is measured by an external floor sensor (Order No. 907321) at input E4.

Here, the focus is on the 2 heating stages with floor temperature limitation.
The automatic change of the operating mode via time switch or presence detector, as well as the change of the operating mode and the frost protection function are not explicitly mentioned again (see previous examples).

5.8.1 Devices

- Amun 716 S (Order No. 7169230)
- HM 6 T (4940240)

5.8.2 Overview



5.8.3 Objects and links

No.	Amun 716 S Object name	No.	HM 6 T Object name	Comment
48	Heating actuating value	0	Channel H1 – continuous actuating value	Actuating value for underfloor heating
49	Actuating value additional heating stage	12	Channel H2 – continuous actuating value	Actuating value for the radiators

5.8.4 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

Amun 716 S:

Parameter page	Parameter	Setting
General	*Function of the external inputs I3 + I4*	*Temperature sensor input..*
Setting	*Control*	*Heating control only*
	Use floor temperature limitation (sensor at I4)	*yes*
Heating control	*Type of control*	*continuous*
	Number of heating stages	*Main stage and additional stage*
	Setting the control parameters	*Via installation type*
	Installation type	*Underfloor heating*
Heating setpoints	*Maximum floor temperature*	*e.g. 30 °C*
Additional stage heating	*Type of actuating value*	*Percent*
	Difference between main stage and additional stage	*0 K*
Channel I4	*Activate channel*	*ON*
	Sensor type	*Floor sensor (9070321)*

HM 6 T:

Parameter page	Parameter	Setting
Channel H1: Configuration options	*Channel function*	*Heating actuator*
	Type of actuating value	*continuous..*
Channel H2: Configuration options	*Channel function*	*Heating actuator*
	Type of actuating value	*continuous..*

6 Typical application of multi-sensor KNX

This application example is intended as a planning aid and does not claim to be complete.
It can be extended and updated as required.

6.1 Meeting room location: Automatic control of ventilation and lighting.

A fan should provide fresh air if the CO_2 content exceeds the set thresholds.
The lighting adapts automatically depending on natural daylight (constant lighting control). The lighting can also be switched and dimmed manually.

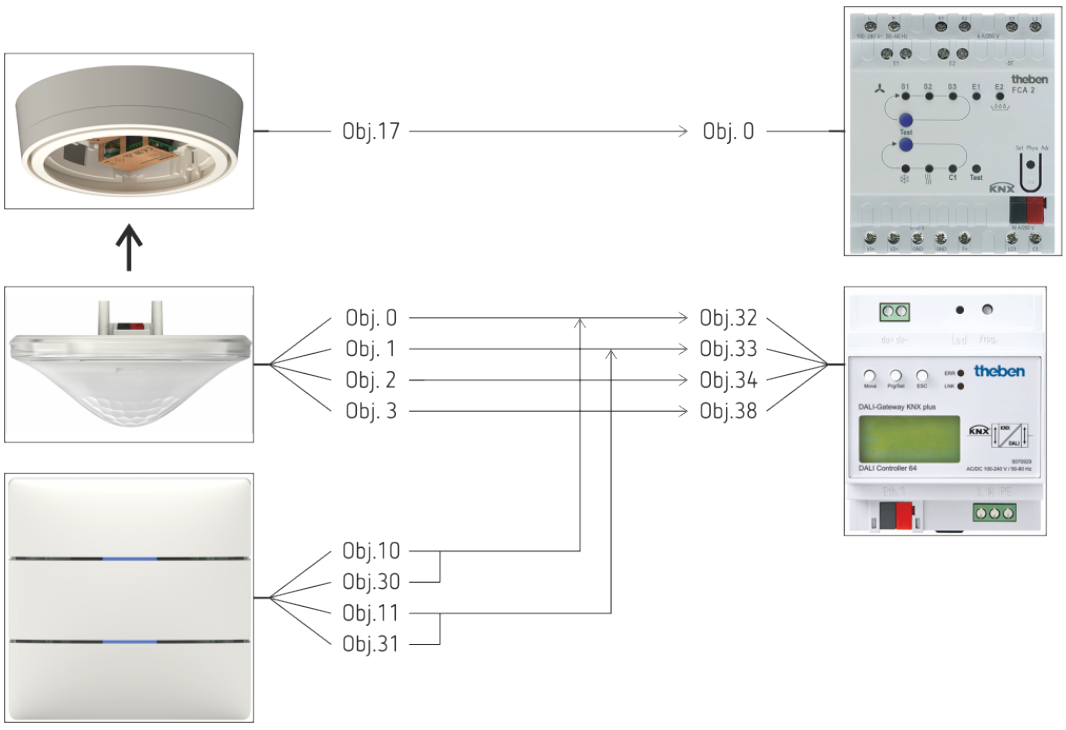
The measurements are taken in a device, which is mounted on the ceiling (multi-sensor KNX).

6.1.1 Devices

- Set basic KNX Multi (Order No. 9070900)
- thePrema P360 KNX (Order No. 2079000)
- FCA 2 (Order No. 4920210)
- DALI Gateway KNX plus (Order No. 9070929)
- iON 102 KNX (Order No. 4969232)

} Multi-sensor KNX

6.1.2 Overview



6.1.3 Objects and links

No.	Amun 716 S Object name	No.	FCA 2 Object name	Comment
17	*Ventilation CO_2 – actuating value 0-100%*	0	*Actuating value for fan*	Fan control depending on CO_2 content

No.	thePrema P360 KNX Object name	No.	DALI Gateway KNX plus Object name	Comment
0	*Channel C1 light / switching*	32	*Group 1 / switching*	DALI Gateway plus KNX from Version V2.0
1	*Channel C1 light / brighter/darker*	33	*Group 1 / dimming*	
2	*Channel C1 light / send value*	34	*Group 1 / set value*	
3	*Channel C1 light / feedback value*	38	*Group 1 / status value*	

No.	iON 102 KNX Object name	No.	DALI Gateway KNX plus Object name	Comment
10	*Button T1 switching*	32	*Group 1 / switching*	Switching on and off via push button
30	*Button T2 switching*			
11	*Button T1 brighter*	33	*Group 1 / dimming*	Dimming via push button
31	*Button T2 darker*			

6.1.4 Important parameter settings

Standard or customer-defined parameter settings apply to unlisted parameters.

Amun 716:

Parameter page	Parameter	Setting
General information	*Device type*	*Set basic KNX Multi (9070900)*
CO_2 thresholds	*Number of CO_2 thresholds*	*3*
Ventilation of CO_2	*Fan control via*	*fixed values*
	If CO_2 below threshold 1	*0%*
	If CO_2 between threshold 1 and 2	*30%*
	If CO_2 between threshold 2 and 3	*70%*
	If CO_2 greater than threshold 3	*100%*

thePrema P360:

Parameter page	Parameter	Setting
General information	*Operating mode*	*Master*
	Master operating mode	*Individual switching*
	Function channel C1 light	*Constant lighting control..*
Channel C1 - light	*Configuration type*	*Fully automatic device*
	Brightness setpoint value	*500 lx*
	Lighting time delay	*10 min (as per customer specification)*

DALI Gateway KNX plus (from Version V2.0):

Parameter page	Parameter	Setting
General information	*Operating mode*	*Normal operation*
	Function of additional object	*no Object*
	Panic mode enabled	*No*
Group 1		
Behaviour	*Switch-on value*	*100%*
	Switch-on behaviour	*Dim to value in 10 seconds*
	Switch-off value	*0%*
	Switch-off behaviour	*Apply value immediately*
	Behaviour on value setting	*Dim to value in 10 seconds*
	Time for dimming	*10 seconds*
	Max. value for dimming	*100%*
	Min. value for dimming	*0%*
	Switch-on via dimming	*No*

iON 102 KNX:

Parameter page	Parameter	Setting
Settings	*Device type*	*iON 102 KNX*
Button T1	*Function*	*Dimming*
	Response to long / short	*brighter / ON*
Button T2	*Function*	*Dimming*
	Response to long / short	*darker / OFF*

7 Appendix

7.1 CO_2 guide values

200 000		Fatal danger
100 000		Extinction of a candle flame
40 000...50 000		Concentration of the air we breathe (at exhalation)
5000		Occupational exposure limit (OEL)
4000		Badly ventilated room
1000		Sense of poor air quality
330...350		Outside air

All values in ppm (parts per million)

7.2 LED colours for room air quality and temperature control[53]

CO2	CO2	[value] < threshold 1
		Threshold 1 < [value] < threshold 2
		Threshold 2 < [value] < threshold 3
		Threshold 3 > [value]

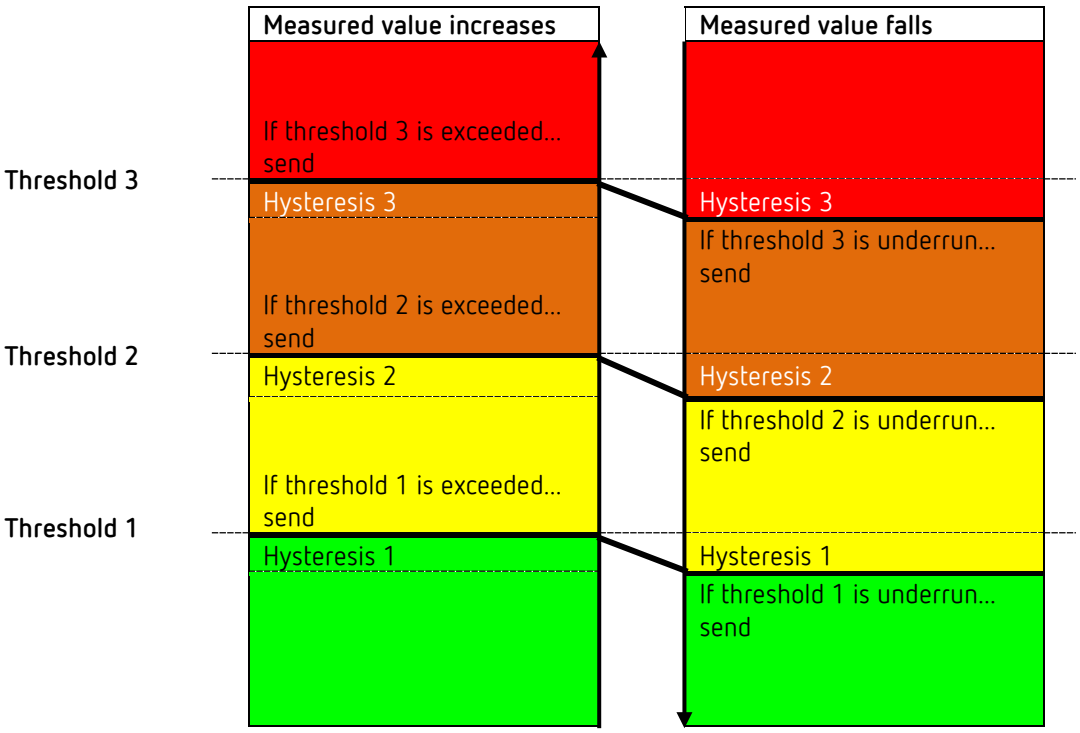
Relative humidity		[value] < threshold 1
		Threshold 1 < [value] < threshold 2
		Threshold 2 < [value] < threshold 3
		Threshold 3 > [value]

Operating Mode	mode	Frost
		Eco
		Standby
		Comfort

Status RTC		Heating
		Cooling

[53] Only Amun 716 S

7.3 Switching response using the example of thresholds for CO_2



The telegram of the last exceeded/underrun threshold is sent.
If several thresholds are exceeded from one measurement cycle to the next then the telegrams are sent at an increasing value (from thresholds 1-3) whereas with cyclical sending, only the telegram for the last exceeded threshold is sent cyclically. The same applies with falling values.

The switching behaviour is identical for the humidity thresholds, however, the colours[54] are different (see above).

[54] Only Amun 716 S

7.4 Fan control

Note the following for fan control using percentage values:

Amun sends a percentage value as the control variable for each threshold.
This control variable (in accordance with the set threshold) is transferred to the fan coil actuator as a fan stage between 0 and 3.

Important: The sent actuating value should always be a little higher than the threshold setting of the fan coil actuator.

Example:

Threshold for Fan stage	Set values for Amun 716 S	Recommended values for FCA 2
1	20%	10%
2	50%	40%
3	80%	70%

If fan stage 2 is selected using the button, the respective object (object 9 or 19) sends the actuating value 50%.
As the threshold for stage 2 in the fan coil actuator is set at 40%, the received control variable of 50% is clearly allocated to fan stage 2 and accepted by the fan.

7.5 Relative humidity

Relative humidity is a measurement for the saturation of air with water vapour.
This is expressed as the relationship to the maximum amount absorbed at the corresponding temperature.

Example: A relative humidity of 60% means that the air contains 60% of the maximum absorbable amount of water vapour.
At 100% the air is completely saturated and cannot absorb any more humidity.

Condensation or mist are produced if the volume of available water vapour exceeds this 100% threshold.

The ability of air to absorb water vapour depends on temperature.
Warm air can absorb more water vapour than cold air.

7.6 Fresh air calibration

If the device is properly used, a calibration procedure is not necessary.

A wrong calibration can cause malfunctions of the device and the connected systems.
Please use the calibration only if absolutely necessary.
Exhaling in close proximity of the device can distort the measurement.

With a fresh air calibration, the CO_2 sensor is calibrated, i.e. gauged to a new reference value. In 2013, an average concentration of 400 ppm was determined in the atmosphere. Normally, this value can be taken as a reference value.

7.6.1 Calibration procedure

First, the CO_2 sensor has to get as much fresh air as possible.
This is achieved either by sufficient ventilation (open all windows), or, if somehow possible, by moving the device into the fresh air for the entire duration of the calibration.
Set the *Enable fresh air calibration* parameter to *yes*, and download the application software with the ETS again.
For starting the calibration:

1. Send a switch on telegram to object 7 *Fresh air calibration*.
2. Within 2 h, press and hold the operating mode button for 5 s.

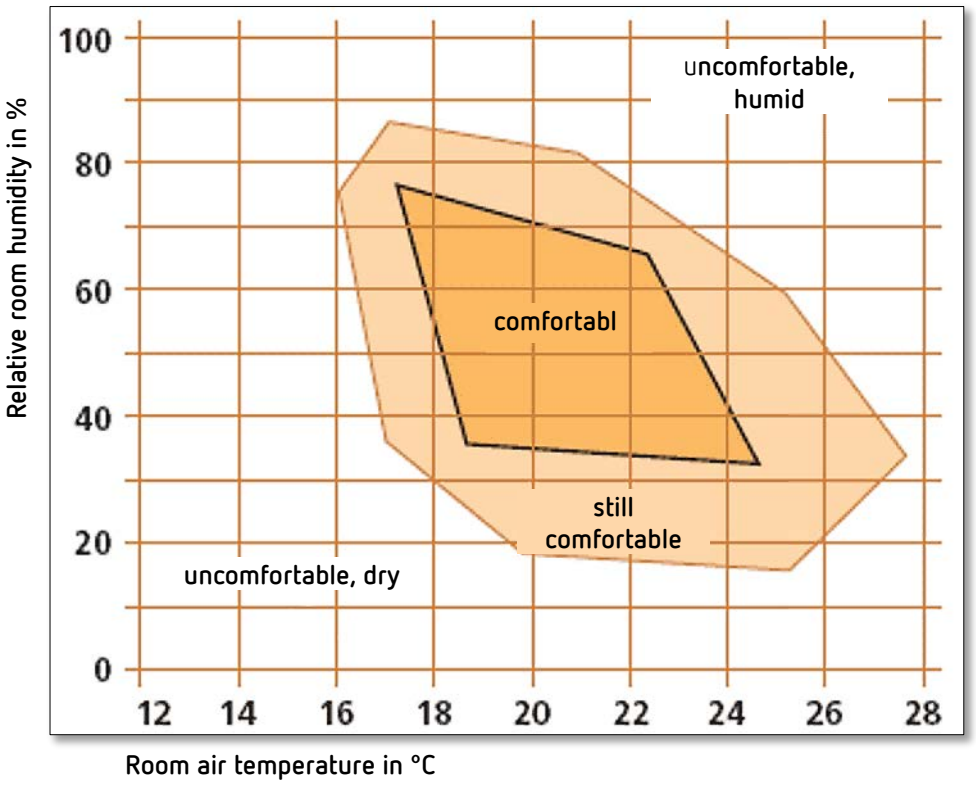
The calibration is started and takes approx. 20 minutes.
During the measurement, the CO_2 LED flashes with an ON time of 750 ms and an OFF time of 250 ms.
After finishing the fresh air calibration, a telegram is written to the *Alarm info* object („CO_2 CAL OK"), and the flashing of the LEDs stops.
In case of an error during the calibration, this is also shown by an alarm message ("CO_2 CAL ERR") and an error code.

During the calibration, the procedure it can be cancelled, by again pressing and holding the operating mode button for 5 s.

7.7 Comfort

In heating and air conditioning technology, the degree of comfort shows the range of the air conditioning in which people have a sense of well-being or not.

Comfort is defined by this diagram:

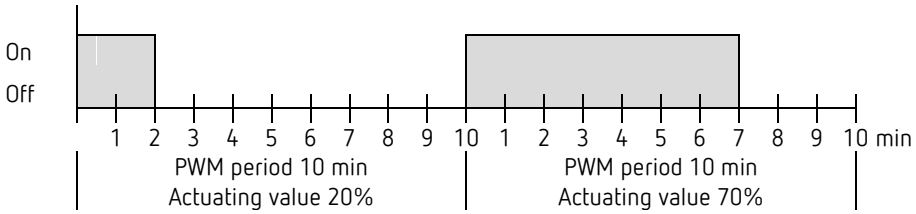


7.8 PWM cycle

7.8.1 Basic principle

The 50% control variable is converted into switch-on/switch-off cycles in order to achieve a heating output of 50%.
The actuator is switched on for 50% of the time and switched off for 50% of the time over a fixed period (10 minutes in our example).

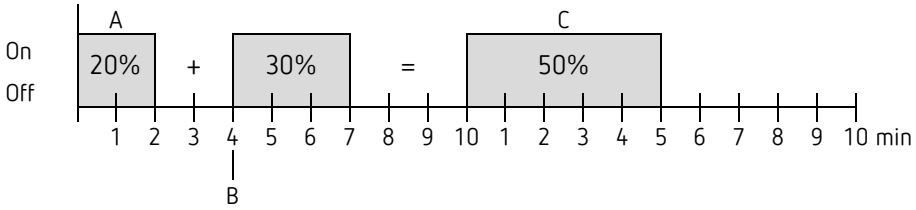
Example: 2 different turn-on times of 2 and 7 minutes indicate the implementation of 2 different actuating values that is once 20% and once 70% during a PWM period of 10 minutes.



7.8.2 Response to changes in the actuating value

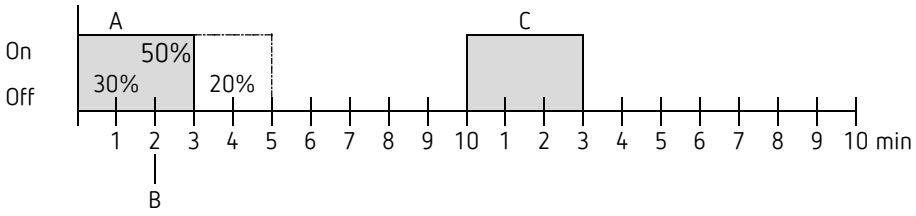
Every change in the actuating value is immediately transferred to the PWM cycle in order to respond to changes in the quickest possible time.

Example 1: The last actuating value was 20% (A).
A new actuating value of 50% is received during the cycle (B).
The output is immediately switched on and the missing 30% switch-on time is added.
The next cycle is executed with 50% (C).



If the rated switch-on time for the current cycle has already exceeded while receiving the new actuating value, the output is immediately switched off and the new actuating value is executed during the next cycle.

Example 2: The last control variable was 50% (A)
A new actuating value of 30% is received during the cycle (B).
The output is switched off after completing 30% of the PWM cycle and thus the new control variable is executed.



7.9 Operating mode as scene (RTC)

7.9.1 Principle

The current operating mode can be saved via an object with the scene functions and restored later at any time.
The current operating mode is allocated to the appropriate scene number when a scene is saved.
The previously saved operating mode is reactivated when a scene number is called.
This allows the device to be easily associated to each chosen user scene.
The scenes are permanently stored and remain intact even after the application has been downloaded again.

In order to save or call up the scene, the respective code is sent to the object *Operating mode as seen*.

Scene	Call up		Save		Scene	Call up		Save	
	Hex	Dec.	Hex	Dec.		Hex	Dec.	Hex	Dec.
1	$00	0	$80	128	33	$20	32	$A0	160
2	$01	1	$81	129	34	$21	33	$A1	161
3	$02	2	$82	130	35	$22	34	$A2	162
4	$03	3	$83	131	36	$23	35	$A3	163
5	$04	4	$84	132	37	$24	36	$A4	164
6	$05	5	$85	133	38	$25	37	$A5	165
7	$06	6	$86	134	39	$26	38	$A6	166
8	$07	7	$87	135	40	$27	39	$A7	167
9	$08	8	$88	136	41	$28	40	$A8	168
10	$09	9	$89	137	42	$29	41	$A9	169
11	$0A	10	$8A	138	43	$2A	42	$AA	170
12	$0B	11	$8B	139	44	$2B	43	$AB	171
13	$0C	12	$8C	140	45	$2C	44	$AC	172
14	$0D	13	$8D	141	46	$2D	45	$AD	173
15	$0E	14	$8E	142	47	$2E	46	$AE	174
16	$0F	15	$8F	143	48	$2F	47	$AF	175
17	$10	16	$90	144	49	$30	48	$B0	176
18	$11	17	$91	145	50	$31	49	$B1	177
19	$12	18	$92	146	51	$32	50	$B2	178
20	$13	19	$93	147	52	$33	51	$B3	179
21	$14	20	$94	148	53	$34	52	$B4	180
22	$15	21	$95	149	54	$35	53	$B5	181
23	$16	22	$96	150	55	$36	54	$B6	182
24	$17	23	$97	151	56	$37	55	$B7	183
25	$18	24	$98	152	57	$38	56	$B8	184
26	$19	25	$99	153	58	$39	57	$B9	185
27	$1A	26	$9A	154	59	$3A	58	$BA	186
28	$1B	27	$9B	155	60	$3B	59	$BB	187
29	$1C	28	$9C	156	61	$3C	60	$BC	188
30	$1D	29	$9D	157	62	$3D	61	$BD	189
31	$1E	30	$9E	158	63	$3E	62	$BE	190
32	$1F	31	$9F	159	64	$3F	63	$BF	191

7.10 Setpoint shift

The set point correction enables a *dynamic adjustment* of the setpoint to the outdoor temperature when cooling.

This function prevents too great a temperature deviation between the outside area and the cooled interior with high outside temperatures.

If the outdoor temperature exceeds a set threshold, adjustment is activated and a corresponding increase of the setpoint is calculated.

The current outside temperature for calculating the correction is received via object *Outside temperature*.

The set point correction is activated on the RTC ***Settings*** parameter page via the *Use set point correction with high outside temperatures* parameter and is set on the ***Set point adjustment*** parameter page.

The set point correction is internally linked to the RTC, so no bus connection is required.

7.10.1 Format of set point correction: Relative

Set point correction is sent as a temperature difference.
Below the set point correction threshold (*set point correction from*) the value 0 is sent.

If the set point correction threshold (*set point correction from*) is exceeded, the setpoint will be increased linearly depending on the change of the outside temperature.

Example: Calculated correction value
Set point correction from: 26 °C

Outdoor temp.	Adjustment							
	1 K/1 K	1 K/2 K	1 K/3 K	1 K/4 K	1 K/5 K	1 K/6 K	1 K/7 K	
20 °C	0 K	0 K	0 K	0 K	0 K	0 K	0 K	Correction value
21 °C	0 K	0 K	0 K	0 K	0 K	0 K	0 K	
22 °C	0 K	0 K	0 K	0 K	0 K	0 K	0 K	
23 °C	0 K	0 K	0 K	0 K	0 K	0 K	0 K	
24 °C	0 K	0 K	0 K	0 K	0 K	0 K	0 K	
25 °C	0 K	0 K	0 K	0 K	0 K	0 K	0 K	
26 °C	**1 K**							
27 °C	2 K	**1 K**						
28 °C	3 K	1 K	**1 K**					
29 °C	4 K	2 K	1 K	**1 K**				
30 °C	5 K	2 K	1 K	1 K	**1 K**			
31 °C	6 K	3 K	2 K	1 K	1 K	**1 K**		
32 °C	7 K	3 K	2 K	1 K	1 K	1 K	**1 K**	
33 °C	8 K	4 K	2 K	2 K	1 K	1 K	1 K	
34 °C	9 K	4 K	3 K	2 K	1 K	1 K	1 K	
35 °C	10 K	5 K	3 K	2 K	2 K	1 K	1 K	
36 °C	11 K	5 K	3 K	2 K	2 K	1 K	1 K	
37 °C	12 K	6 K	4 K	3 K	2 K	2 K	1 K	
38 °C	13 K	6 K	4 K	3 K	2 K	2 K	1 K	
39 °C	14 K	7 K	4 K	3 K	2 K	2 K	2 K	
40 °C	15 K	7 K	5 K	3 K	3 K	2 K	2 K	

7.10.2 Format of set point correction: Absolute

Sends the corrected setpoint to the bus for additional room thermostats.

This setpoint is calculated from:
Base setpoint without correction + dead zone + adjustment.

Example: *Set point correction from*: 25 °C, *start setpoint*: 20 °C, *dead zone* = 2 K

Outdoor temp.	Adjustment							
	1 K/1 K	1 K/2 K	1 K/3 K	1 K/4 K	1 K/5 K	1 K/6 K	1 K/7 K	
20	22.00	22.00	22.00	22.00	22.00	22.00	22.00	Setpoint
21	22.00	22.00	22.00	22.00	22.00	22.00	22.00	
22	22.00	22.00	22.00	22.00	22.00	22.00	22.00	
23	22.00	22.00	22.00	22.00	22.00	22.00	22.00	
24	22.00	22.00	22.00	22.00	22.00	22.00	22.00	
25	**23.00**							
26	24.00	**23.00**						
27	25.00	24.00	**23.00**					
28	26.00	24.00	24.00	**23.00**				
29	27.00	25.00	24.00	24.00	**23.00**			
30	28.00	25.00	24.00	24.00	24.00	**23.00**		
31	29.00	26.00	25.00	24.00	24.00	24.00	**23.00**	
32	30.00	26.00	25.00	24.00	24.00	24.00	24.00	
33	31.00	27.00	25.00	25.00	24.00	24.00	24.00	
34	32.00	27.00	26.00	25.00	24.00	24.00	24.00	
35	33.00	28.00	26.00	25.00	25.00	24.00	24.00	
36	34.00	28.00	26.00	25.00	25.00	24.00	24.00	
37	35.00	29.00	27.00	26.00	25.00	25.00	24.00	
38	36.00	29.00	27.00	26.00	25.00	25.00	24.00	
39	37.00	30.00	27.00	26.00	25.00	25.00	25.00	
40	38.00	30.00	28.00	26.00	26.00	25.00	25.00	

7.11 Temperature control

7.11.1 Introduction

If the device is not configured as a switching controller, it can alternatively be configured as a P or as a PI controller, whereby PI control is preferable.

With the proportional controller (P controller), the actuating value is statically adjusted to the control deviation.
The proportional integral controller (PI controller) is far more flexible, i.e. it controls dynamically, i.e. more quickly and more accurately.

To explain the function of both temperature controls, the following example compares the room to be heated with a vessel

The filling level of the vessel denotes the room temperature.
The water feed stands for the radiator output.
The heat losses of the room are shown by a discharge.

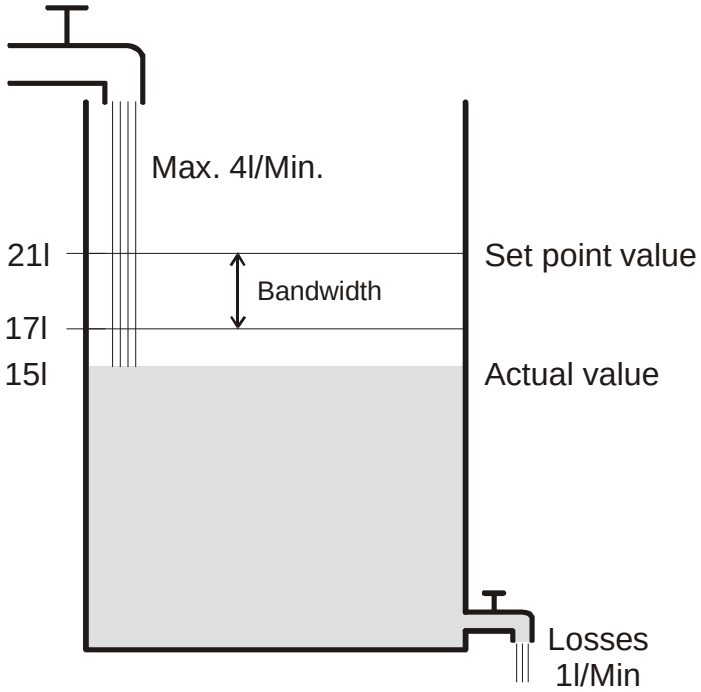
In our example, the maximum feed is assumed at 4 litres per minute and at the same time is the maximum heat output of the radiator.
This maximum output is achieved with an actuating value of 100%.
Accordingly, with an actuating value of 50% only half of the water volume, i.e. 2 litres per minute, would flow into our vessel.

The bandwidth is 4 l.
This means, the controller will control at 100%, as long as the actual value will be smaller or equal (21 l − 4 l) = 17 l.

Task:

Desired filling volume:
21 litres (= setpoint)
When should the feed be reduced, in order to prevent an overflow? :
4 l below the desired filling volume, i.e. at 21 l − 4 l = 17 l (= bandwidth)
Original filling volume
15 l (=actual value)
The losses are 1 l/minute

7.11.2 Response of the P controller



If the filling quantity is 15 l, there is a control deviation of 21 l – 15 l = 6 l
As our actual value lies outside the bandwidth, the control will operate the feed at 100%, i.e. with 4 l/minute.

The feed quantity (= actuating value) is calculated from the control deviation (setpoint – actual value) und the bandwidth.
Actuating value = (control deviation / bandwidth) x 100

The following table illustrates the behaviour and also the limits of the P controller.
Table 1

Filling level	Actuating value	Feed	Losses	Increase of filling level
15 l	100%	4 l/min	1 l/min	3 l/min
19 l	50%	2 l/min		1 l/min
20 l	25%	1 l/min		0 l/min

The last line shows that the filling level cannot be increased any more, because the inlet feeds as much water as can be discharged by the losses.
The result is a permanent control deviation of 1 l. The setpoint can never be achieved.
If the losses were increased by 1 l, the permanent control deviation would be increased by the same amount, and the filling level would never exceed the 19 l mark.
In case of a room, this would mean that the control deviation increases with decreasing outdoor temperature.

P controller as temperature controller

Just as in the previous example, the P controller behaves in a heating control.
The setpoint temperature (21 °C) can never be completely reached.

The permanent control deviation is increased the higher the heat losses, i.e. the colder the outdoor temperatures.

7.11.3 Response of the PI controller

In contrast to the pure P controller, the PI controller functions dynamically.
With this type of controller, the actuating value remains unchanged, even at a constant deviation.

At the first moment, the PI controller sends the same actuating value as the P controller.
However, this will be increased further the longer the setpoint will not be reached.
This increase is time-controlled over the so-called integration time.
During this calculation method, the actuating value will not be changed anymore when the setpoint equals the actual value.
In our example, this results in the balance between feed and discharge.

A good control depends on the adjustment of bandwidth and integration time with the room to be heated.
The bandwidth influences the increment of the actuating value change:
Large bandwidth = finer increments for the actuating value change.
The integration time influences the response time to temperature changes:
Long integration time = slow response.
Poor adjustment can result in either the setpoint being exceeded (overshoot), or the controller taking too long to reach the setpoint.

The best results are generally achieved using the standard settings or with the settings via installation type.

7.12 Continuous and switching control

A switching (2 point) control recognises only 2 statuses, On or Off.
A continuous control works with an actuating value between 0% and 100% and can thus exactly dose the energy input. This provides a pleasant and precise degree of control.

Table 2: Overview of control functions

Operating mode/stage	Type of control	Hysteresis
Heating	2-point/PI controller	positive
Cooling	2-point/PI controller	negative
Additional stage	2-point/P control	negative

7.13 Hysteresis

ⓘ Hysteresis determines the difference between a controller's switching on and off temperature.

It can be both positive and negative.
With a combination of heating and cooling control, it influences the amount of the dead zone.

Without hysteresis, the controller would activate and deactivate continuously, as long as the temperature lies within the range of the setpoint.

7.13.1 Negative hysteresis:

Heating: Is provided until the setpoint has been reached.
Afterwards, the heating is only switched on again when the temperature falls below the "Hysteresis set point value" threshold.
Cooling: Lasts until the "Hysteresis setpoint" threshold has been achieved.
Afterwards, it is only switched on again when the temperature rises above the setpoint.

Example of additional heating stage:
Additional stage with a setpoint of 20 °C, hysteresis 0.5 K and starting temperature 19 °C.
The additional stage is switched on and does not switch off again until the setpoint (20 °) is reached.
The temperature decreases, and the additional stage only is switches on at 20 °C-0.5 K= 19.5 °C.

Cooling example:
Cooling with setpoint of 25 °C, hysteresis = 1 °C and ambient temperature 27 °C.
The cooling is switched on and switches off again only when a temperature of 24 °C (25 °C − 1 °C) is achieved.
It switches on again when the temperature rises above 25 °C.

7.13.2 Positive hysteresis

Heating lasts until the temperature reaches the "setpoint + hysteresis " threshold.
The heating is only switched on again when the temperature falls below the set point value.

Heating example:
Heating with setpoint 20 °C, hysteresis = 1 °C and ambient temperature 19 °C.
The heating is switched on and only switches off again when a temperature of 21 °C (= 20 °C + 1 °C) is achieved.
It switches on again when the temperature falls below 20 °C.

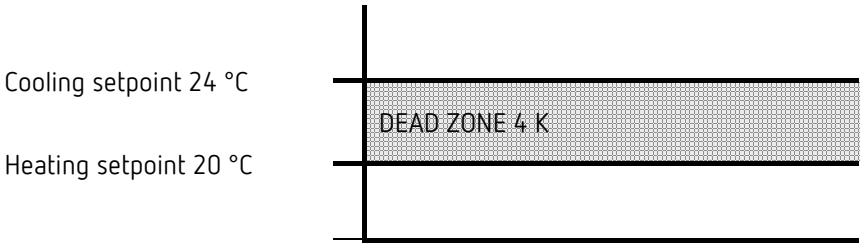
7.14 Dead zone

The dead zone is a buffer area between heating and cooling mode.
Within this dead zone, neither heating nor cooling occurs.

Without this buffer area, the system would permanently switch between heating and cooling. As soon as the setpoint was fallen below, the heating would be activated. After hardly reaching the setpoint, the cooling would immediately start, the temperature would fall below the setpoint and switch on the heating again.

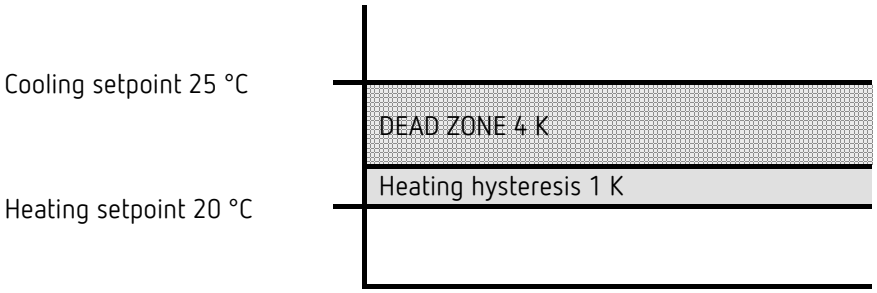
Depending on the type of control, the dead zone can be extended by the value of the hysteresis.

Case 1: Heating and cooling with continuous control



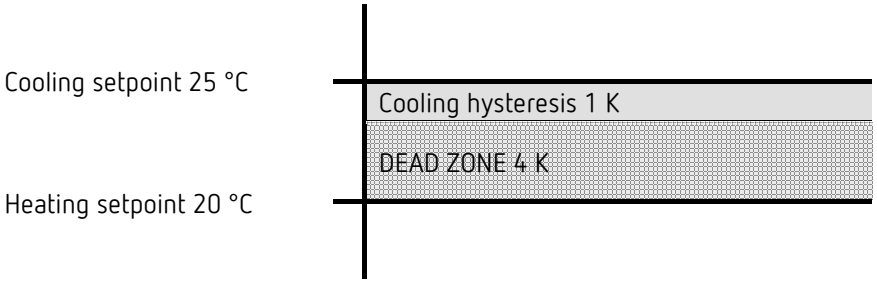
The dead zone (4 K) is not affected.

Case 2: Heating with 2-point control and cooling with continuous control



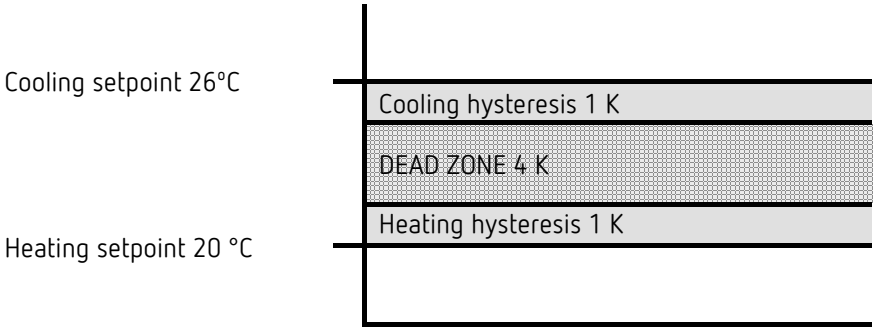
The dead zone (4 K) is increased by the value of the hysteresis (1K) and offsets the cooling set point value to 25 °C.

Case 3: Heating with 2-point control and cooling with continuous control



The dead zone (4 K) is increased by the value of the hysteresis (1K) and offsets the cooling set point value to 25 °C.

Case 4: Heating and cooling with 2-point control



The dead zone (4 K) is increased by the value of both hysteresis (2K) and offsets the cooling setpoint to 26 °C.

7.15 Operating mode selection

7.15.1 Priorities for operating mode selection

The operation mode selection between comfort, standby, night operation and frost protection can
happen in 3 different ways:

- Via the object *Operating mode preset*
- Manually at the device
- Via scene controls

All 3 possibilities are all on the same priority level.

> In principle the following applies: The last instruction overwrites the previous one.
> **Exception:** Frost mode via window contact has priority over all other operating modes.

Upon selection of the *presence button* parameter, the following also applies:
If a new operating mode is received on the object with the presence object set (*operating mode preset)*, it is accepted and the presence object is reset (only with presence button).

Reception of the same operating mode as prior to the presence status
(e.g. via cycl. sending) is ignored.

If the *presence object* is set during night/frost mode, it is reset after the configured comfort extension has expired (see below).

If the *presence object* is set during standby mode, the comfort operating mode is accepted without time restriction.

7.15.2 Determining the current operation mode

The current setpoint can be adjusted to the relevant requirements via the choice of operating mode.
The operating mode can be specified via the objects *operating mode preset, presence, and window setting*.
For this, there are two methods:

7.15.2.1 New operating modes

If *objects for determining the operating mode*" = *New*:... was selected on the ***Settings*** parameter page, then the current operating mode can be defined as follows:

Obj. Operating mode preset	Obj. Presence	Obj. Window position	Obj. Current operating mode
any	any	1	Frost/heat protection
any	1	0	Comfort
Comfort	0	0	Comfort
Standby	0	0	Standby
Night	0	0	Night
Frost/heat protection	0	0	Frost/heat protection

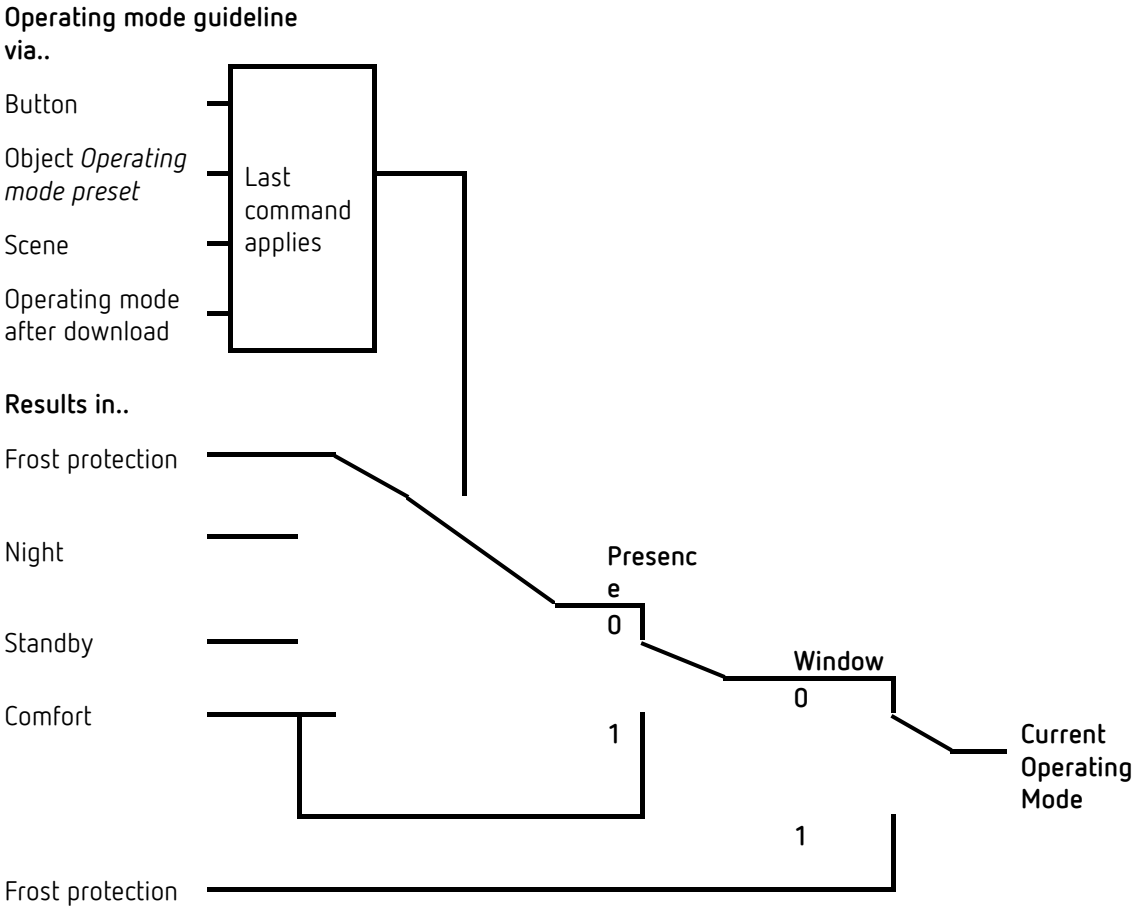
Typical application:

In the morning, the *Operating mode* object activates "Standby" or "Comfort", and in the evening, "Night" is activated via a time switch (e.g. TR 648).
During holiday periods, frost/heat protection is selected via another channel, also via the same object.

Object *Presence* is linked to a presence detector. If presence is detected, the controller switches to comfort operating mode (see table).

Object *Window status* is linked to a window contact via the bus (external input).
As soon as a window is opened, the controller switches to frost protection operating mode.

Determining the operating mode when using a presence detector



7.15.2.2 Old operating modes

If on the ***Settings*** parameter page *objects for determining the operating mode = Old:...* was selected, then the current operating mode can be defined as follows:

Obj. Night/standby	Obj. Comfort	Obj. Frost/heat protection	Obj. Current operating mode
any	any	1	Frost/heat protection
any	1	0	Comfort
Standby	0	0	Standby
Night	0	0	Night

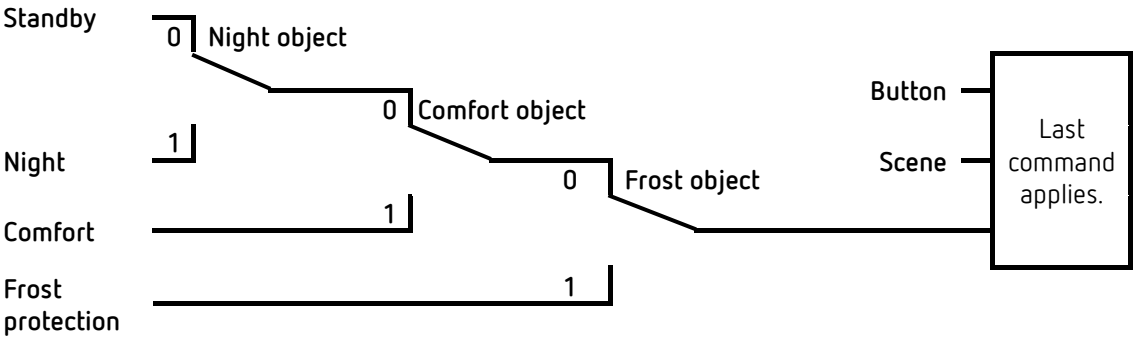
Typical application:

In the morning, "standby" operating mode, and in the evenings "night" operating mode is activated via the object by a time switch.

In holiday periods, frost/heat protection is selected on another channel via the object.

The object *Comfort* is linked with a presence detector. If presence is detected, the controller switches to comfort operating mode (see table).

The object *Frost protection* is linked with a window contact: As soon as a window is opened, the controller switches to frost protection mode.



The old method has 2 disadvantages over the new method:
To switch from Comfort to Night operating mode, 2 telegrams (2 time switch channels if necessary) are required:
The object *Comfort* must be set to "0", and object *Night/standby* to "1".

If the window is opened and then closed again during periods when "Frost/heat protection" is selected via the time switch, the "Frost/heat protection" mode is cleared.

7.16 Determination of the setpoint

7.16.1 Setpoint calculation in heating mode

See also: *Base setpoint and current setpoint*

Current setpoint during heating:

Operating Mode	Current setpoint
Comfort	Base setpoint +/- set point offset
Standby	*Base setpoint* +/- set point offset – *reduction in standby mode*
Night	*Base setpoint* +/- set point offset – *reduction in standby mode*
Frost/heat protection	configured *setpoint for frost protection mode*

Example: Heating in comfort mode.

Parameter page	Parameter	Setting
Setpoints	*Base setpoint after reset*	*21 °C*
	Reduction in standby mode (during heating)	*2 K*
Heating setpoints	*Maximum valid set point offset*	*+/- 2 K*

The setpoint was previously increased by 1 K using the + button.

Calculation:

Current setpoint = base setpoint + set point offset
= 21 °C + 1 K
= 22 °C

If operation is switched to standby mode, the current setpoint is calculated as follows:

Current set point = base setpoint + set point offset – reduction in standby mode
= 21°C + 1K – 2K
= 20 °C

7.16.2 Setpoint calculation in cooling mode

Current setpoint during cooling:

Operating Mode	Current setpoint
Comfort	Base setpoint + set point offset + dead zone
Standby	Base setpoint + set point offset + dead zone + increase in standby mode
Night	Base setpoint + set point offset + dead zone + increase in night mode
Frost/heat protection	configured setpoint for heat protection mode

Example: Cooling in comfort operating mode.
The room temperature is too high, the controller has switched to cooling mode

Parameter page	Parameter	Setting
Heating setpoints	*Maximum valid set point offset*	*+/- 2 K*
	Base setpoint after loading the application	*21 °C*
Cooling setpoints	*Dead zone between heating and cooling*	*2 K*
	Increasing in standby mode (during cooling)	*2 K*

The setpoint was previously lowered by 1 K on the device.

Calculation:

Current setpoint = base setpoint + set point offset + dead zone
= 21 °C - 1 K + 2 K
= 22 °C

Changing to standby mode causes a further increase in the setpoint (energy saving), resulting in the following setpoint.
Setpoint = base setpoint + set point offset + dead zone + increase in standby mode
= 21 °C - 1 K + 2 K + 2 K
= 24 °C

7.17 Set point offset

With this function, the user can increase or reduce the room temperature individually, as desired.

The current setpoint can either be offset via the object *manual set point offset*, or via the rotary control. [55]
See *Parameter: Rotary control function*. [56]

The offset limits are defined on the ***Setpoints*** parameter page via the *Maximum valid setpoint offset* parameter.

The offset always refers to the set base setpoint and not to the current setpoint.

Example [57] Base setpoint of 21°C, *function of the rotary control = base setpoint*:

If the value of +2 K is received, the new setpoint is calculated as follows:
21 °C + 2 K = 23 °C.
In order to afterwards take the setpoint to 22 °C, the difference to the set base setpoint (here 21 °C at the rotary control) is resent to the object, in this case 1 K (21 °C + 1 K = 22 °C).
See object *Manual set point offset/set point offset at rotary control*.

[55] Only Amun 716 S
[56] Only Amun 716 S
[57] Only Amun 716 S

7.18 Base setpoint and current setpoint

The *base setpoint* is the standard temperature for the comfort mode and the reference temperature for reduction in standby and night modes.
The base setpoint can be defined directly at the rotary control [58], or via the object base setpoint (see parameter *function of the rotary control* [59]).
The configured base setpoint (see *base setpoint after loading application*) is stored in the object *base setpoint* and can be changed any time via the bus by sending a new value to this object (only when *function of the rotary control = manual offset*).
After reset (restoration of the bus supply), the previously used base setpoint will be restored.

The *current setpoint* is the value that actually is used for control. It is the result of all reductions or increases associated with the operating mode and control function.

Example [60]**:** At a base setpoint of 22 °C and a reduction in night mode of 4 K, the current setpoint (in night mode) is: 22 °C - 4 K = 18 °C. During the day (in comfort mode) the current setpoint is 22 °C (in heating mode).

The formation of the current setpoint on the basis of the basic setpoint can be observed in the block diagram on the next page:
The base setpoint on the left is specified via object, or set on the device.
The current setpoint is on the right, i.e. the value upon which the room temperature is effectively controlled.

As you can see in the block diagram, the current setpoint depends on the operating mode (5) and the selected control function (4).

The base setpoint limits (2) prevent an incorrect base setpoint from being specified at the object. These are the following parameters:

- *Minimum valid base setpoint*
- *Maximum valid base setpoint*

If the setpoint is outside the configured values for frost and heat protection, because of a set point offset, it is restricted to these values by the safety limits (11).

[58] Only Amun 716 S
[59] Only Amun 716 S
[60] Only Amun 716 S

7.19 CO_2 calibration

The CO_2 measurement value can be corrected either as an offset via object 84 or with a reference value via object 85.

EXAMPLE:
Amun 716 S sends a value of 500 ppm.
A reference instrument reports a CO_2 value of 450 ppm,
i.e. a difference of -50 ppm must be corrected.

This leads to 2 possibilities:

- Send -50 to object 84 (DPT9.* 2 byte floating-point number).
- Send 500 to object 85 (DPT9.008).

The offset value remains active after reset.
The current offset value can be read out via the bus at any time.

ⓘ If the resulting CO_2 value is less than 400 ppm when setting the offset, the offset value will be adjusted to result in a CO_2 value of 400 ppm. [61]

ⓘ By sending zero to one of the two objects, the offset will be set to 0.
The same applies when performing a fresh air adjustment or a KNX master reset.

ⓘ This function is available from application programme version 3.1.
Only for devices manufactured as of 2113, firmware version 3.0.1 (18 01).

7.20 Comparator

This function can be used to compare different values.
The resulting final value is sent out via object.
For each of the three inputs, a control value (CO_2, humidity and RTC) or an input object (DPT5.1) can be configured.

The output value can then be the minimum value, the maximum value or the calculated mean value of all active inputs. If all three inputs are not active, the comparator is also not active.

The comparator does not send until all configured control values are available, as it can take a little longer until the first CO_2 value is reported by the measuring module.

If a control value (CO_2 ventilation or humidity ventilation) is blocked via an object, it will still be evaluated.

[61] In 2013, an average concentration of 400 ppm was determined in the atmosphere. Normally, this value can be taken as a reference value.

7.21 Firmware version

Information for the advanced. [62]

If the *Firmware Version* diagnostic object is read out [63], the version number is displayed in the ETS as DPT217.001 in the form of 2 hexadecimal numbers.

DPT	Info
217.001 DPT Version	18 01

The number 18 01, for example, corresponds to the version number V3.0.1 and is composed as follows:

A 16-bit bit pattern:

Magic number					Version number					Revision number					
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U

Contains the 3 digits of the version number

Magic number					Version number					Revision number					
U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
3					0					1					

16-bit bit pattern

0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1

If the 16-bit bit pattern is separated in the middle, 2 hexadecimal numbers result. They are displayed in this form by the ETS when reading out the group address.

0	0	0	1	1	0	0	0
1				8			

0	0	0	0	0	0	0	1
0				1			

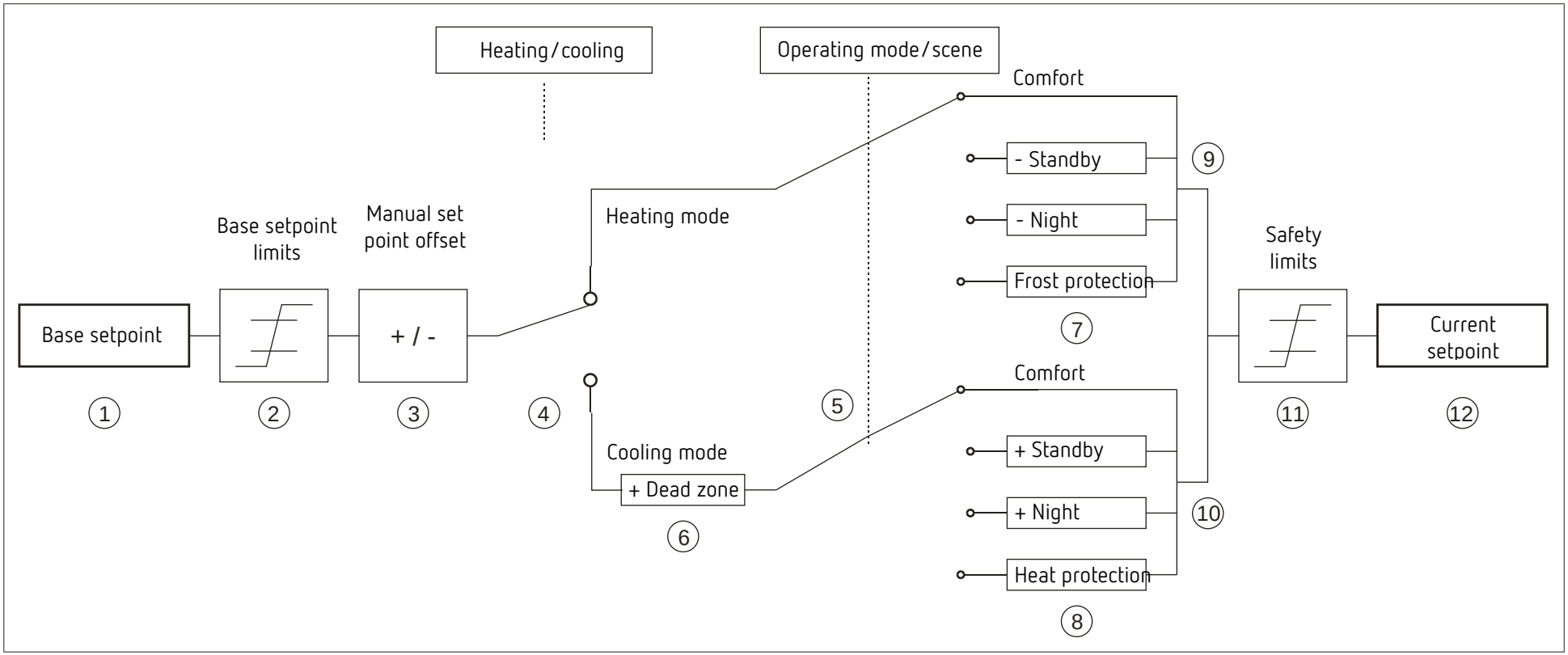
Result:
V3.0.1 => 18 01

The first subsequent version numbers would thus look like this:

ETS	Firmware version
18 02	3.0.2
18 03	3.0.3
18 04	3.0.4
18 05	3.0.5

[62] Requires knowledge of binary and hexadecimal number systems.
[63] Diagnosis menu/ group monitor

7.21.1 Setpoint calculation



1 Preset base setpoint from object or rotary control
2 Max. and min. valid base setpoints
3 Manual set point offset
4 Change between heating and cooling: Automatically or via object
5 Selection of operating mode, by operator, object, switching program or scene.
6 The setpoint is increased in cooling mode by the amount of the dead zone
7 The setpoint is replaced by the setpoint for frost protection mode
8 The setpoint is replaced by the setpoint for heat protection mode
9 Setpoint after reductions caused by the operating mode
10 Setpoint after increases caused by the operating mode
11 The limits for frost and heat protection must be adhered to
12 Current setpoint after increases, reductions and limits caused by the operation